



NEAR EAST UNIVERSITY

FACULTY OF ECONOMICS AND ADMINISTRATIVE SCIENCES

DEPARTMENT OF COMPUTER INFORMATION SYSTEMS

2005/2006 Fall Term

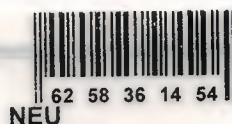
CIS 400

Graduation Project

Retail Banking System

Submitted To : Department of Computer Information Systems
Submitted By : Fatih ÇAY (20020704)

Lefkoşa,
2006





ACKNOWLEDGMENTS

For giving the chance of improving myself in this university with these conditions, I want to express my gratitude to Near East University.

First of all I want to thank my supervisor **Mr. Tansel Devin** for his advices and his invaluable guidance in my Project . He forced my creativity to improve myself and my project. I am also grateful to him for giving his valuable time to me.

I also want to thank **Miss. Nadire Çavuş** who help me all my advisory problems, and also thank to **Mr.Yalçın Akçalı** for sharing endless experiments in system analysis methods.

Secondly I would like to thank all my friends especially **CanerCakir and Zekai Kıray** who causes question marks in my mind and forced me to solve the problems I faced.

Especially I want to thank **Emel Yazici** for being my inspiration and for her support.

At last **My Family**, the ones who deserves the most gratitude for giving their endless support and encouragement not only for my study, for all my life. I really thank you. It could not be done without you.

ABSTRACT

EMFA Bank was requested from me, to preparing computerised Retail Banking System to handle their retail banking routines.

I thought and I designed a system that every part of the application must be module by module. Main aim of the project was Fund Transfer and preparing SWIFT report for foreign bank's transfers so my fund transfer part has to be separated from crediting and other security and account transactions.

I prepared a database design and showed it to Mr. Yalçın Akçalı who is the CEO of EMFA Bank. Mr. Akçalı notified me to do firstly fund transferring module to handle it fastly with computers.

I designed the system's fund transfer module, security module, customer and accounts module and applied it on computer with using Oracle Database and Java Object Oriented Programming Language.

Oracle database support system to strong data warehouse, secure, thrustworthy, easily accessible if users have enough permission and faster. Oracle database is suitable for kind of new banks. If banks grow and database needs to be upgrade Oracle supports upgrade and does it easily. Bank computer center ersonel could ake easily back-ups for keeping them safe with avoiding dangers of nature and computer crashes.

Java Programming laguage supports Object Oriented Programming to do it's job easily and safe. Object oriented programming is the newest technology in application programming area. Also Java supports desktop programming and web programming together. If bank needs to publish internet banking program on web my program easily convert to internet programming to using internet banking.

Program needs only 1.5 MB of hdd for running. Java and Oracle orientation supports instance connection for keeping database ports open to other users. This benefit gets more efficient network and database connection. Thats mean Retail Banking System could be faster and needs less resource and less effort to work stable.

System Processes:

- 1) User Process
- 2) Customer Process
- 3) Account Process
- 4) Fund Transfers Process
- 5) Credit Process
- 6) Credit Type Process
- 7) Bank Process
- 8) Country Process
- 9) Reports Process

INTRODUCTION

Retail Banking Systems has major components like fund transferring, crediting, interested or noninterested accounts. Banks need faster transactions, security and trusted non data loss systems to keep transactions. Many of the banks have transferred their system to computer for computerized banking. Computerising has got lots of benefit to banks and their workers. Computerising gets more efficient time management, customer – bank relation and improving workers performance.

The Retail Banking System has been added new inventions to their system with computerising. ATM (Automated Teller Machine) get a chance to customers drawing money from bank without any branch or banking personnel. Internet banking supports user to do their all banking operations without any bank personnel or banking hardware, customers could do every banking operation on internet instantly. Money moves fastly and securely than before.

Also country economy has improved with banking transactions on computer because money can transfer easily with less effort and less expence, so citizens of country become spend money more faster and getting money more faster.

Computerised Retail Banking Systems do crediting be secure and be faster too. Crediting is possible now in 20 minutes in some of banks with phone or internet. Customers are more comfortable and more equipped with computerising banking. Bank's managers are more faster in decision making because they could take their related reports faster from computer than hard recorded.

As a conclusion Computerised Retail Banking Systems are useful for both customers and bank employees. Both sides could be take different benefits and less expences than before.

Computerised Retail Banking System prepared by Java Programming Language and Oracle Database for EMFA Bank.

System Processes:

1) User Process:

This process gets username & password from users or managers to check if they have permission to enter program.

2) Customer Process:

This process supplies adding new customer to database.

3) Account Process:

This process supplies adding new account to database with account owner, account amount.

4) Fund Transfers Process:

This process is the main process of the system. It supplies transferring money :

From customer to destination name,

From customer to destination customer,

From customer to destination customer's account,

From customer to customer's other accounts,

From customer to destination foreign bank customer,

From customer to destination foreign bank customer

5)Credit Process:

This process supplys to give a defined type credit to an account.

6)Credit Type Process:

This process uses by managers to add new credit type to database.

7)Bank Process:

This process uses by managers to add other banks to database.

8)Country Process:

This process uses by managers to add new country to database.

9)Reports Process:

This process uses for taking reports when necessary to inform both users and managers.

DESIGN OF THE SYSTEM

EXPLANATION

Minimum System Reuirements:

For Application:

Intel 486 Based Cpu
64 Mb Ram
1.5 Mb Free Hdd Area
Monitor
Any Java Supported Operating System

For Oracle Database Server:

Intel 486 Based 800 Mz Processor
128 Mb Ram
3.5 Gb FreeHdd Area
Ms Windows Nt or 2000 Server

Recommended System

For Application:

Intel 486 Based 1 Ghz Cpu
128 Mb Ram
1.5 Mb Free Hdd Area
Monitor
Any Java Supported Operating System

For Oracle Database Server:

Intel 486 Based 3.00 Ghz Processor
512 Mb Ram
3.5 Gb FreeHdd Area
Ms Windows 2003 Server

System Build Software

Oracle 9.02 Database For Data Storing
Java SDK1.5 For Application

I used Oracle and Java because Oracle is the most trusted and reliable database on this range and Java programing language is an open source and free tool to programming. Also Java gets system to be faster and hard secured.

Banking System must be secured ,faster and trusted.

EMFA Bank Retail Banking System

1)Project Identification And Selection

Aim Of The Proect:

The Retail Banking System solution was necessary for EMFA Bank to automate fund transfers,crediting and Bank has to store them in a database securely.

Bank also decrease the worker size with this program because program gets rid of checking prblems and more effort to record customers and their account transactions.

New system gets faster transactions,secure data entry and avoiding errors for fund transfers and crediting because system checks every details of customer to be enough account or account credited but has not closed up to new crediting date.

Project Boundaries:

- 1)Taking customer information to keep them in an database which is Oracle.
- 2)Keeping customer's accounts for adding ar drawing money from accounts.
- 3)Sending and receiving fund transfers between customers,accounts,other banks with using SWIFT International bank to bank fund transfer system.
- 4)Preparing SWIFT document for SWIFT transfers.
- 5)Calculating and storing bank's revenues from fund transfers with SWIFT.
- 6)Helping to User when gives a credit to user if it's has credited account.If customer has a credited account ,customer could not get any credit with this account.
- 7)Calculating credit installment number and showing that when customer pays an inssallment.
- 8)Calculating credit installment amount monthly.
- 9)Storing credit types and their ratios for calculating bank revenues.
- 10)Calculating and taking bank credit revenue from customer credit.
- 11)Taking monthly credit installments from customer.
- 12)Adding amount to customer account.
- 13)Drawing money from customer account
- 14)Keeping which user did any process with accounts to further disaggreements.
- 15)Giving whole or spesific reports about customer,account,fund transfers,bank revenues and crediting customers.

Resource Limits:

Designing,building and installment of system has a time limit up to 20-February-2006 because of the system will be operated on this date.System firstly building for fund transfers and if time will be enough crediting system will be add as soon as posible.

System needs a computer for deploying Oracle Database,client side will be work on any regular computer.

2)Project Initiation And Planning

Technical Feasibility :

Hardwares that are necessary are fallows :

- a. A server computer with MS server edition operating system for Oracle Database
- b. Client computers how many user will be use this Java aplication

Operational Feasibilty :

Managers has to learn how new system could be use effectively,because they will assign the credit type,new banks for SWIFT and new countries for ne banks.

Users has to learn their part of the system fund transfers or crediting.

If Users does not use computer they has to take a short system using course for How to System could be operated.

Sysem is user friendly so user could use system easily with taking short course.

Economical Feasibilty :

Oracle server cost , Oracle server computer and other user computers could be easily buys by EMFA Bank because system could be easily earns their money from banking transactions to avoiding errors.

3) Analysis

Requirement Determination :

Requirements Are Taken From Managers and Users Of EMFA Bank.

- 1) Speed of fund transfer.
- 2) Exact money has to be transfer to exact account,name or foreign bank.
- 3) Trusted datastore for future.
- 4) Increase user transfer and crediting performance.
- 5) Prevent data duplications (account,customer,crediting).
- 6) Prevent loss of data.
- 7) Easy and quick access to data.
- 8) Simple and easy user graphical interface.
- 9) Secure datas from outside attacks.
- 10) Restrict some part of the operations(Giving users to ranks) for reaching their related ranks.
- 11) Preventing loss of amount errors for fund transfer.
- 12) Preventing to giving new credit to credited account.
- 13) Security on trasactions and database.

Existing System Capabilities:

EMFA Bank has done all transactions with hand.

Documentation has been taken by hand and it's storing on paper.Users has to find all necessary information from hard reports.

Managers was not take reports when they need instantly.

New System Providences:

New system is faster than old system because system has a database and automated trnsactions for users.

Also new system is secure with ranking system.Back up could be take easily from computer and it takes only a cd.

System supports user to avoid fund transfer and crediting errors.

Managers could take reports easily and faster.

Users could check customers account amounts easily.

System could be recover with old back ups i its needed.

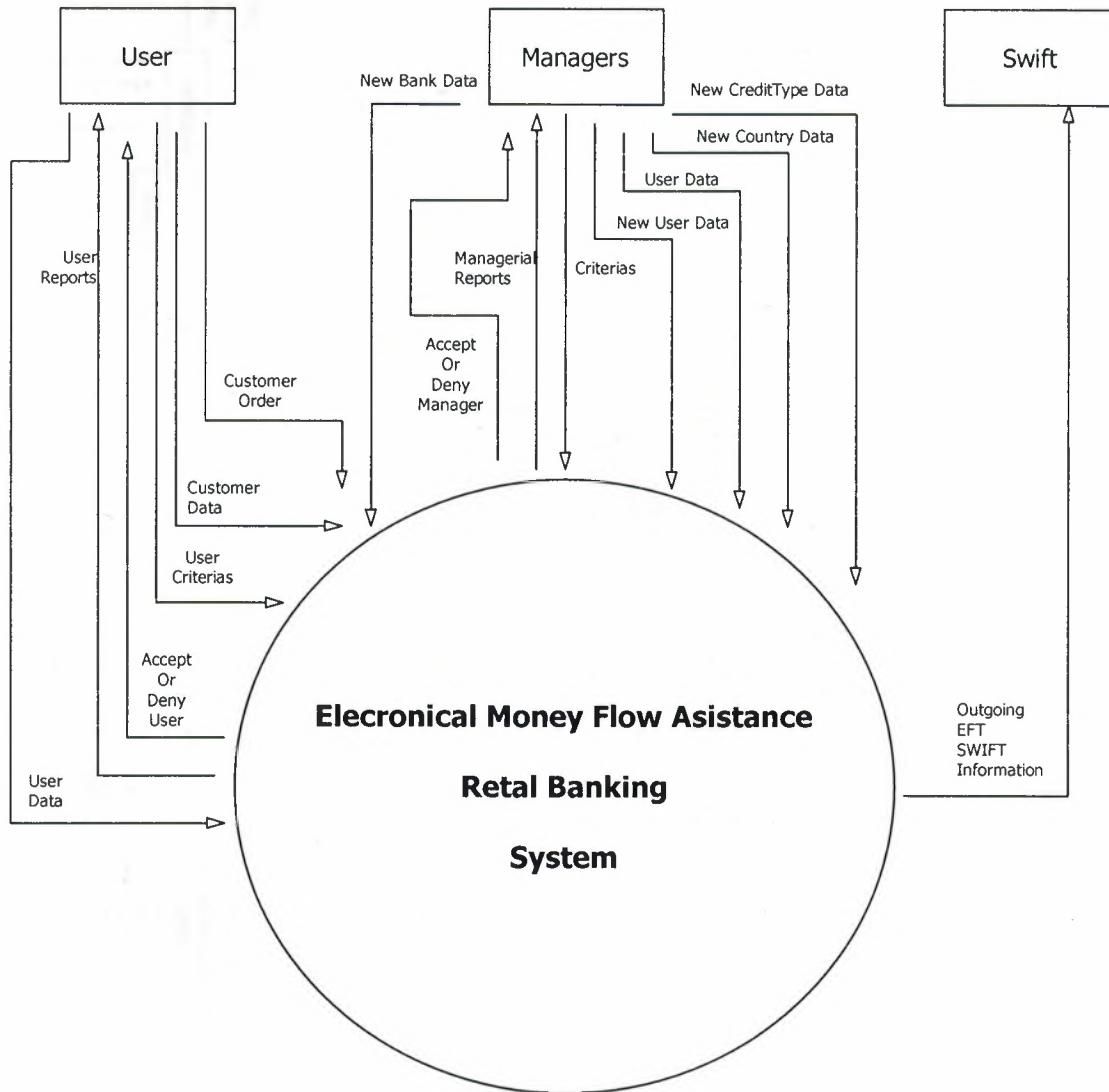
System avoids data loss it supports to keep minimally.

Database:

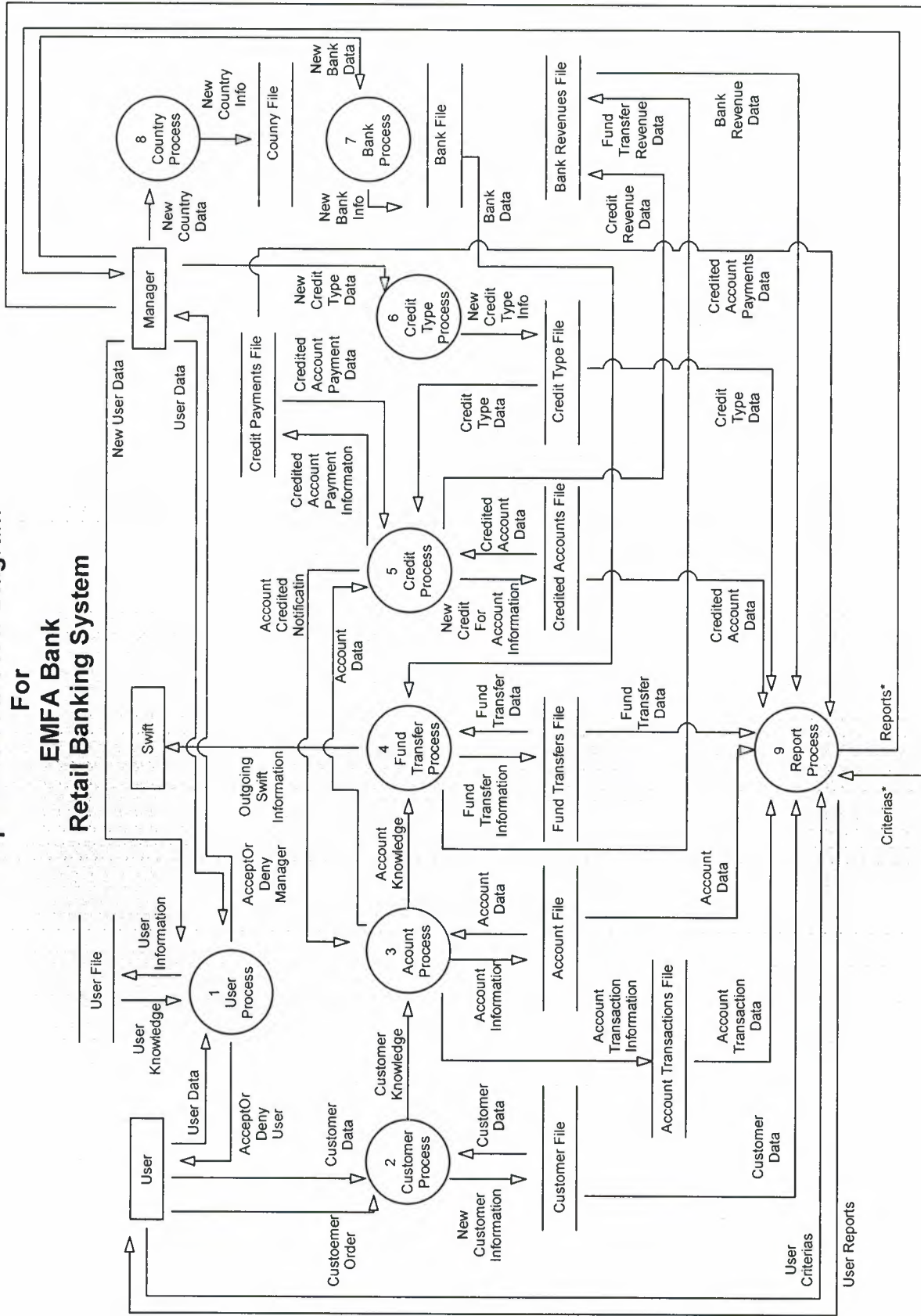
System has 11 table in Retail Banking (RB) database.

- 1)Users,
- 2)Customers,
- 3)Accounts,
- 4)Account Transactions,
- 5)Fund Transfers,
- 6)Banks,
- 7)County,
- 8)Credited Accounts,
- 9)Credit Payments,
- 10)Bank Revenues,
- 11)Credit Types

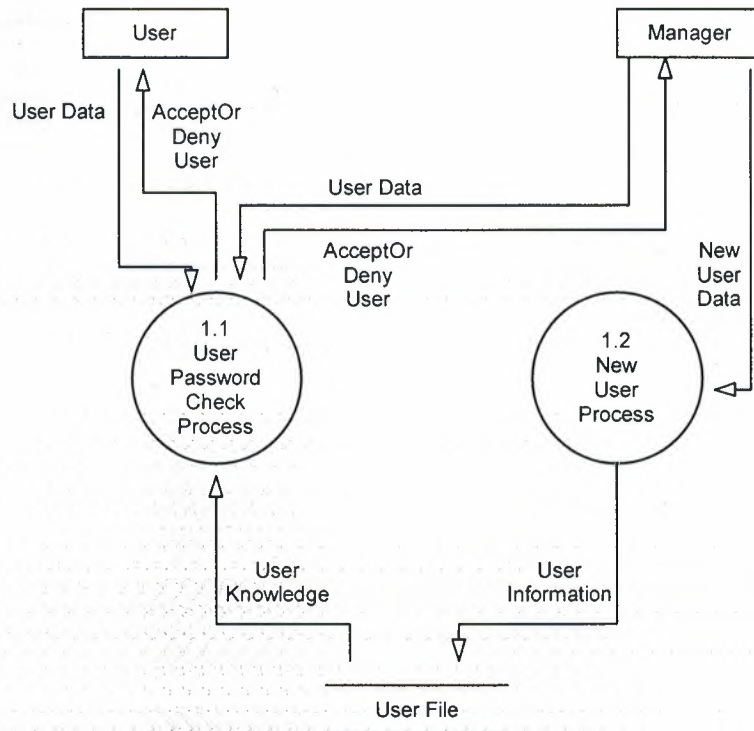
Context Data Flow Diagram For EMFA Bank Retail Banking System



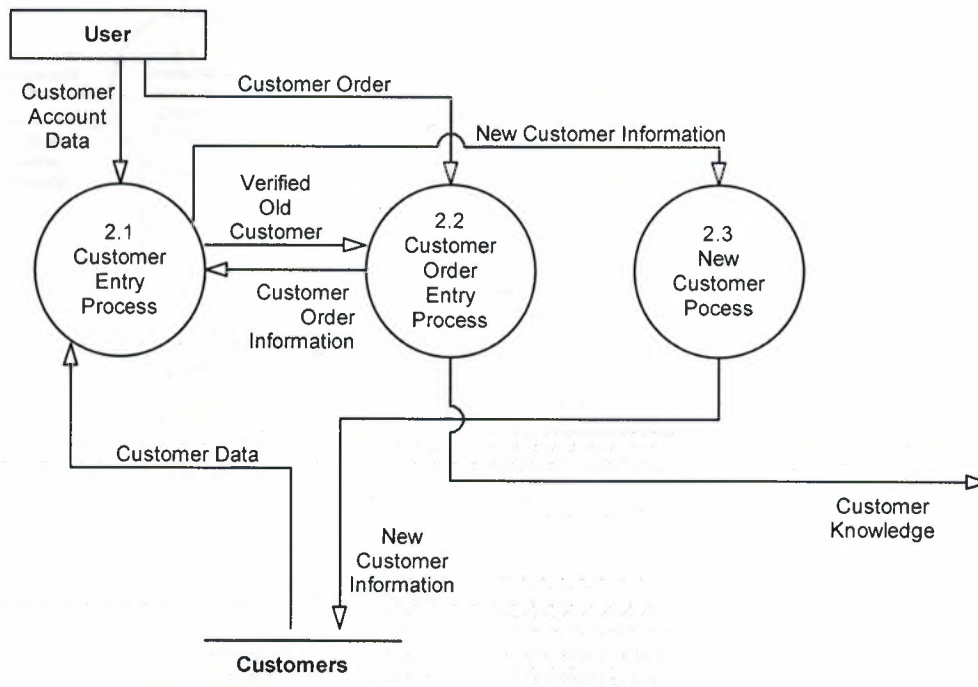
Top Level Data Flow Diagram For EMFA Bank Retail Banking System



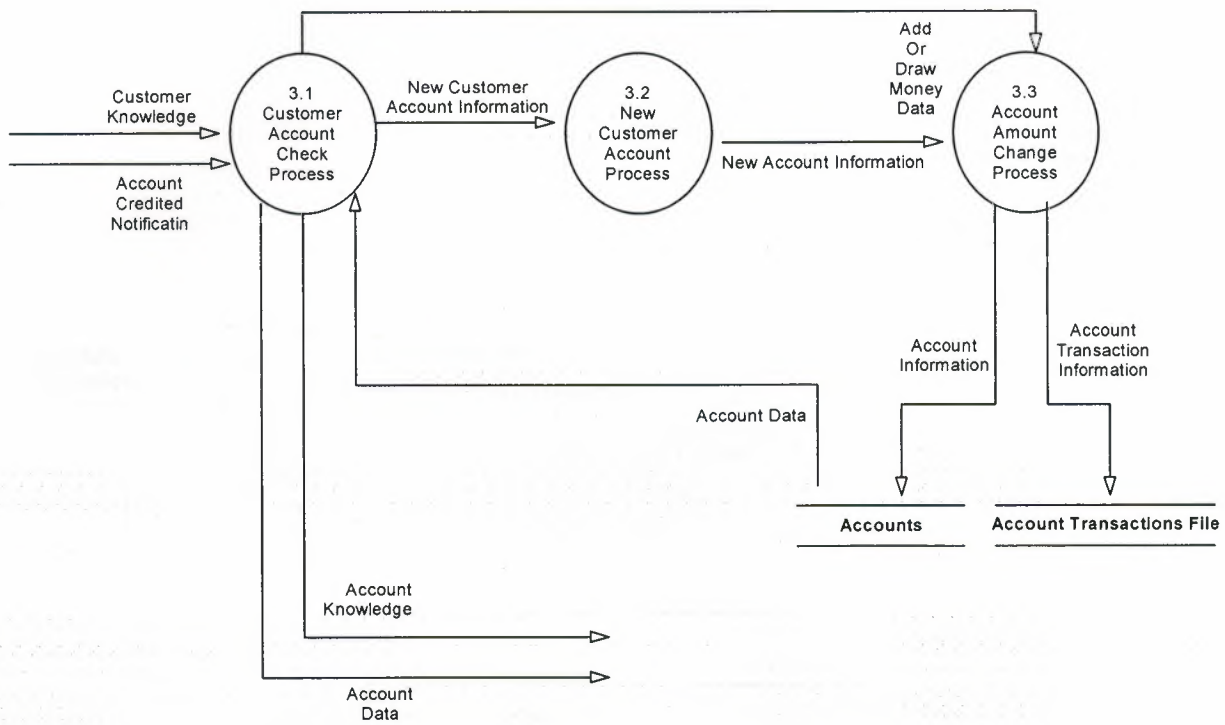
1
User Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



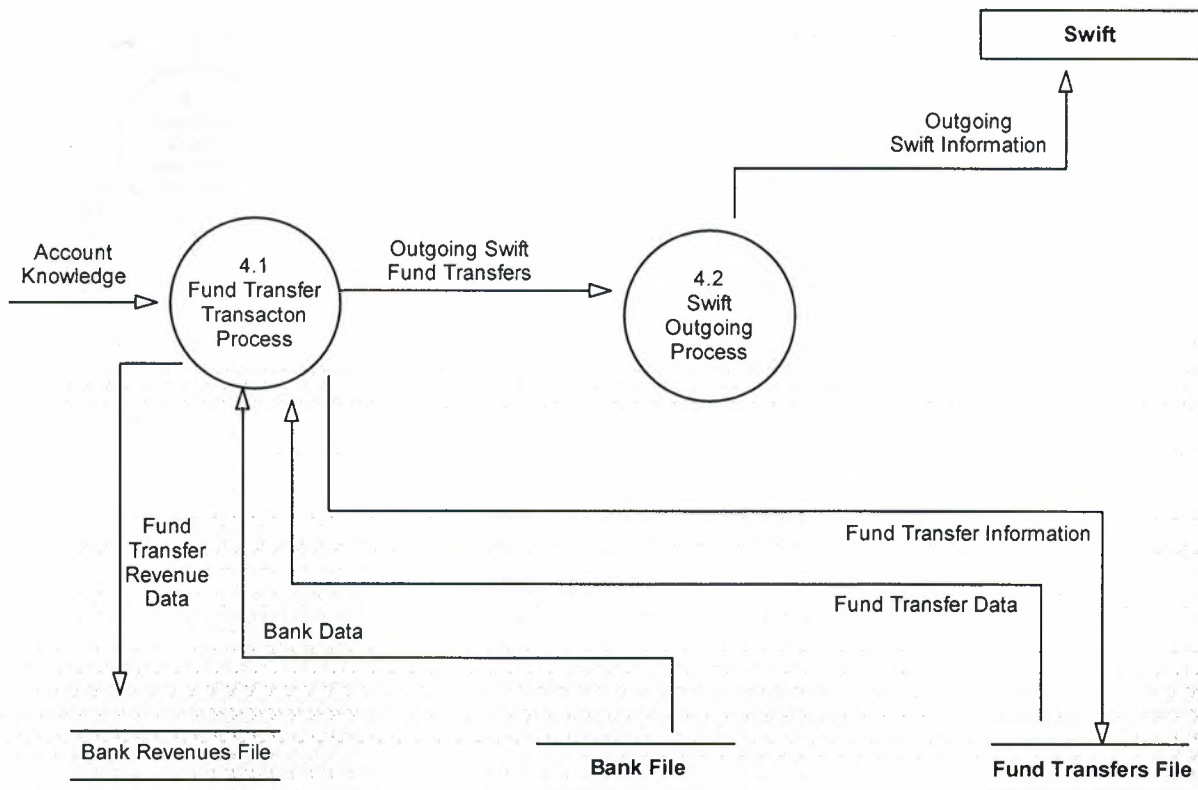
2
Customer Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



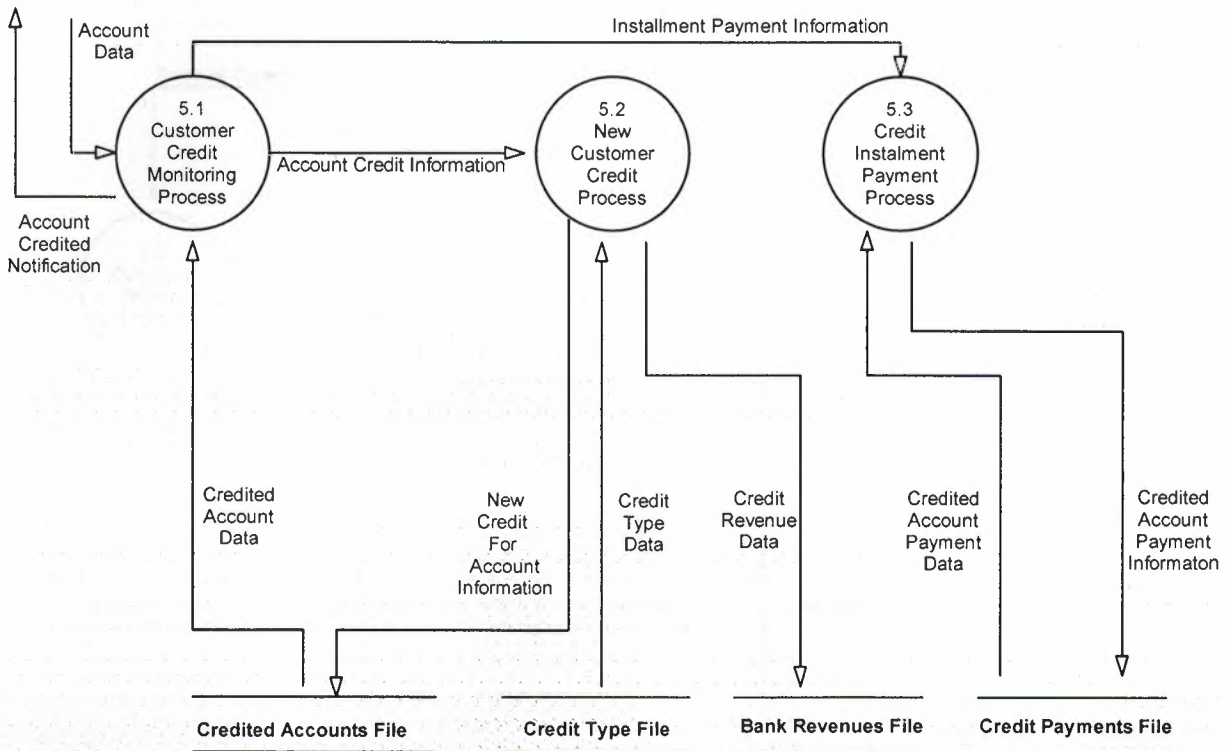
3
Account Transaction Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



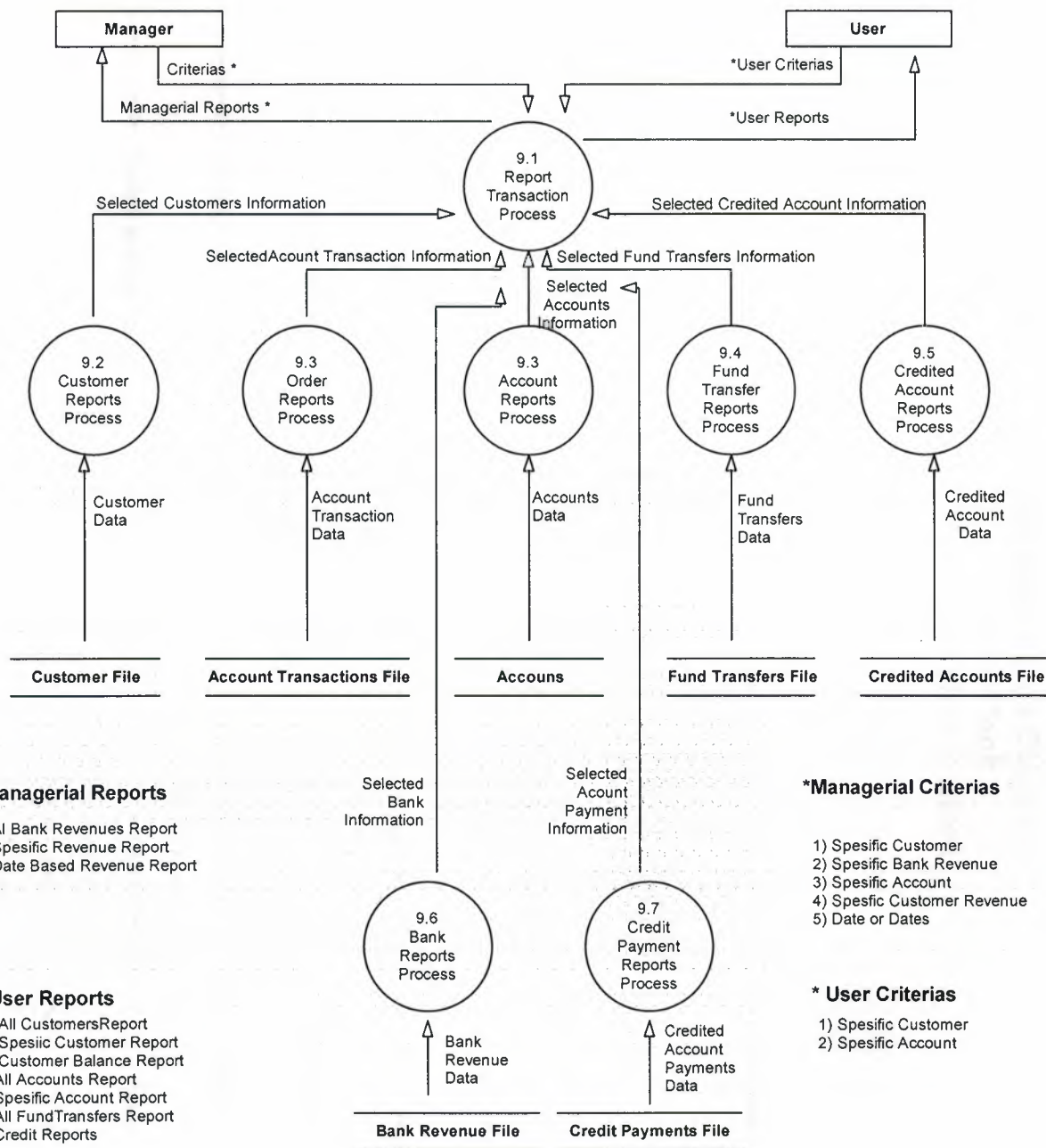
4
Fund Transfer Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



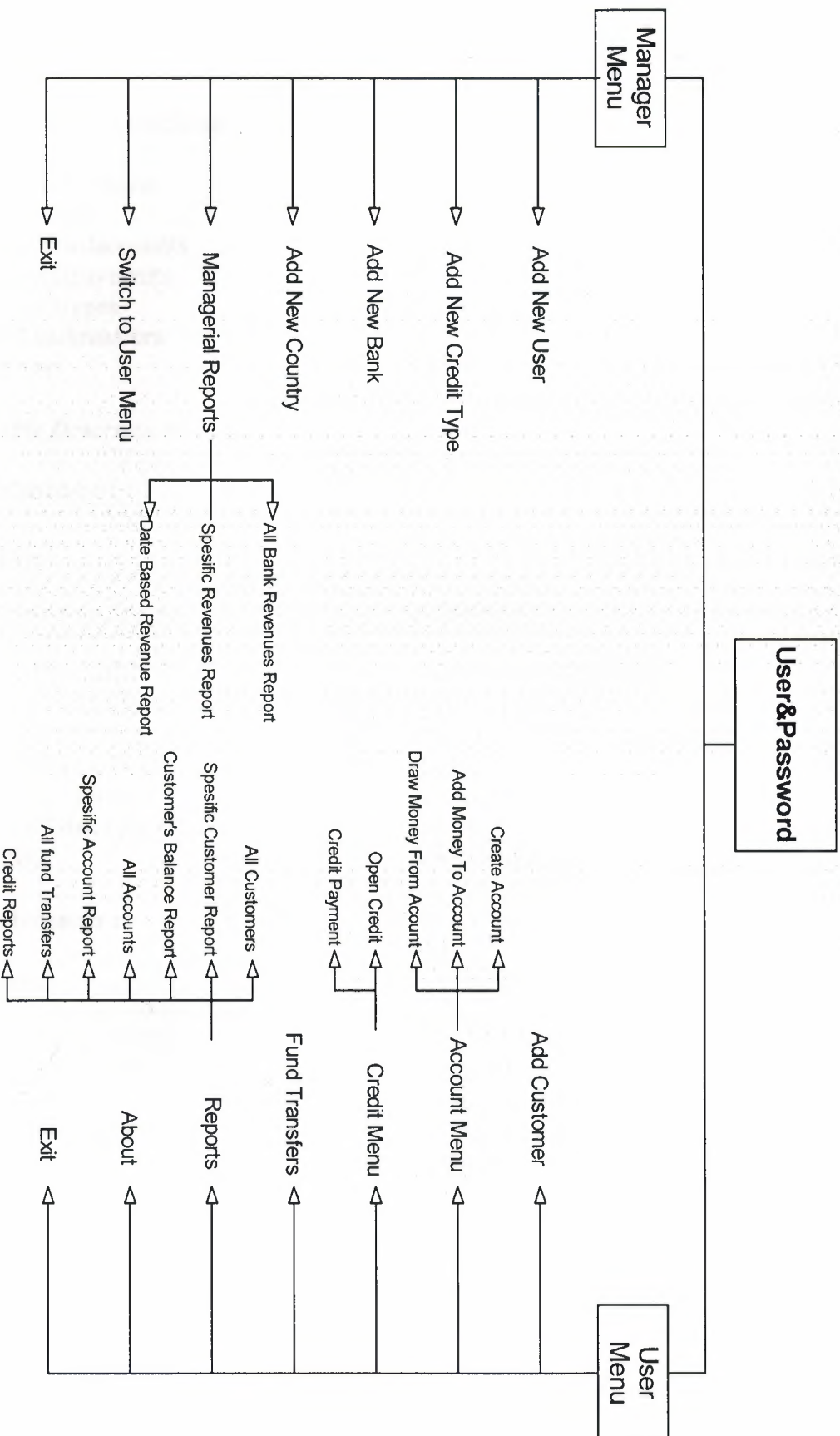
5
Credit Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



9
Reports Process
Detailed Data Flow Diagram
For
EMFA Bank
Retail Banking System



System Block Diagram EMFA Bank Retail Banking System



Database Design

Tables:

- 1)Customers
- 2)Accounts
- 3)Accounttransactions
- 4)Banks
- 5)Bankrevenues
- 6)Country
- 7)Creditedaccounts
- 8)Creditpayments
- 9)Credittypes
- 10)Fundtransfers
- 1)Users

Table Descriptions:

1)Customer :

Name	Type
CUSTNUM	NUMBER(11)
CUSTNM	VARCHAR2(20)
CUSTSNM	VARCHAR2(20)
CUSTGENDER	VARCHAR2(7)
CUSTPHONE	VARCHAR2(11)
CUSTREGDATE	DATE
CUSTADDRESS	VARCHAR2(50)
CUSTBIRTHDATE	DATE
USR	VARCHAR2(10)

2)Accounts :

Name	Type
ACCOUNTNUM	NUMBER(6)
ACUSTNUM	NUMBER(11)
AAMOUNT	NUMBER(12)
ACREDITED	VARCHAR2(3)
USR	VARCHAR2(10)

3)Accounttransactions :

Name	Type
ATRNUM	NUMBER(12)
ATRACCOUNT	NUMBER(6)
ATAMOUNT	NUMBER(12)
ATTYPE	VARCHAR2(4)
ATDATE	DATE
USR	VARCHAR2(10)

4)Banks :

Name	Type
BANKCODE	NUMBER(12)
BANKNAME	VARCHAR2(20)
COUNTRY	VARCHAR2(2)
USR	VARCHAR2(10)

5)Bankrevenues :

Name	Type
REVID	NUMBER(12)
REVCUSTOMER	NUMBER(11)
REVACCOUNT	VARCHAR2(6)
REVAMOUT	NUMBER(12)
REVDESC	VARCHAR2(21)
REVENUE DATE	DATE

6)Country :

Name	Type
COUNTRYID	VARCHAR2(4)
COUNTRYNAME	VARCHAR2(50)
USR	VARCHAR2(10)

7)Creditedaccounts :

Name	Type
CAID	NUMBER(2)
CANUMBER	NUMBER(6)
CAAMOUNT	NUMBER(12)
CTYPE	NUMBER(2)
CREDITCLOSED	VARCHAR2(1)
TOTINSNUM	NUMBER(2)
INSLEFT	NUMBER(2)
INSAMOUNT	NUMBER(12)
AMOUNTLEFT	NUMBER(12)
USR	VARCHAR2(10)
OPENDATE	DATE

8)Creditpayments :

Name	Type
PAYMENTID	NUMBER(7)
CUSTNUM	NUMBER(11)
ACCNUM	NUMBER(6)
PAYMENTAMOUNT	NUMBER(12)
PAYMENTDATE	DATE
USR	VARCHAR2(10)

9)Credittypes :

Name	Type
TYPEID	NUMBER(2)
TYPE	VARCHAR2(20)
RATIO	NUMBER(2)
MAXINSNUMBER	NUMBER(2)
USR	VARCHAR2(10)

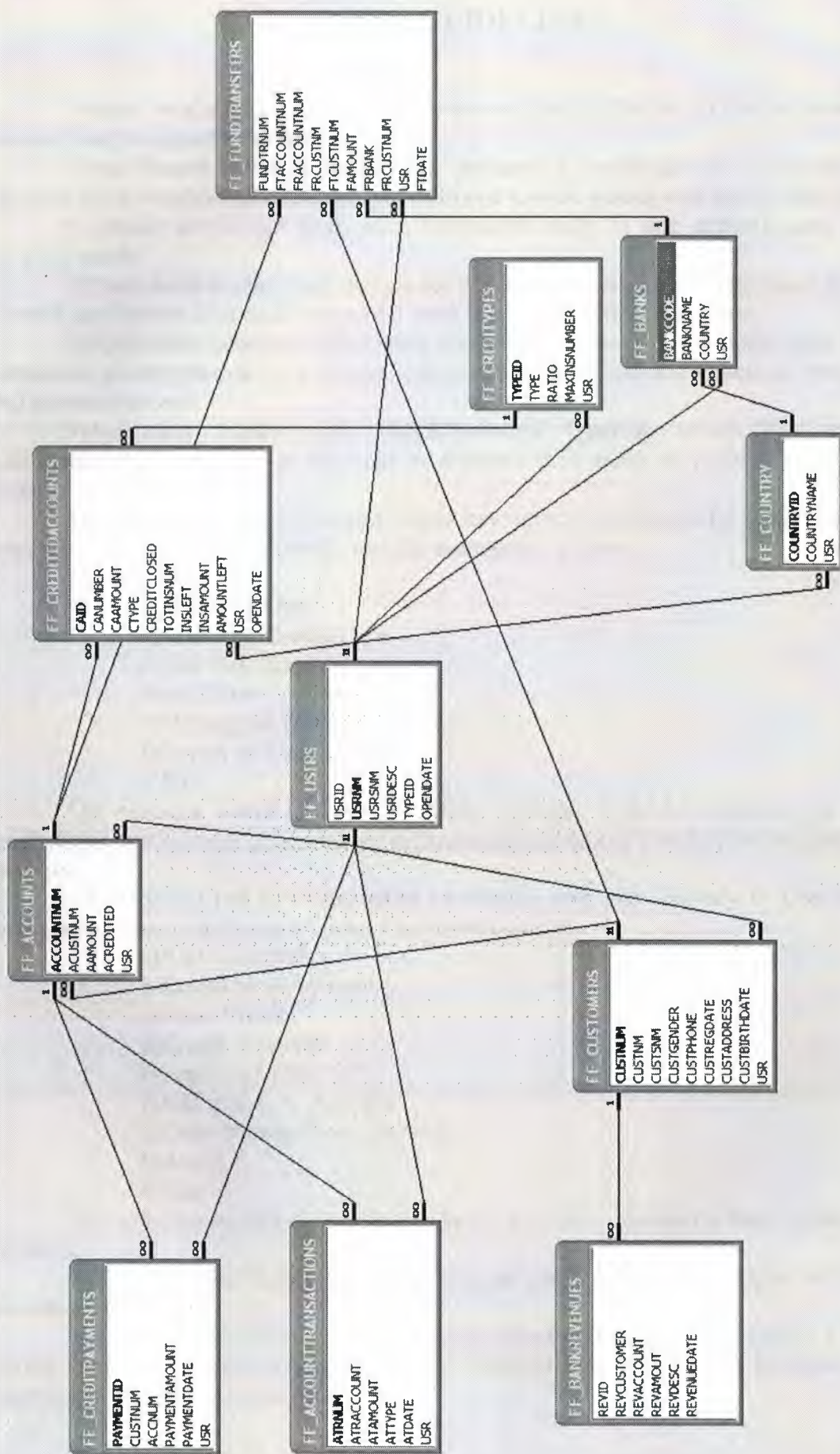
10)Fundtransfers :

Name	Type
FUNDTRNUM	NUMBER(10)
FTACCOUNTNUM	NUMBER(6)
FRACCOUNTNUM	NUMBER(6)
FRCUSTNM	VARCHAR2(20)
FTCUSTNUM	NUMBER(11)
FAMOUNT	NUMBER(12)
FRBANK	NUMBER(12)
FRCUSTNUM	NUMBER(11)
USR	VARCHAR2(10)
FTDATE	DATE

11)Users :

Name	Type
USRID	NUMBER(10)
USRNM	VARCHAR2(10)
USRSNM	VARCHAR2(10)
USRDESC	VARCHAR2(15)
TYPEID	VARCHAR2(10)
OPENDATE	DATE

Relationships



USER GUIDELINES

System works with Oracle server to storing data. So first of all Oracle server must be installed and managed by system engineers.

When Oracle server installed and prepared to available for application, network personnel has to connect all client computers in one Lan for getting and setting data to server.

If Oracle server and application computers ready in one network, users could use program easily.

1) Users have to click only RB.jar for running program. There is no setup is necessary to work application. Java application will work instantly on client computer.

2) Application connects with Oracle server when it is necessary, other times application terminates the connection for avoiding both network traffic jam and Oracle server processing and connection caos.

3) Application comes with User&Password Entering screen. That screen gets user&password from user or manager to sepearate their ranks to preparing their different menus.

4) If user is manager application ceates new screen for Managerial Menu that is serving managers the permissions to reach spesific managerial screens.

Those are:

- a) Add New User
- b) Add New Credit Type
- c) Add New Bank
- d) Add New Country
- e) Managerial Reports
- f) Switch to User Menu
- g) Exit

5) If Manager wants to see User Menu manager could be swith on to this menu easily. Other menu items gets manager to do managerial works that they are explained in item captions.

6) If user has not had managerial permission user goes directly to User Menu. This menu serves user to doing user's jobs. This menu contains:

- a) Add Customer
- b) Create New Account
- c) Open Credit
- d) Credit Payments
- e) Fund Transfers
- f) Add money to Account
- g) Draw Money From Account
- h) About
- i) Exit

7) If user choose Add Customer, user has to give new customer's datas on this screen to record it.

8) If user choose Create New Account, user could create new account with choosing recorded customer.

9) If user choose Open Credit, user could choose customer and customer's account to giving credit. If customer account has been credited user will be warn by application and crediting is impossible to account.

10)If user choose Credit Payments,user could choos customer and customer's account which is credited before and ready to payment.If customer or account has not been credited system will warn user with customer has not credited or account has not credited.

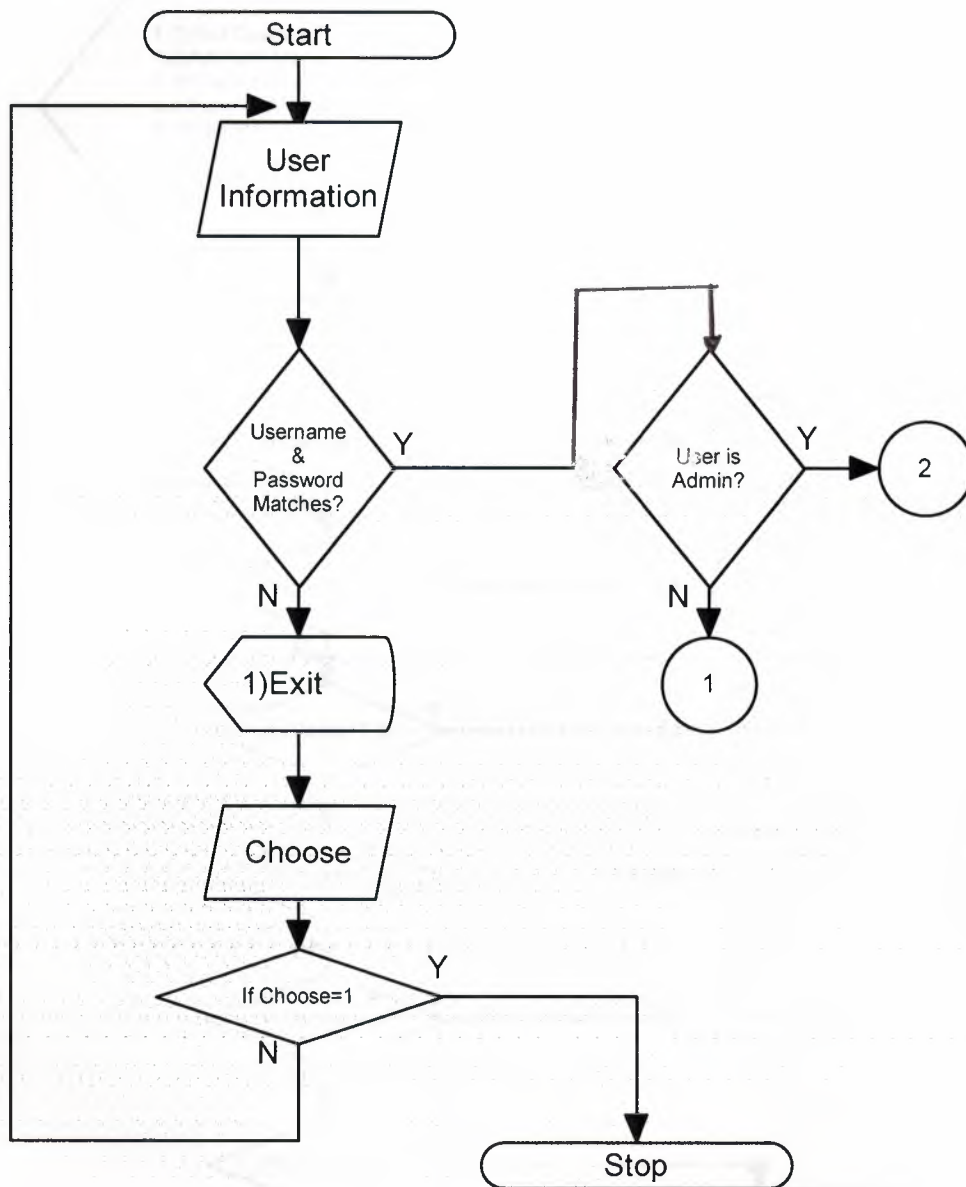
11)If user choose Fund Transfers,user could choose transiver customer,customer's account to transfer and user could sent money to another customer name,another customer account,transiver customers othe account or foreign bank customer.If user choose foreing bank customer user has to give foreign bank name and receiver foreign bank customer Identity number and name,surname.

12)Add money to account and Draw Money From account is easy screens to adding or receiving money from customer accounts.User could choose customer and it's account for those transactions.

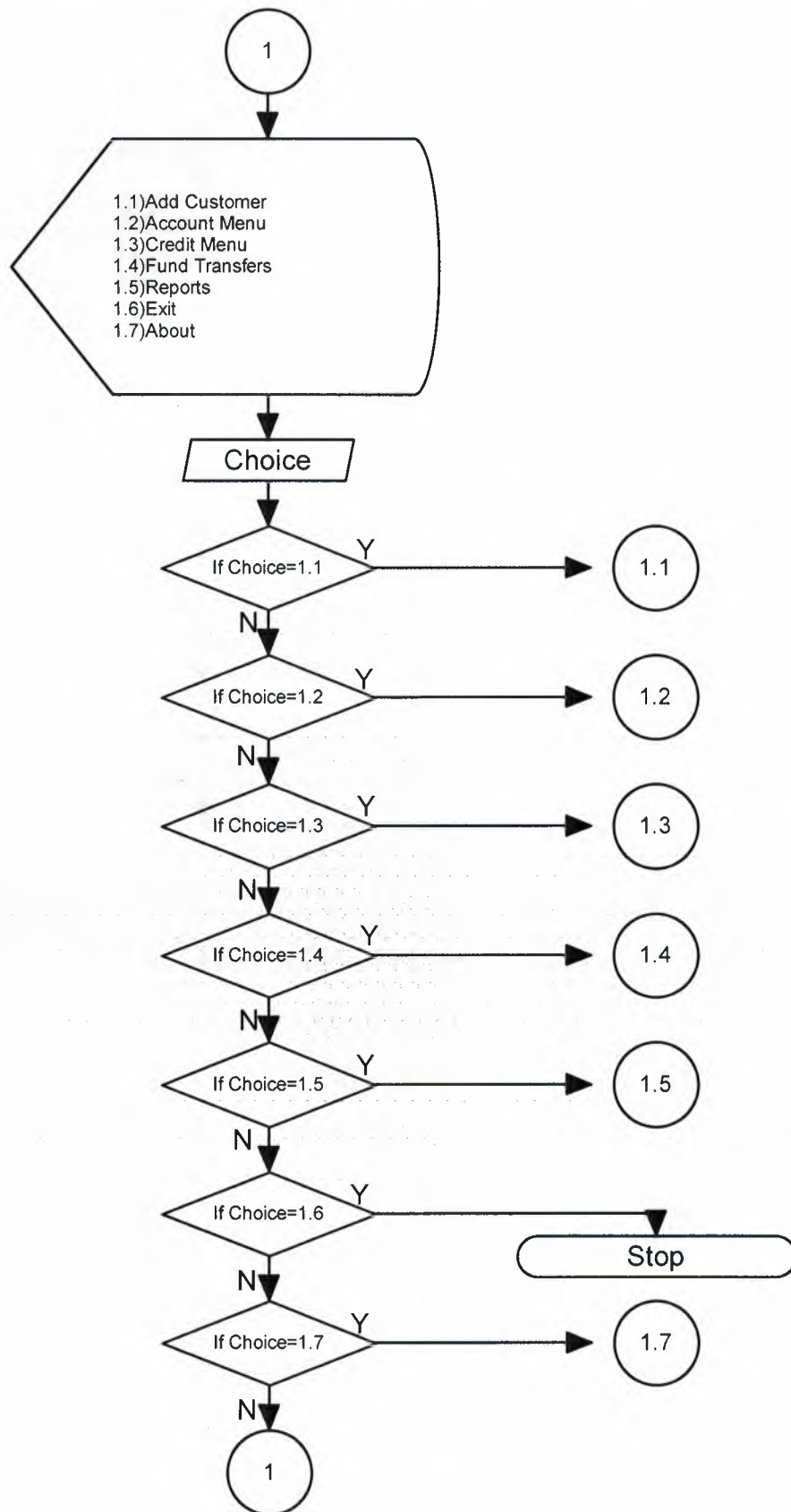
13)About shows the System's analiser,designer and programmers communication details.

USER FLOWCHARTS

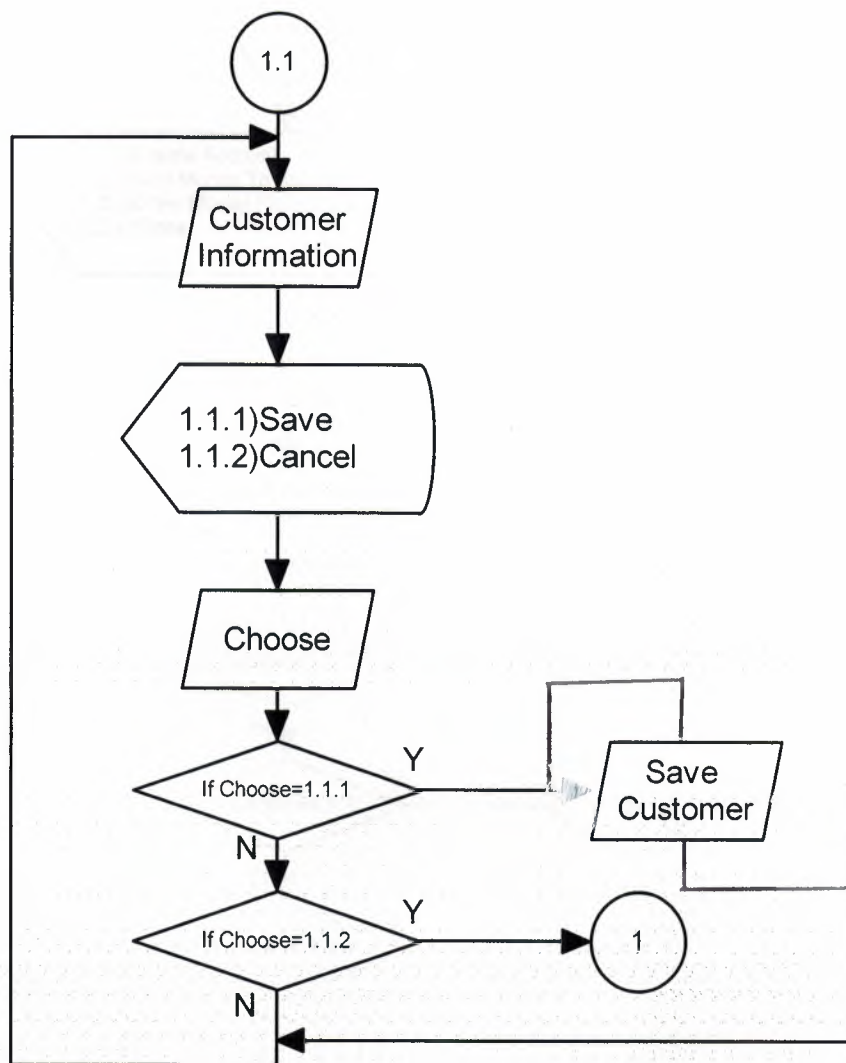
User & Password Flowchart For EMFA Bank Retail Banking System



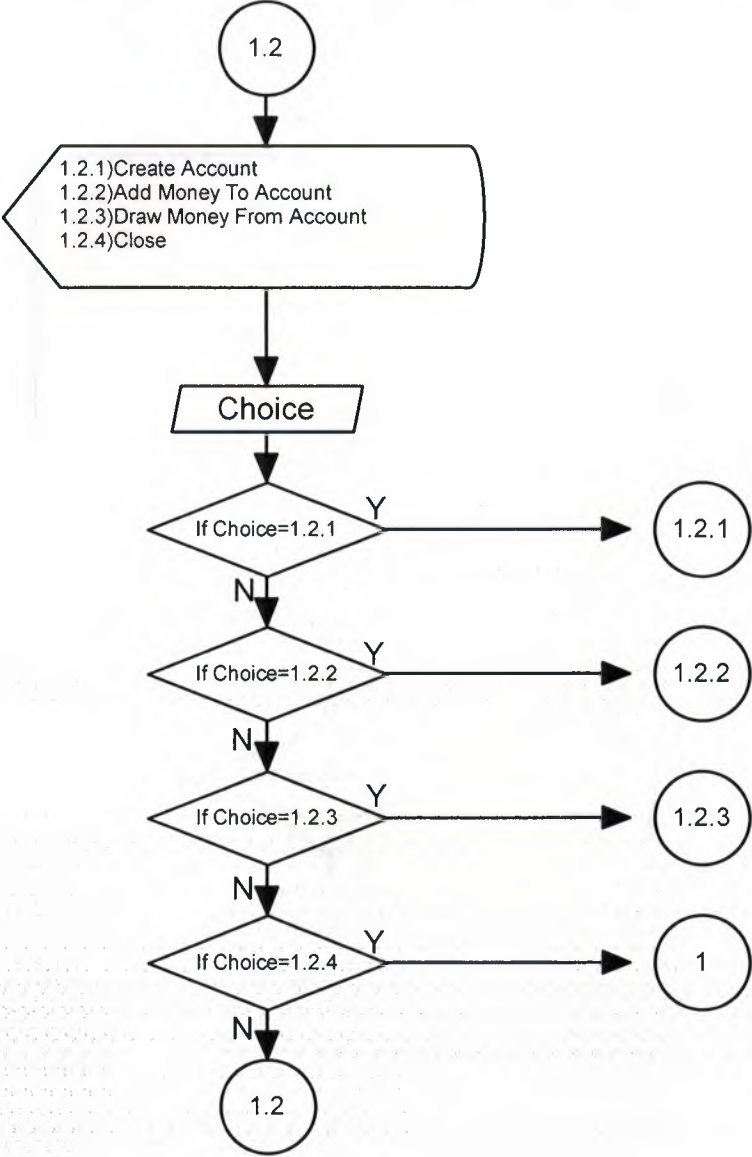
User Menu Flowchart For EMFA Bank Retail Banking System



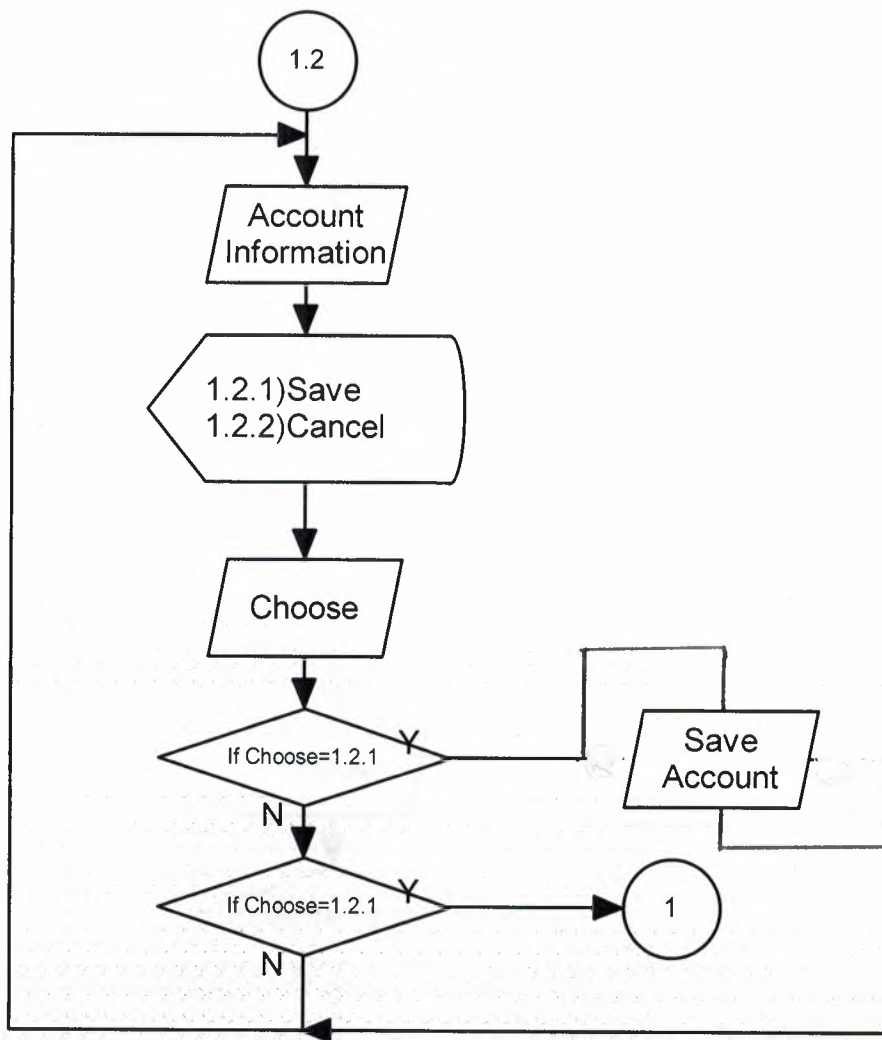
Add New Customer Flowchart For EMFA Bank Retail Banking System



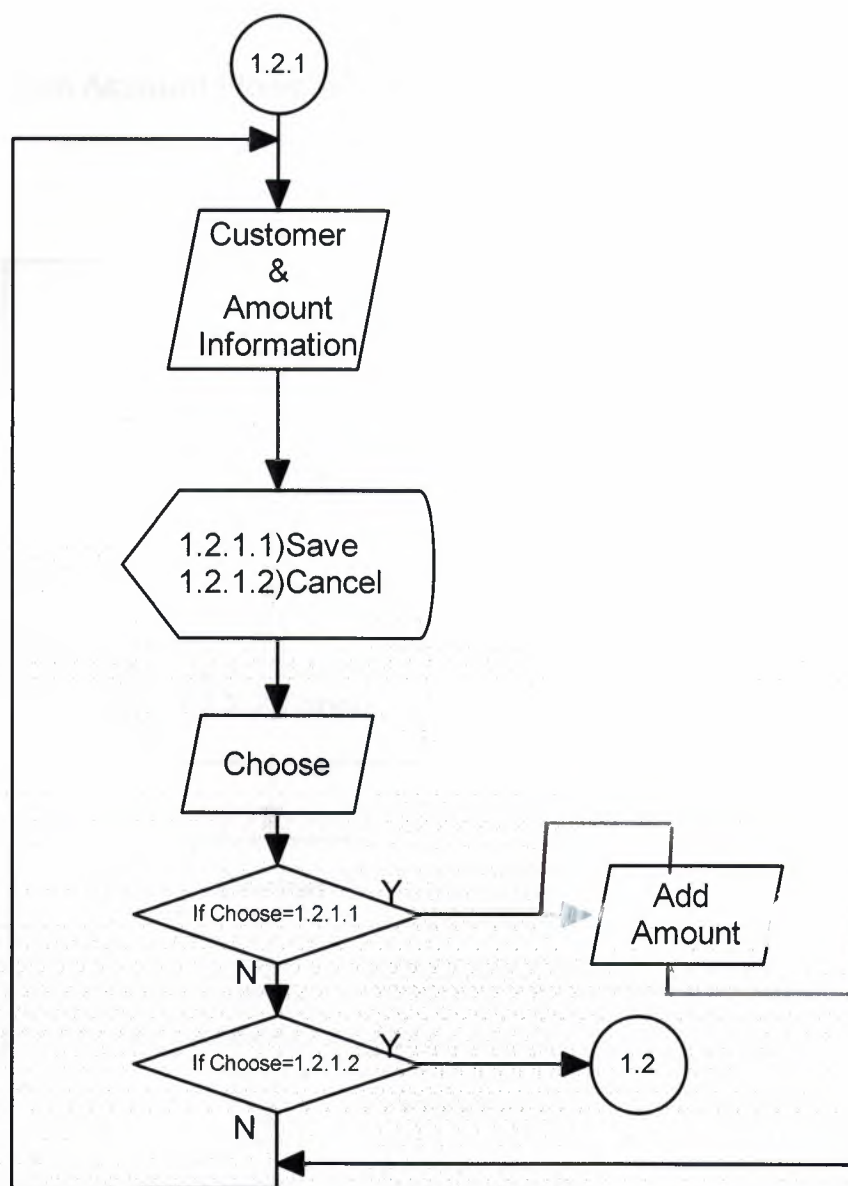
Account Menu Flowchart For EMFA Bank Retail Banking System



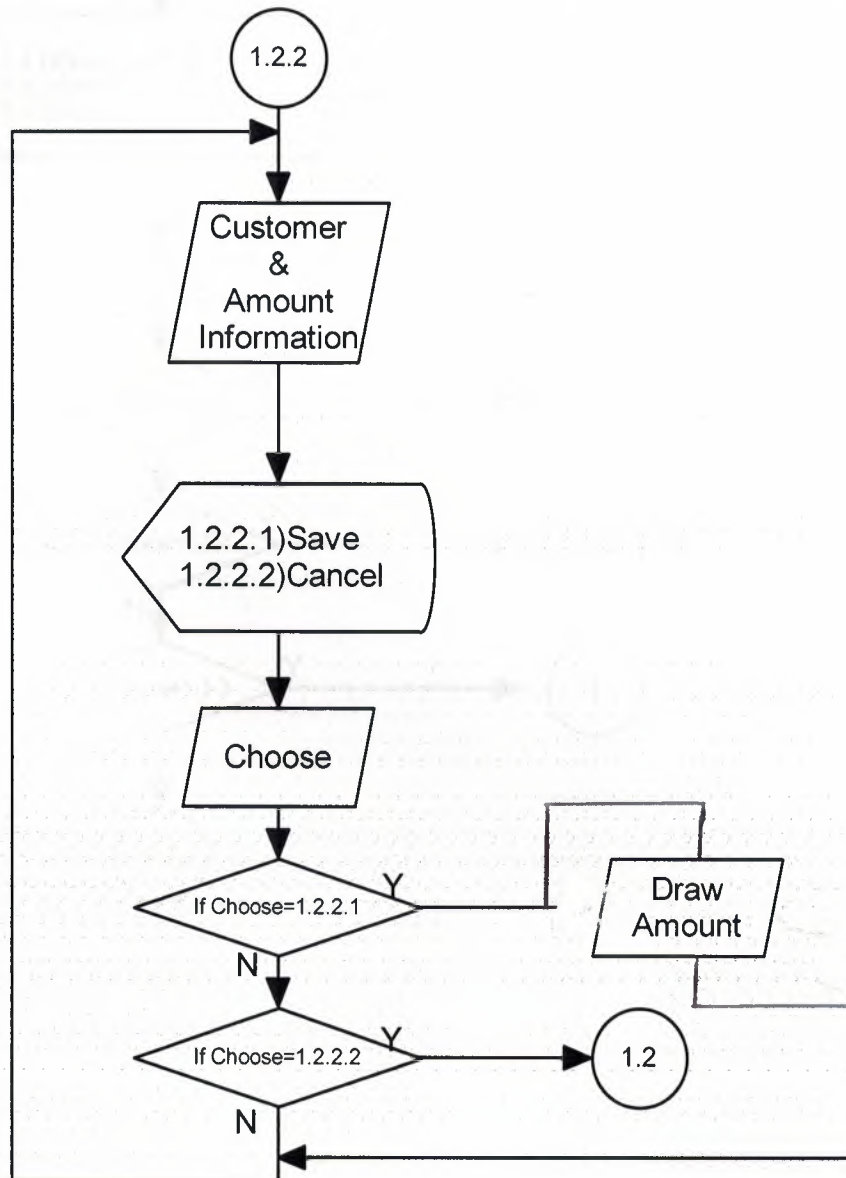
Create New Account Flowchart For EMFA Bank Retail Banking System



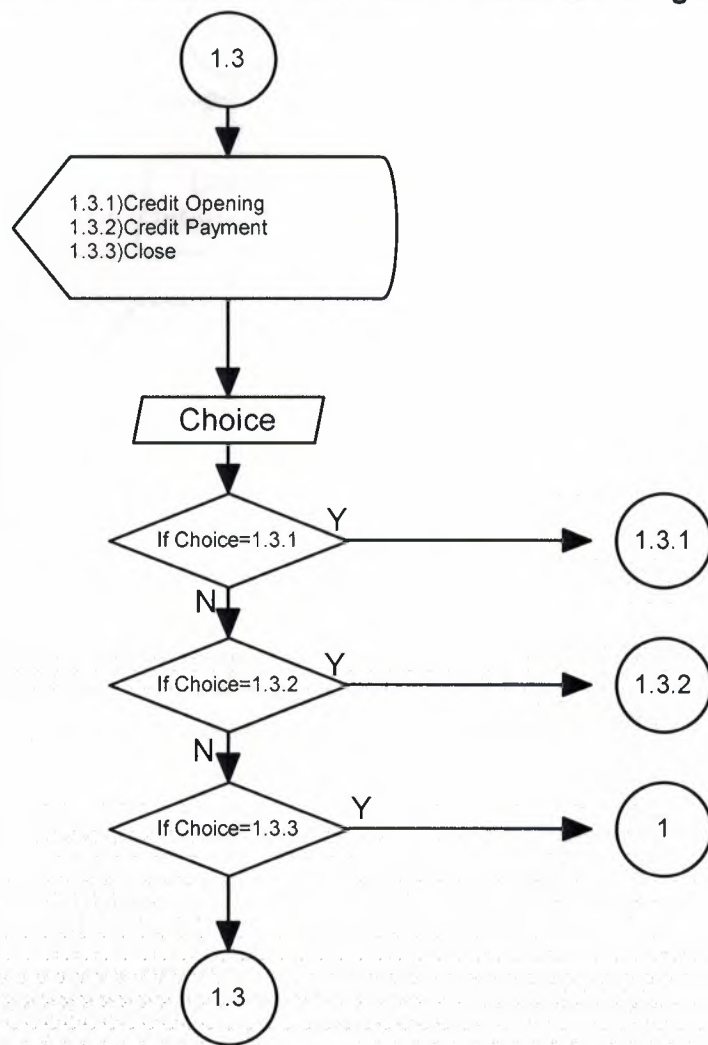
Add Money To Account Flowchart For EMFA Bank Retail Banking System



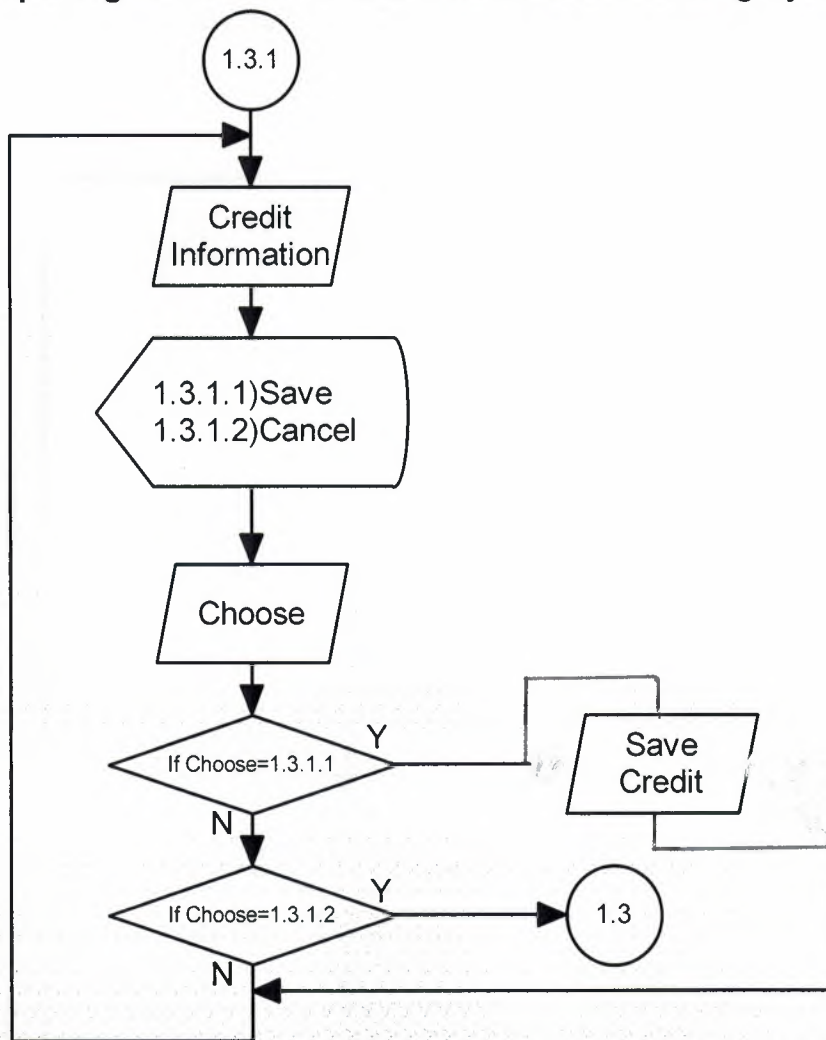
Draw Money From Account Flowchart For EMFA Bank Retail Banking System



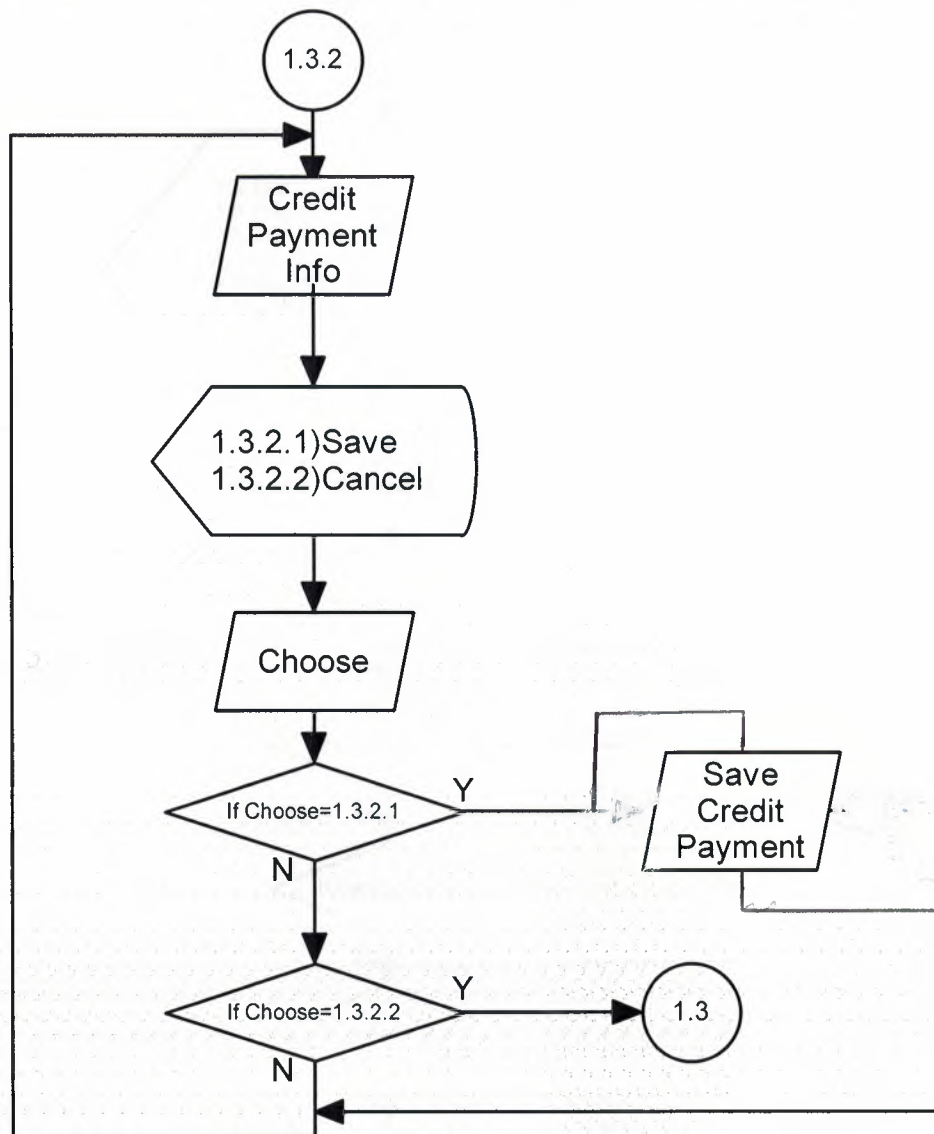
Credit Menu Flowchart For EMFA Bank Retail Banking System



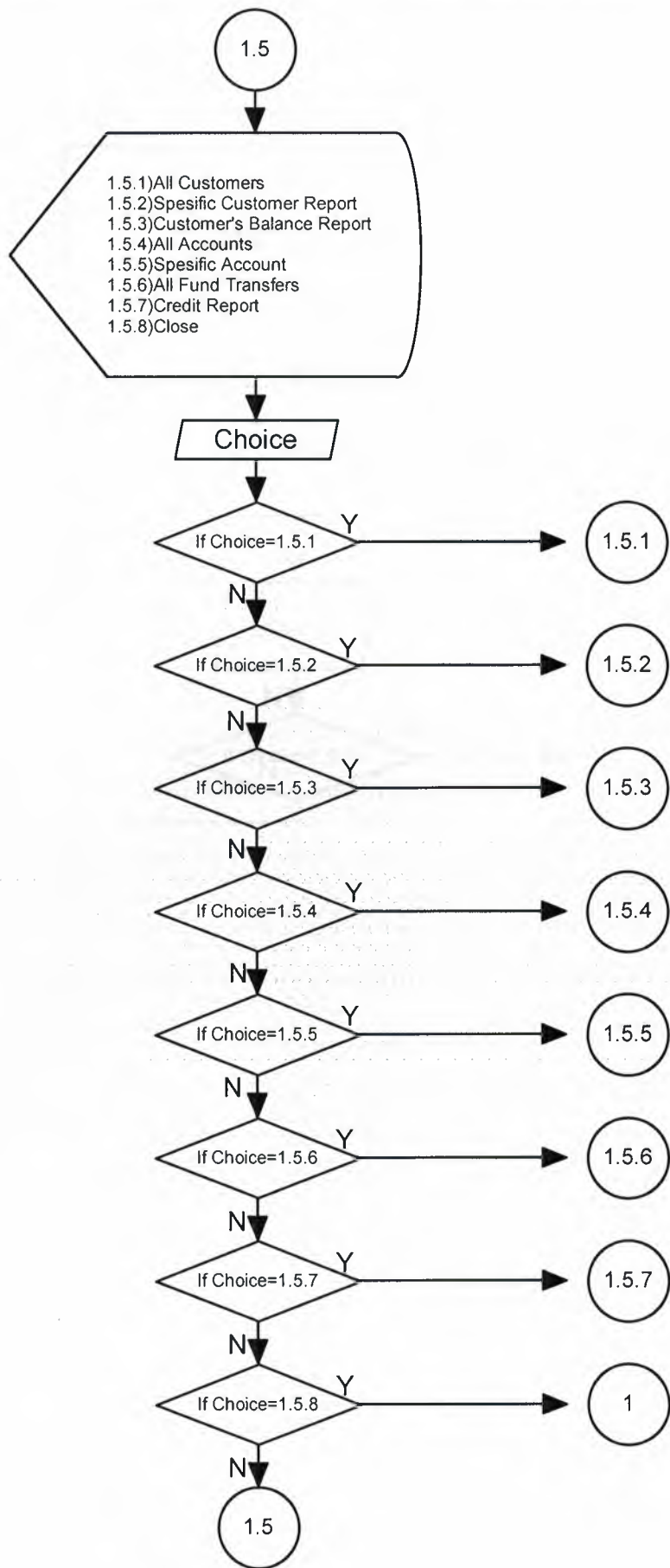
Credit Opening Flowchart For EMFA Bank Retail Banking System



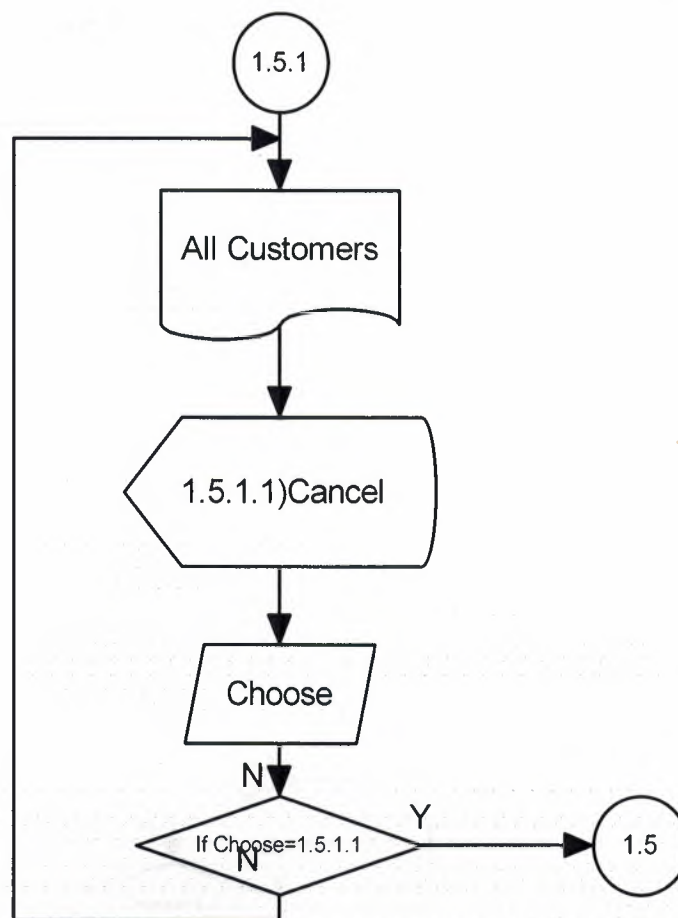
Credit Payment Flowchart For EMFA Bank Retail Banking System



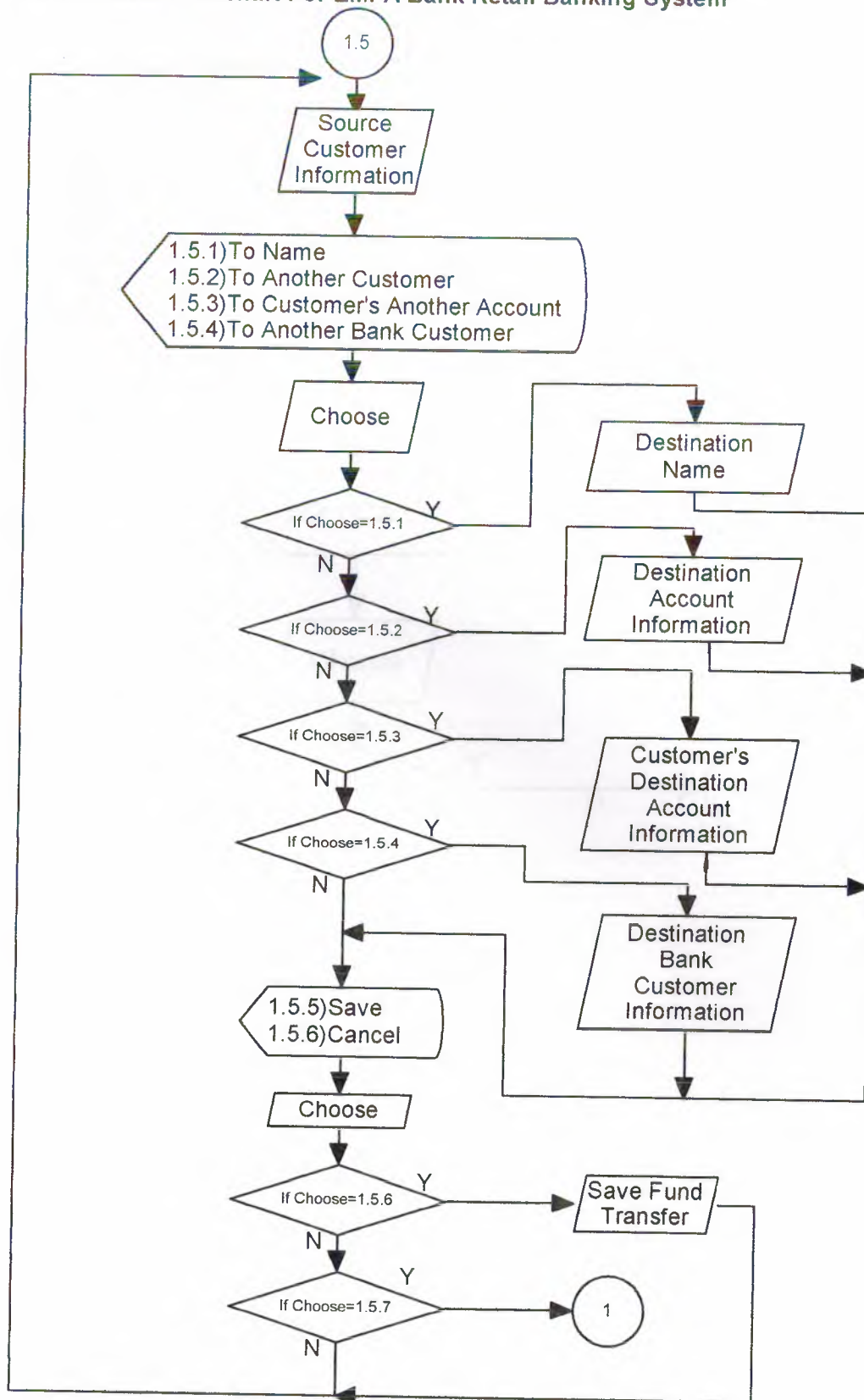
User Reports Menu Flowchart For EMFA Bank Retail Banking System



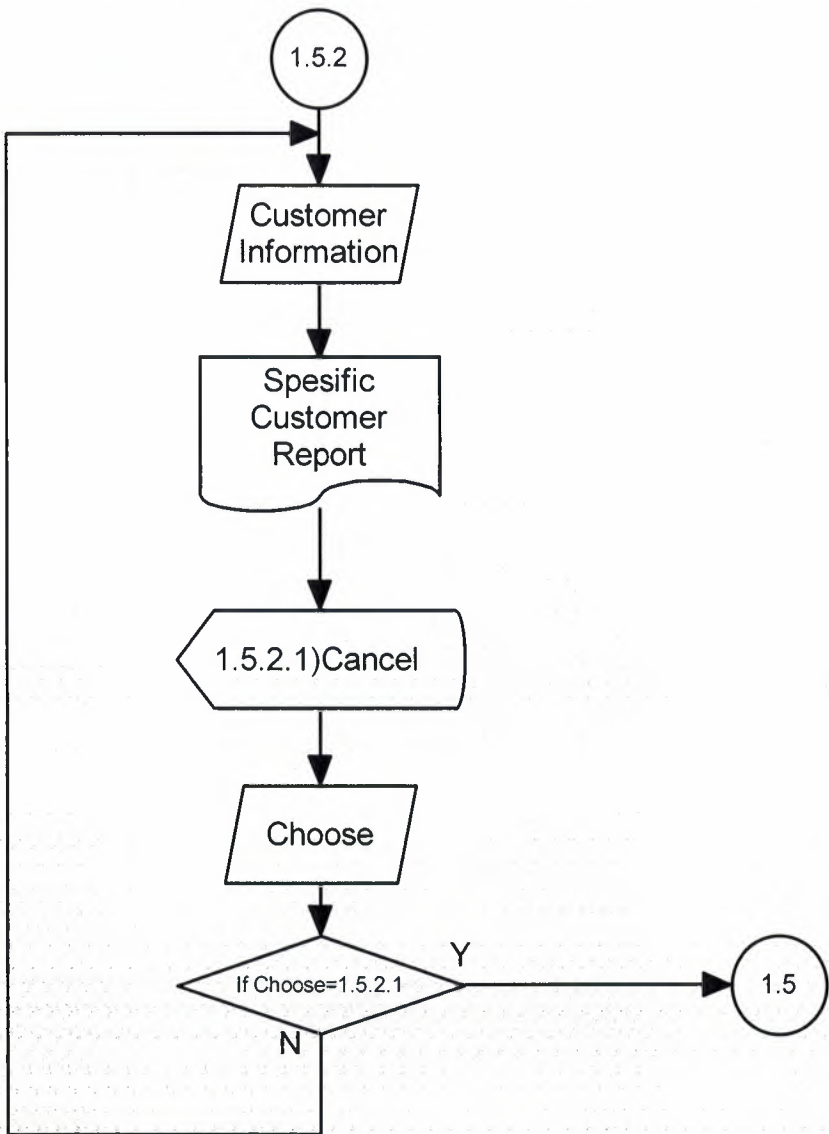
All Customers Report Flowchart For EMFA Bank Retail Banking System



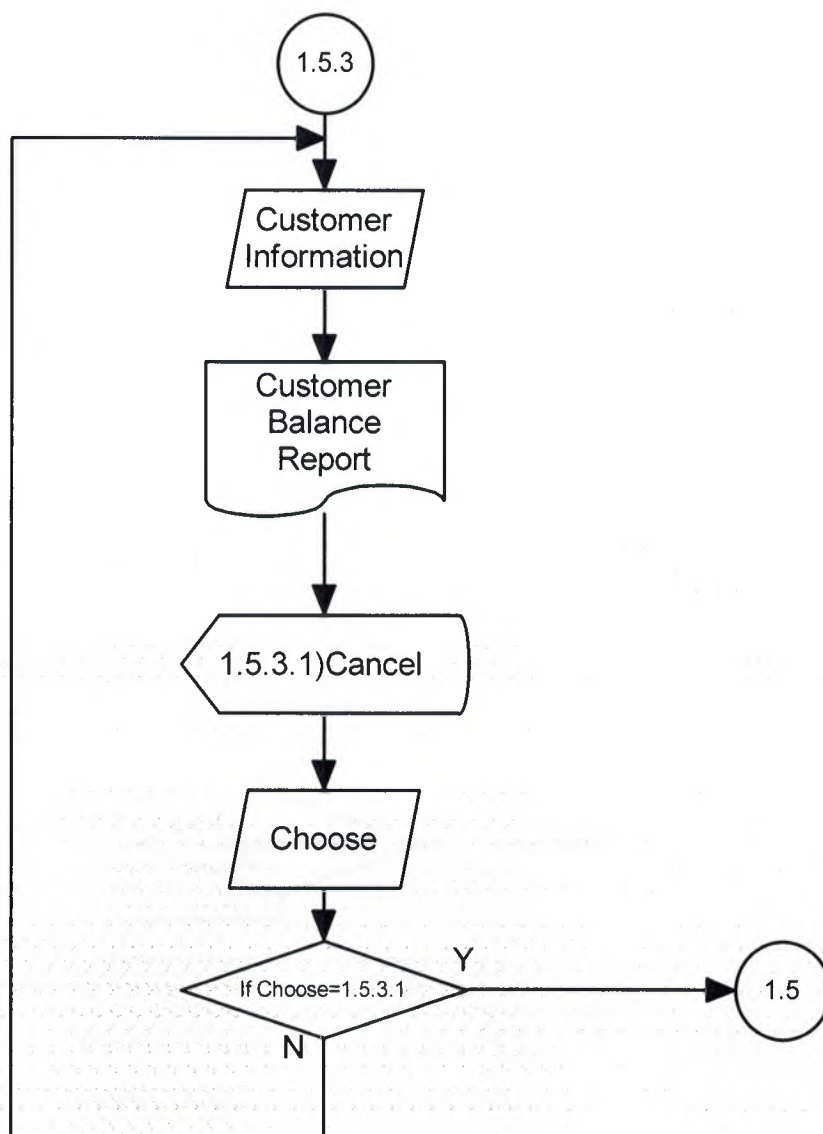
Fund Transfers Flowchart For EMFA Bank Retail Banking System



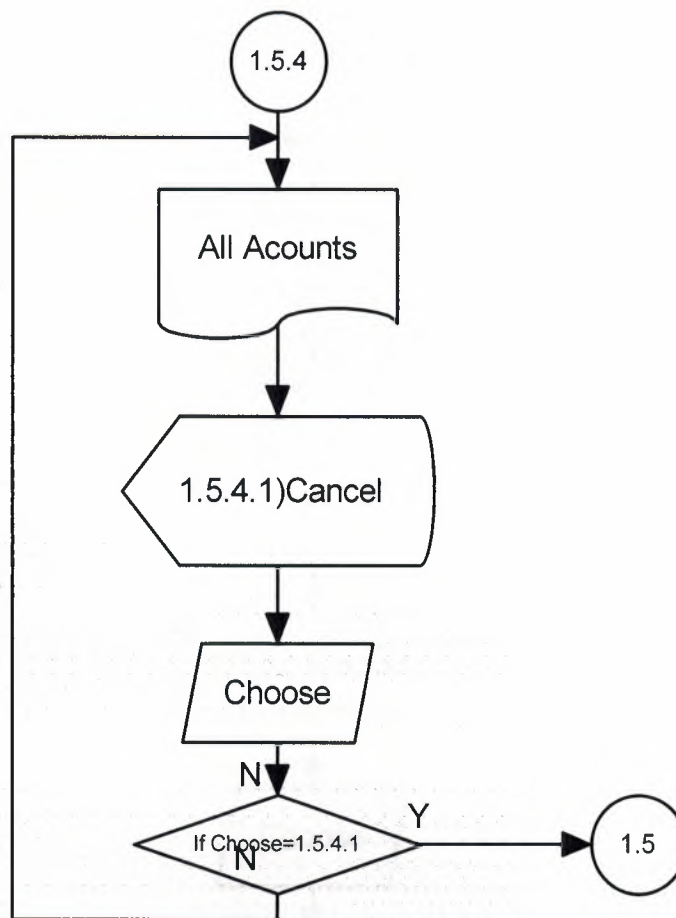
Spesific Customer Report Flowchart For EMFA Bank Retail Banking System



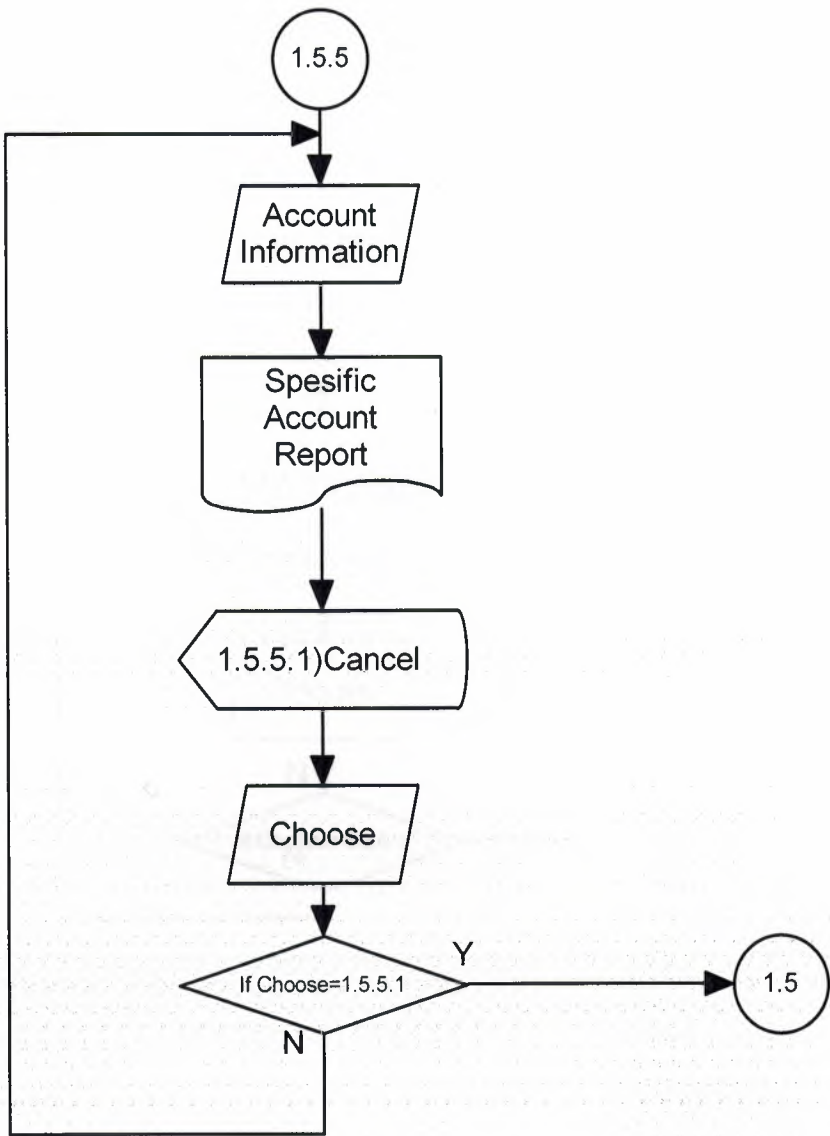
Customer Balance Report Flowchart For EMFA Bank Retail Banking System



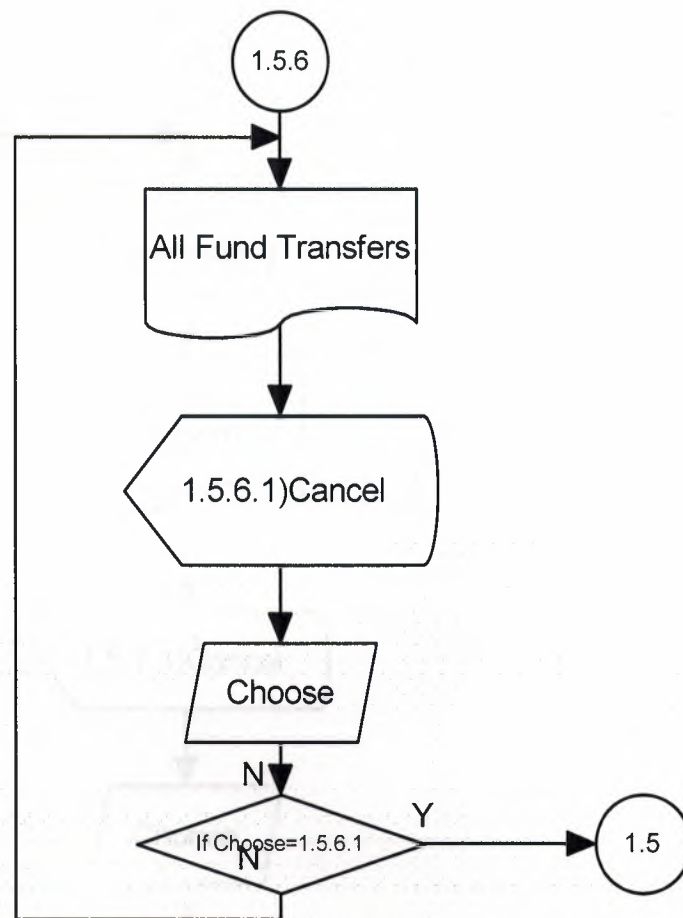
All Accounts Report Flowchart For EMFA Bank Retail Banking System



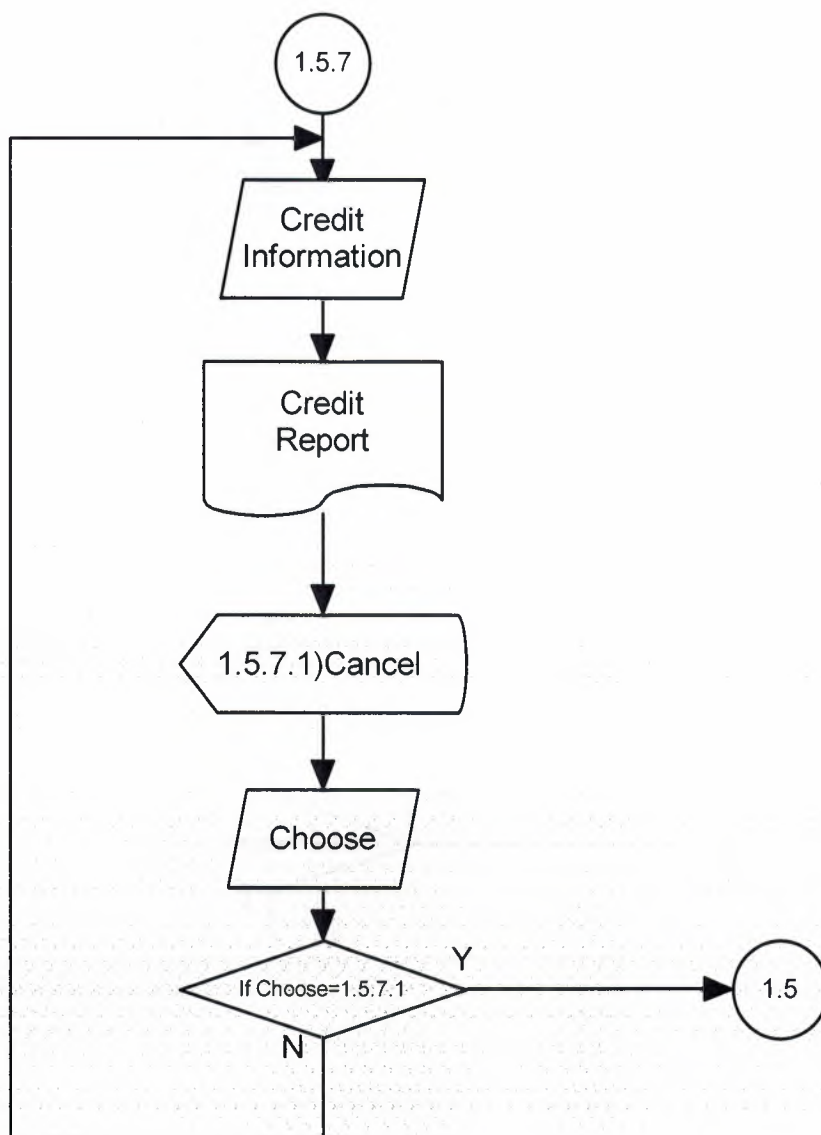
Spesific Account Report Flowchart For EMFA Bank Retail Banking System



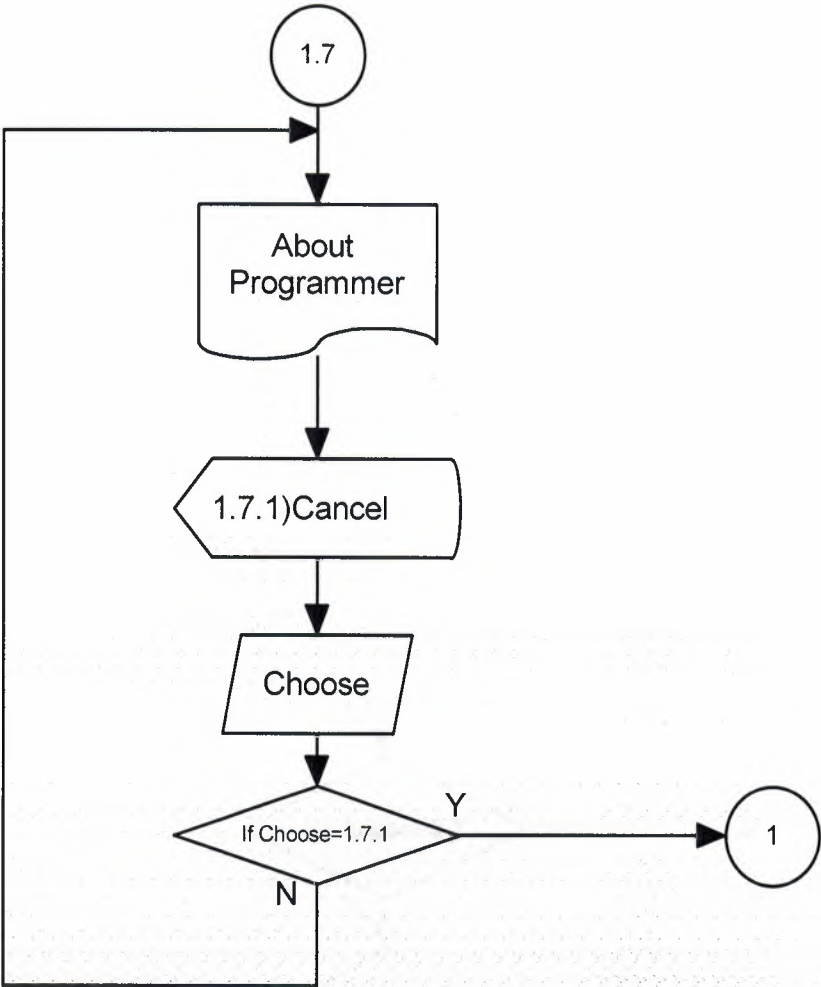
All Fund Transfers Report Flowchart For EMFA Bank Retail Banking System



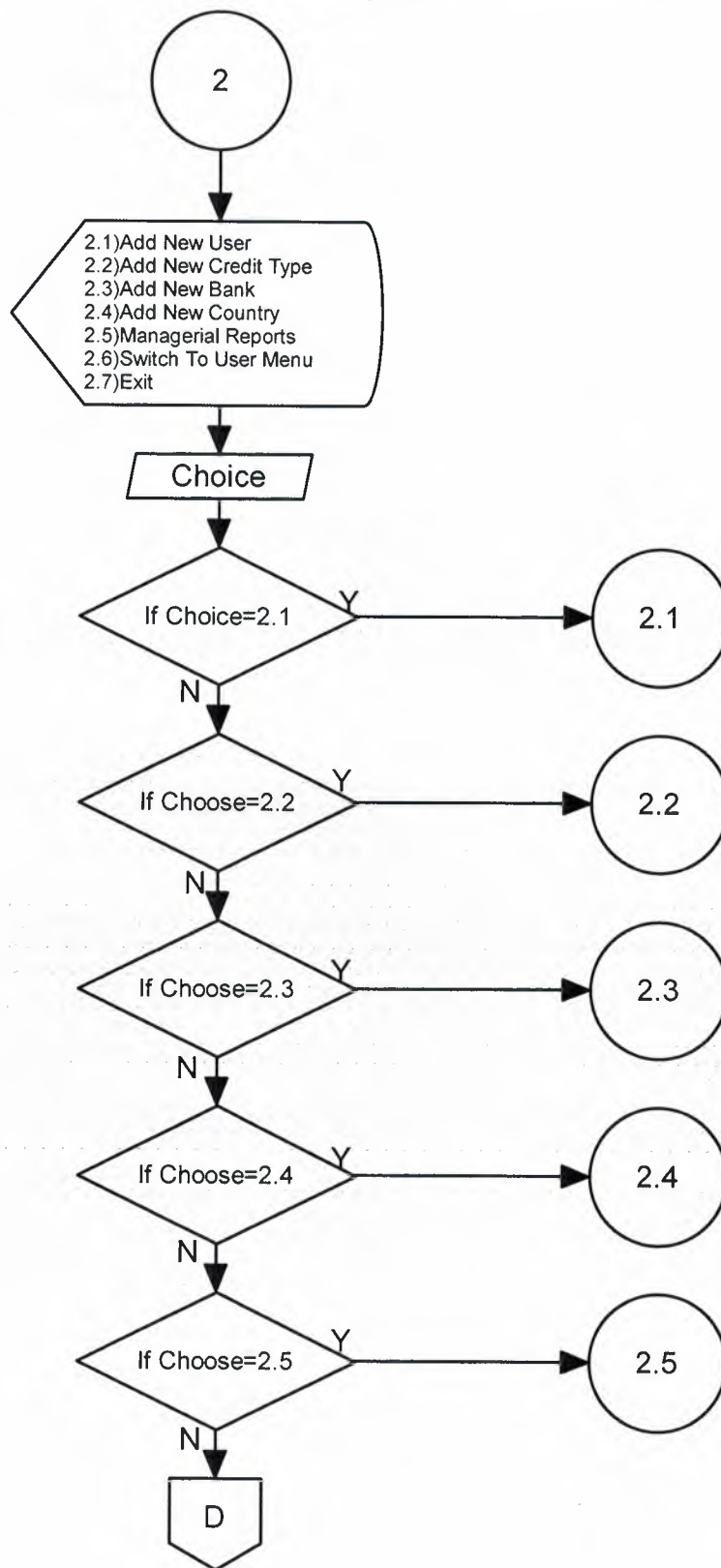
Credit Report Flowchart For EMFA Bank Retail Banking System

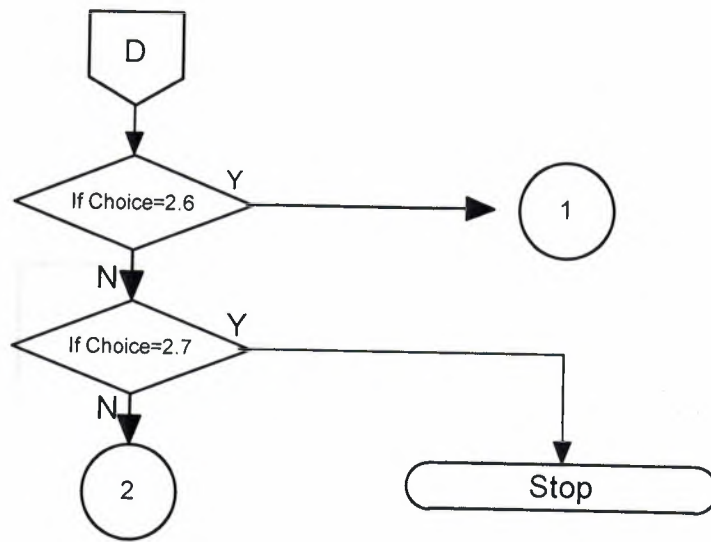


About Flowchart For EMFA Bank Retail Banking System

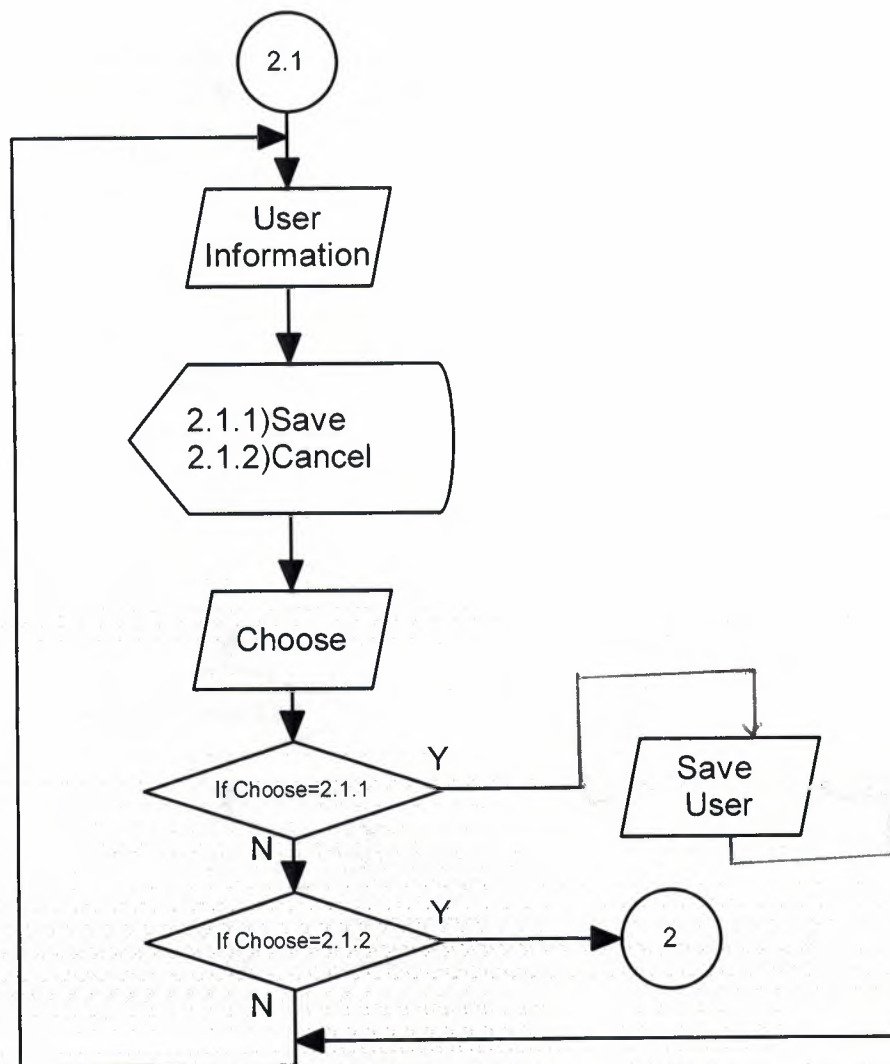


Managerial Menu Flowchart For EMFA Bank Retail Banking System

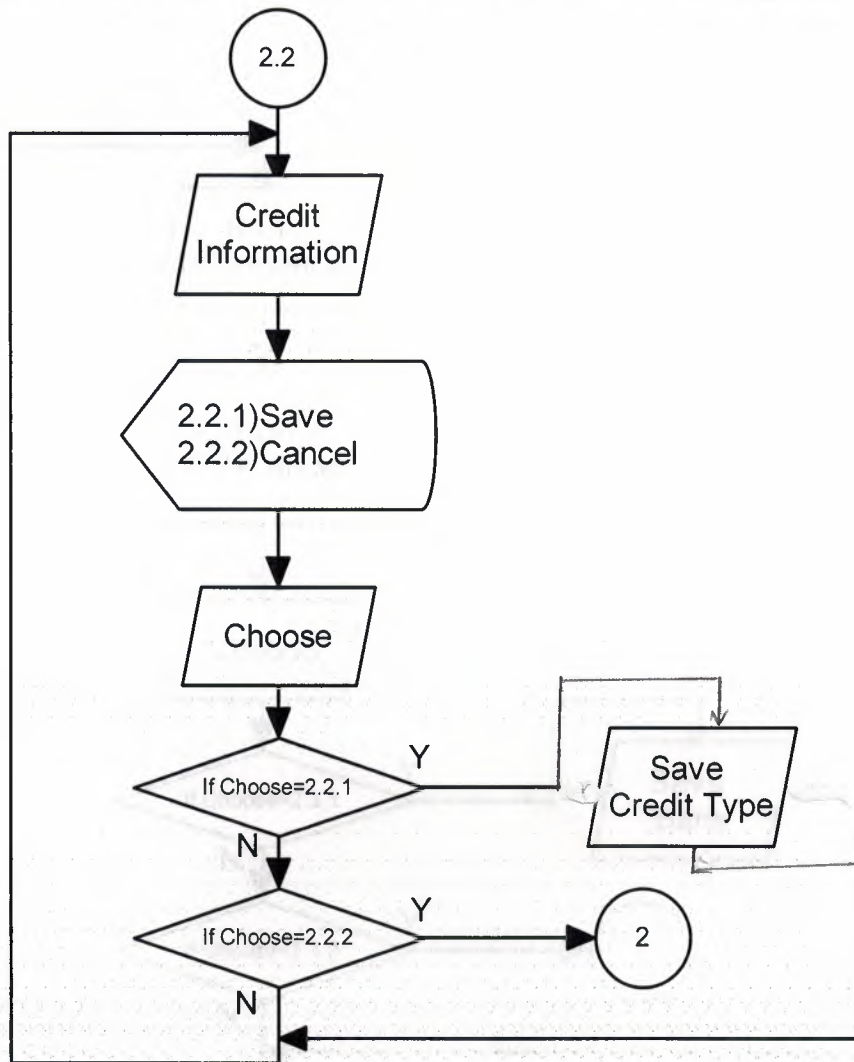




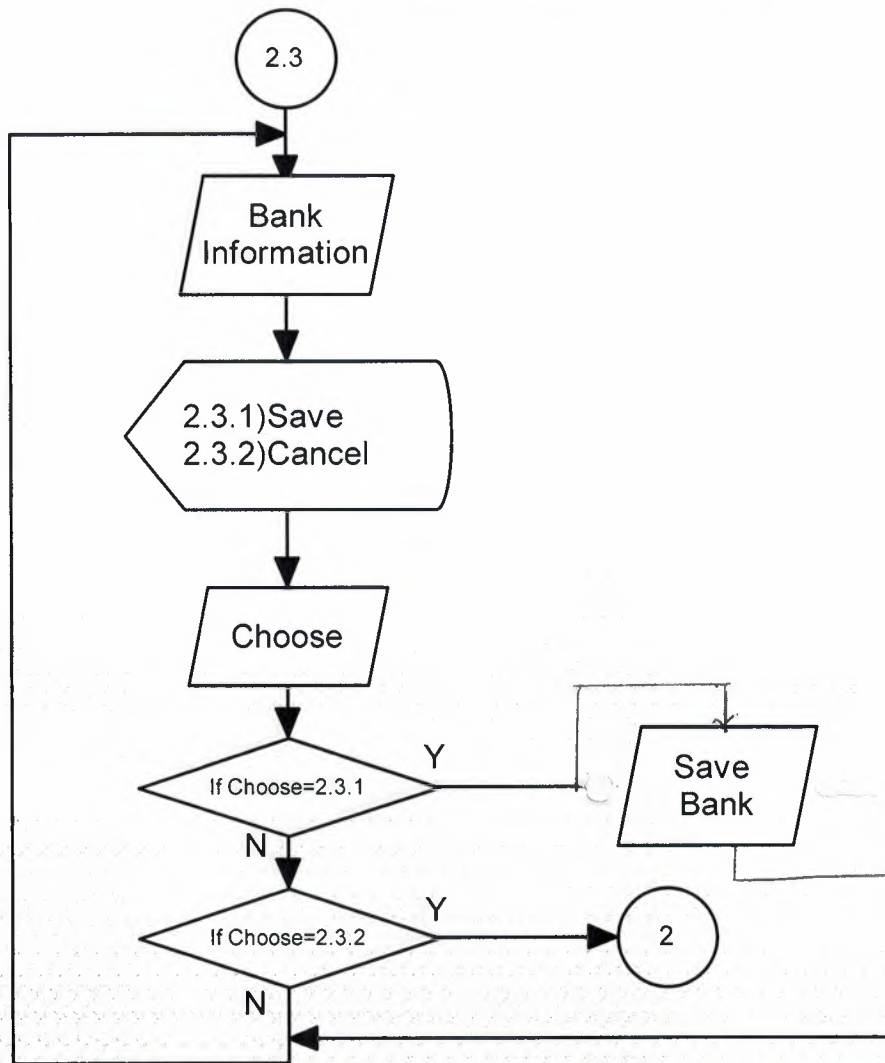
Add New User Flowchart For EMFA Bank Retail Banking System



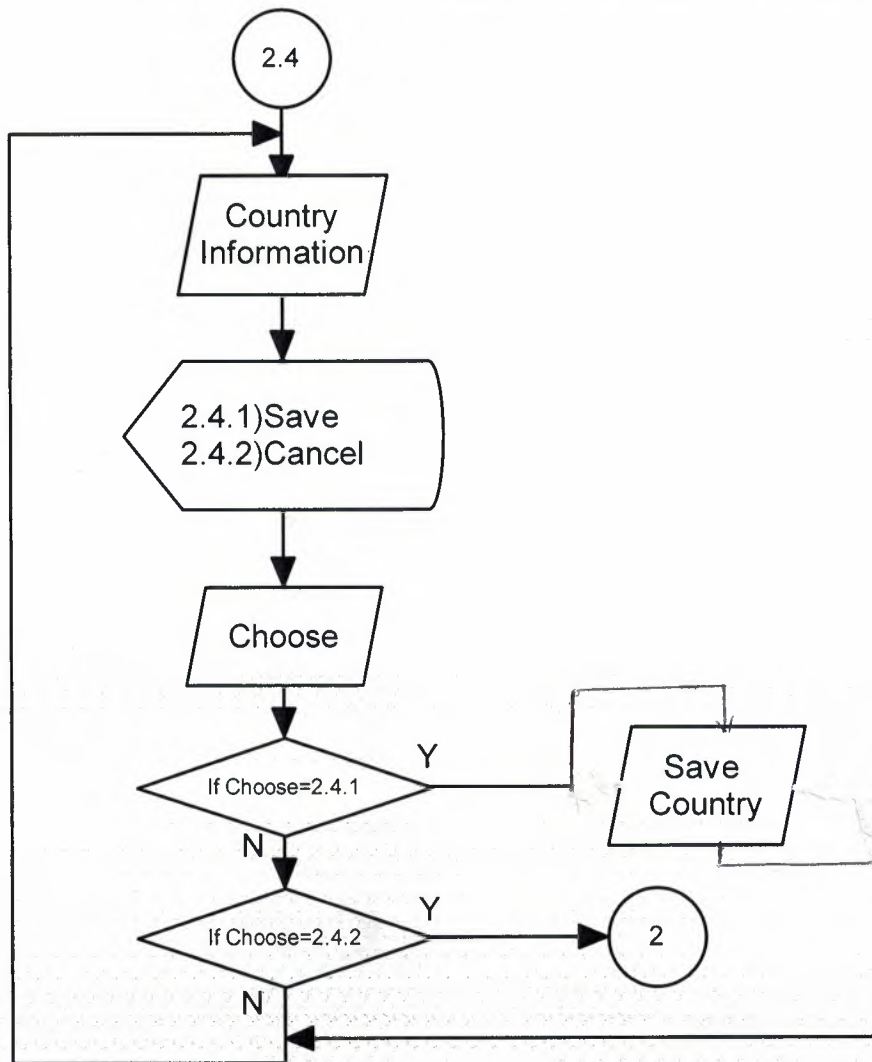
Add New Credit Type Flowchart For EMFA Bank Retail Banking System



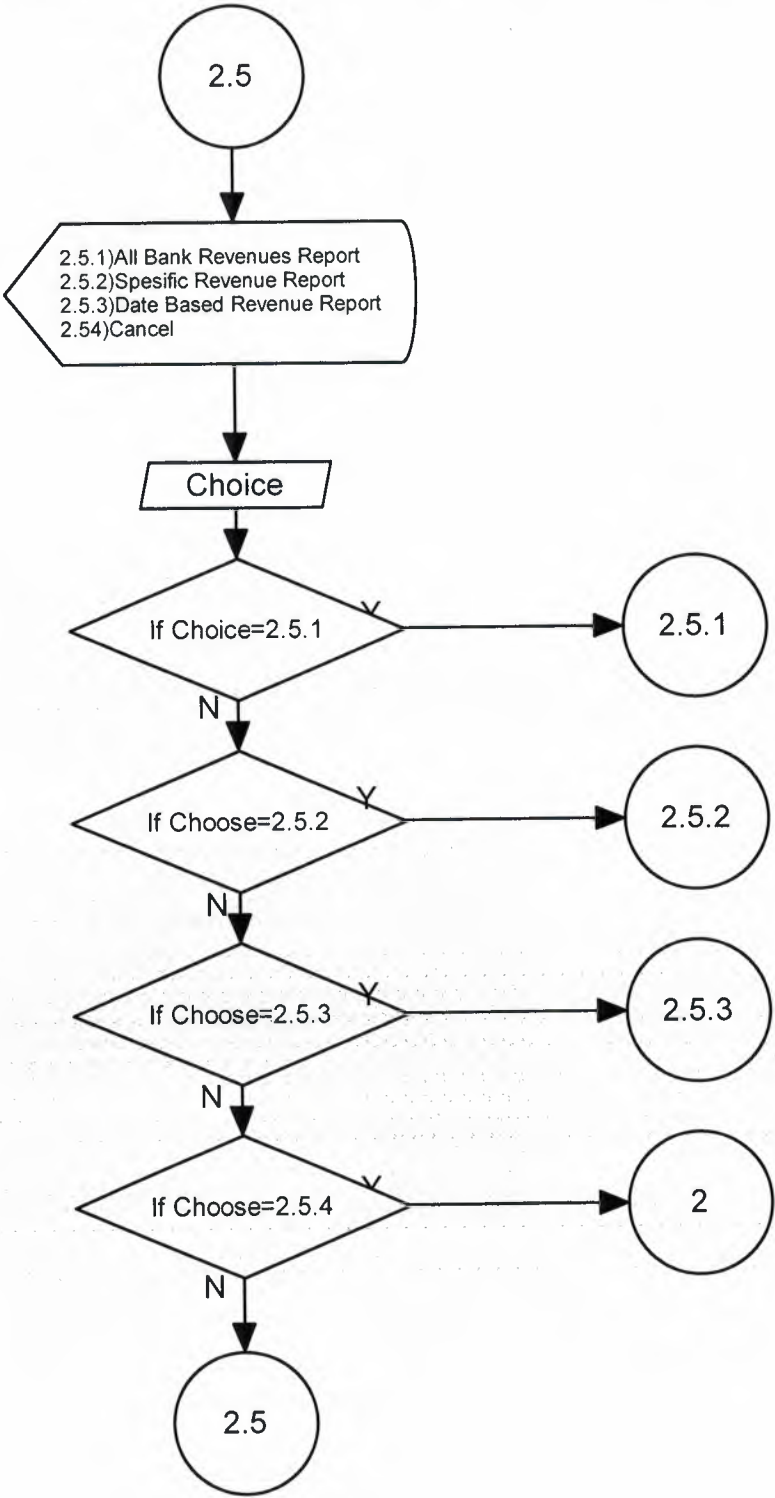
Add New Bank Flowchart For EMFA Bank Retail Banking System



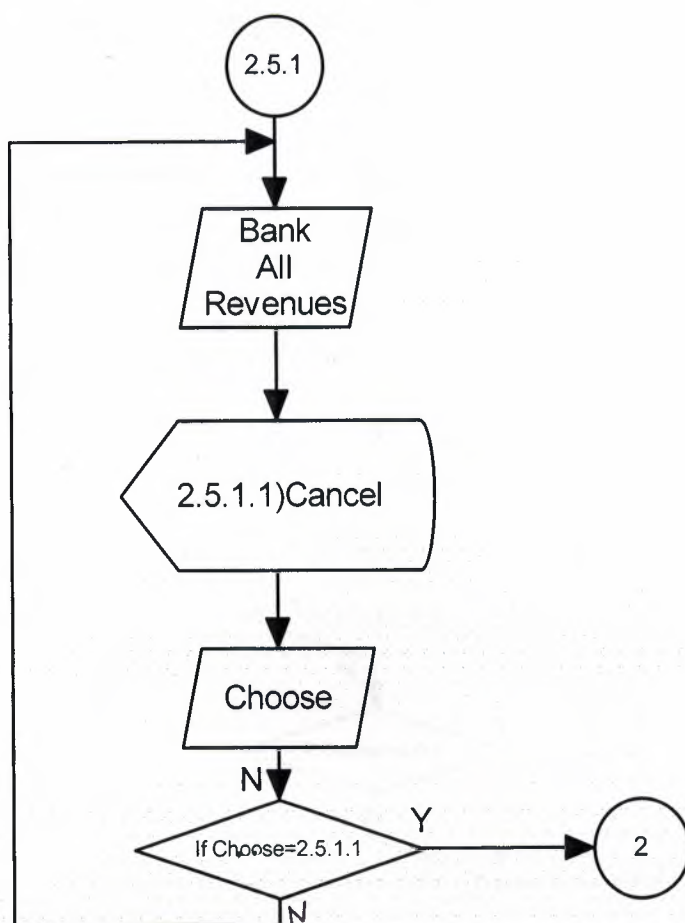
Add New Country Flowchart For EMFA Bank Retail Banking System



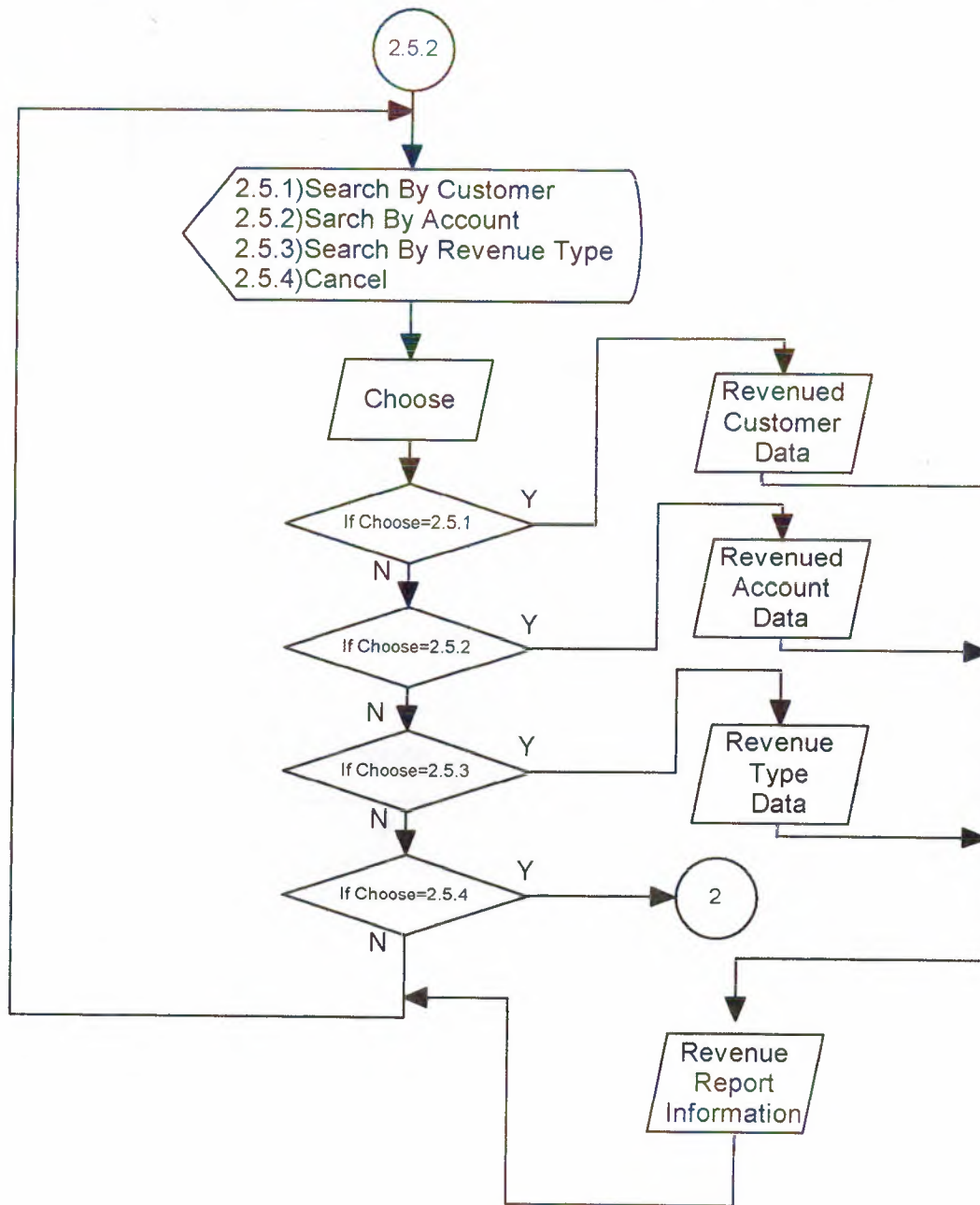
Managerial Reports Menu Flowchart For EMFA Bank Retail Banking System



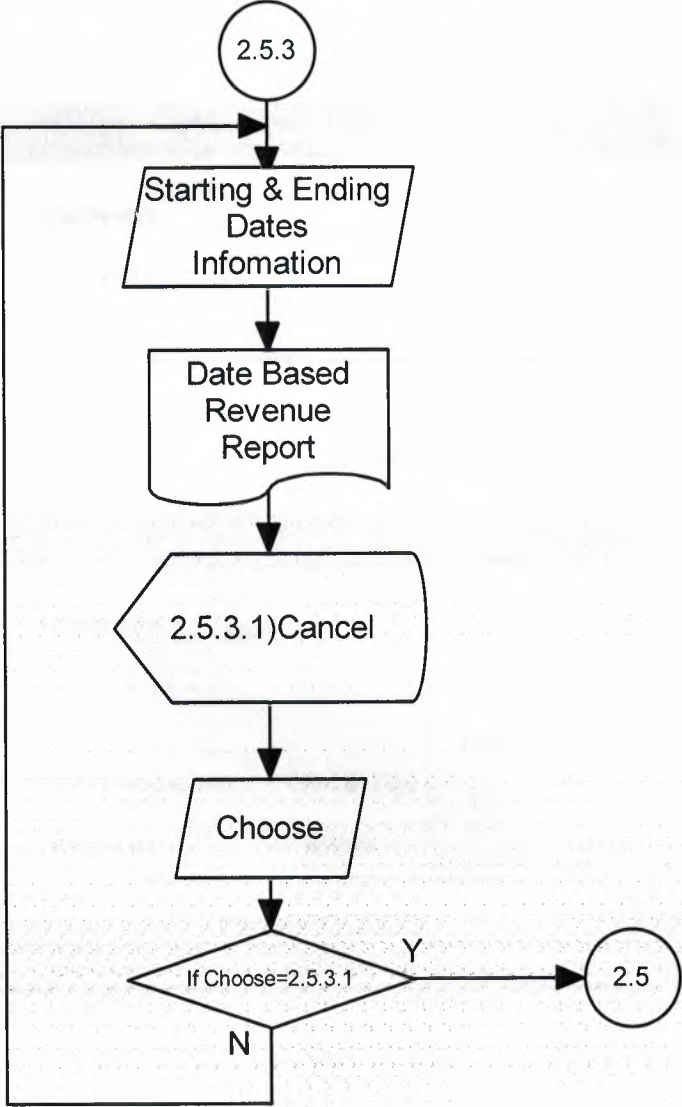
All Bank Revenues Flowchart For EMFA Bank Retail Banking System



Specific Bank Revenue Report Flowchart For EMFA Bank Retail Banking System



Date Based Revenue Report Flowchart For EMFA Bank Retail Banking System



APPENDIX I

Screen Ouputs

Figure 1:

A screenshot of a login window titled "Please Enter Your Username and Password". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. Inside the window, there are two input fields: "Username" and "Password". To the right of these fields are two buttons: "Connect" and "EXIT".

Figure 2:

A screenshot of a user profile window titled "Please Enter Your Username and Password". The window has a standard Windows-style title bar. Inside, the "Username" field is filled with "fatih". Below it, the "User Description" is "FATIH CAY" and the "User Type" is "ADMIN". There are three buttons: "Continue" (positioned to the right of the user details), "EXIT" (positioned to the right of the "Continue" button), and "Change Password" (positioned below the user details).

Figure 3:

A screenshot of a login window titled "Please Enter Your Username and Password". The window contains the following elements:

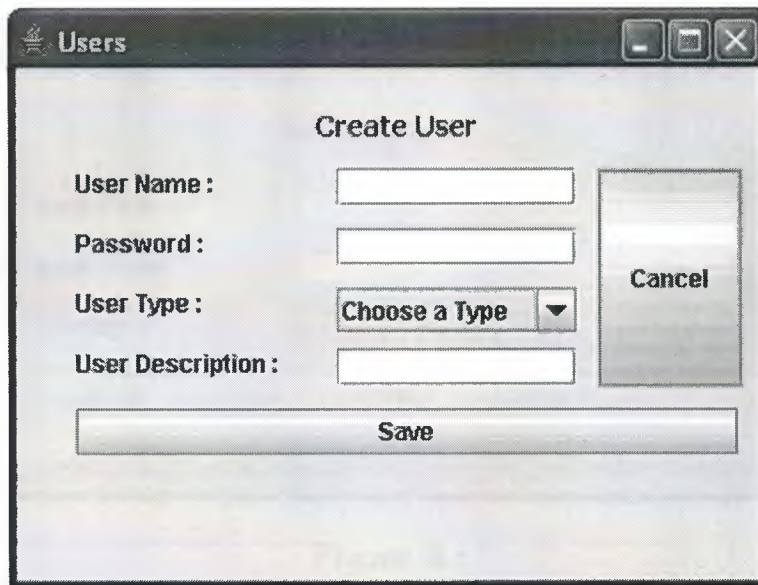
- Username:** A text input field containing the text "fatih".
- User Description:** A label "User Description :" followed by the text "FATIH CAY".
- User Type:** A label "User Type :" followed by the text "ADMIN".
- Buttons:** A "Continue" button, a "Change Password" button, and an "EXIT" button.
- Password Fields:** Two text input fields for "New Password" and "Re enter New Password", each preceded by a colon.
- Apply Changes:** A button at the bottom labeled "Apply Changes".

Figure 4 :

A screenshot of a "Main Menu" window. The window has a title bar with the text "Main Menu". The main content area is titled "Managerial Menu" and contains the following elements:

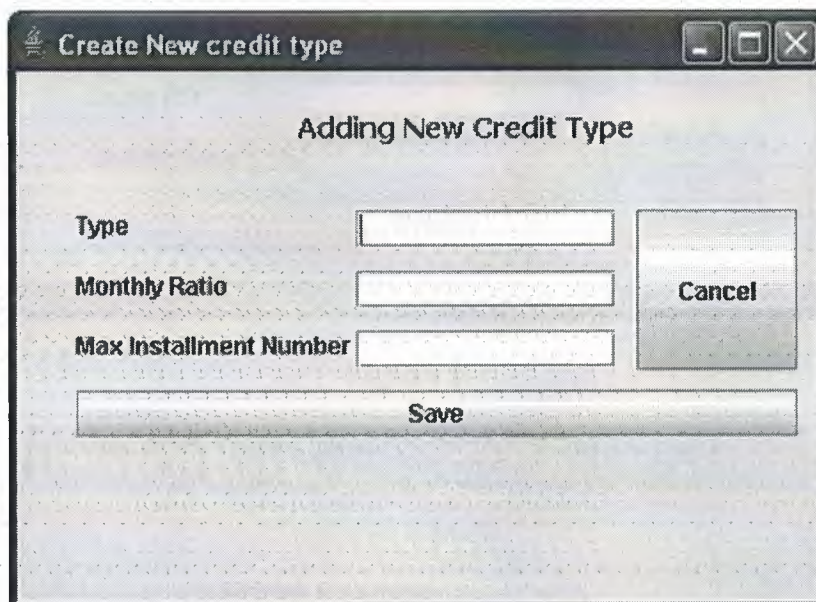
- Buttons:** A vertical stack of buttons labeled "Add New User", "Add New Credit Type", "Add New Bank", "Add New Country", "Managerial Reports", and "Switch To User Menu".
- Exit:** An "Exit" button located to the right of the main menu buttons.

Figure 5 :



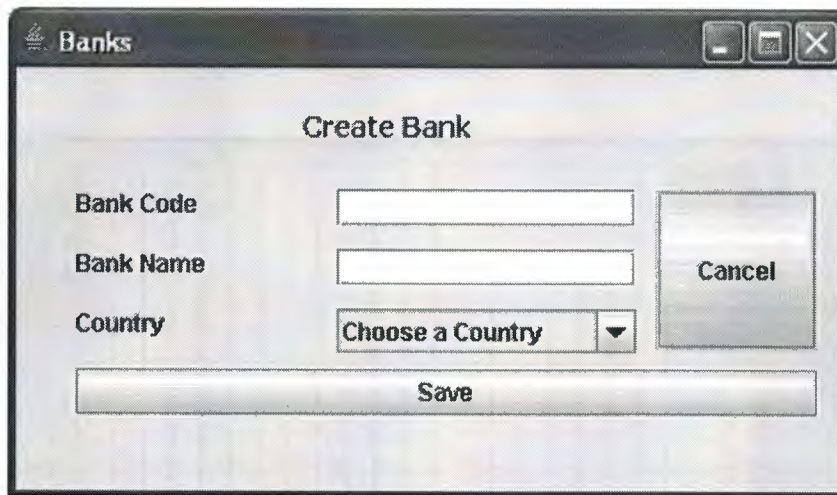
The 'Users' dialog box features a title bar with a standard icon and window controls. The main heading is 'Create User'. It contains four input fields: 'User Name', 'Password', 'User Type' (a dropdown menu currently showing 'Choose a Type'), and 'User Description'. To the right of these fields is a 'Cancel' button. At the bottom center is a 'Save' button.

Figure 6 :



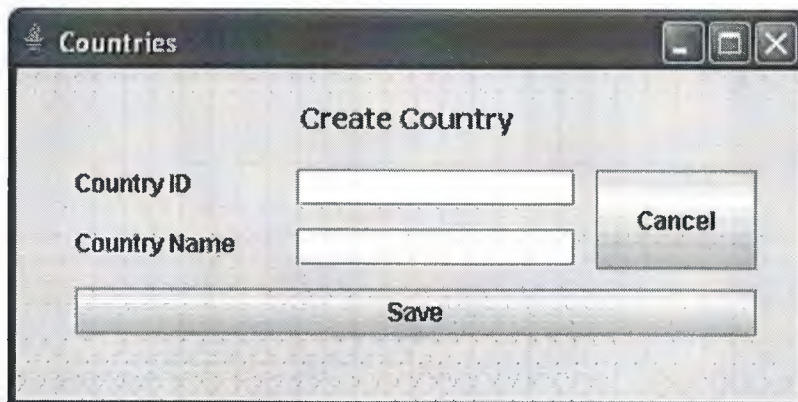
The 'Create New credit type' dialog box has a title bar with a standard icon and window controls. The main heading is 'Adding New Credit Type'. It contains three input fields: 'Type', 'Monthly Ratio', and 'Max Installment Number'. To the right of these fields is a 'Cancel' button. At the bottom center is a 'Save' button.

Figure 7 :



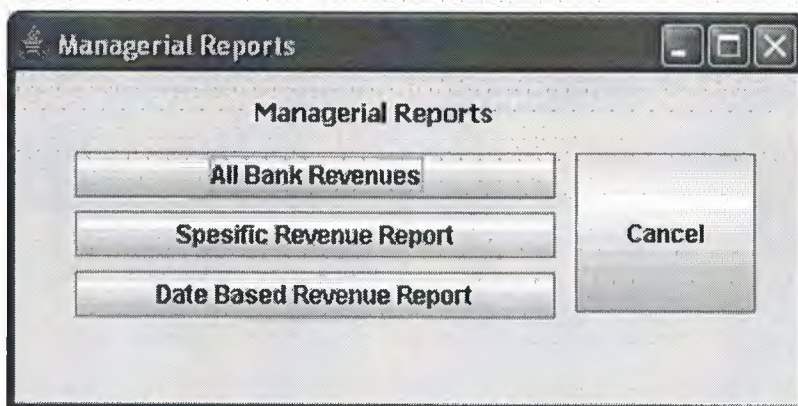
The screenshot shows a window titled "Banks" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, the title "Create Bank" is centered. Below the title, there are three input fields: "Bank Code" (a text box), "Bank Name" (a text box), and "Country" (a dropdown menu with the text "Choose a Country" and a downward arrow). To the right of these fields is a "Cancel" button. At the bottom of the form is a wide "Save" button.

Figure 8 :



The screenshot shows a window titled "Countries" with a standard Windows-style title bar. Inside, the title "Create Country" is centered. Below it are two input fields: "Country ID" (a text box) and "Country Name" (a text box). To the right of these fields is a "Cancel" button. At the bottom of the form is a wide "Save" button.

Figure 9 :



The screenshot shows a window titled "Managerial Reports" with a standard Windows-style title bar. Inside, the title "Managerial Reports" is centered. Below the title, there are three buttons stacked vertically: "All Bank Revenues", "Spesific Revenue Report", and "Date Based Revenue Report". To the right of these buttons is a "Cancel" button.

All Accounts

[illegible]

Cancel

Figure 11 :

Specific Revenue Report

Revenue Search

☒ Search By Customer

☐ Search By Account

☐ Search By Revenue Type

Choose Customer Choose a Customer

Cancel

Revenue Date	Number	Revenue Amount	Revenue Type

Figure 12 :

Date based Revenue Report

Date Based Revenue Report

Starting Date January 1 2005

Ending Date January 1 2006

Cancel

Search

Revenue Date	Customer Num...	Account Number	Revenue Amount	Type

Figure 13 :

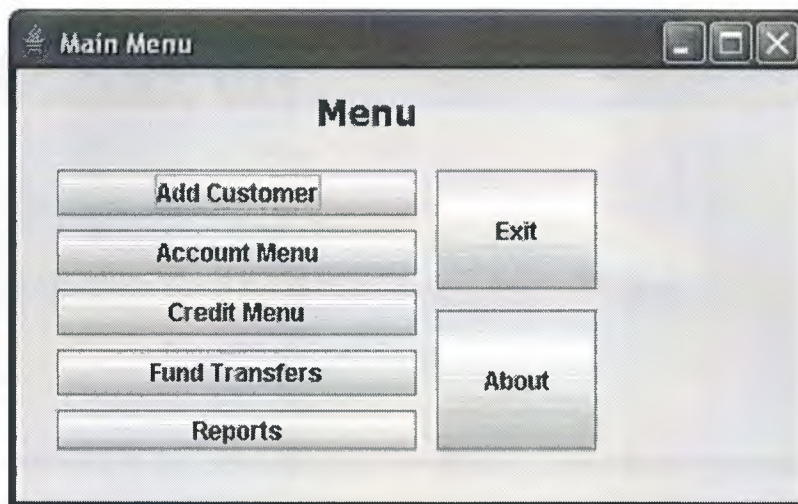


Figure 14 :

The screenshot shows a window titled "Add Customer" with a standard Windows-style title bar. The main content area is titled "New Customer" and contains several input fields and buttons. The fields are: "Number" (text box), "Name" (text box), "Surname" (text box), "Gender" (dropdown menu with "Male" selected), "Birthdate" (date picker showing "January 1, 1989"), "Phone" (text box), and "Address" (text box). To the right of the fields is a vertical button labeled "Cancel". At the bottom of the window is a wide button labeled "Save".

Figure 15 :

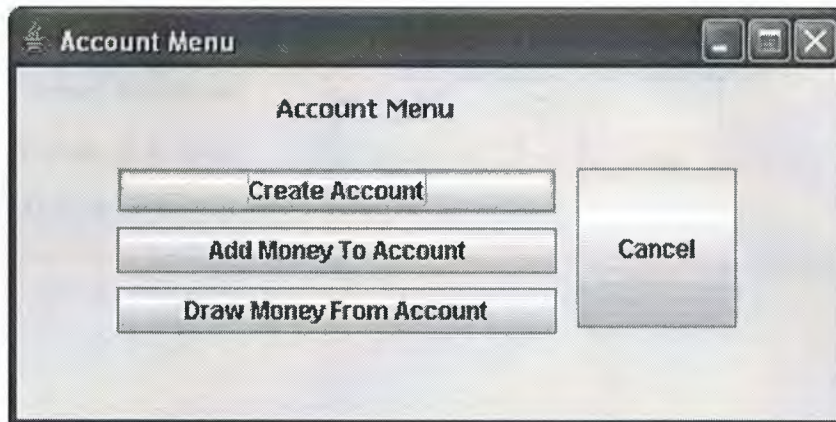


Figure 16 :

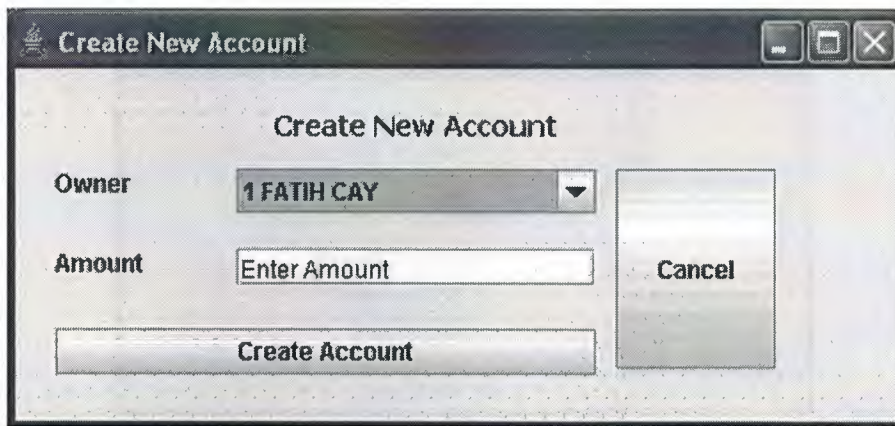


Figure 17 :

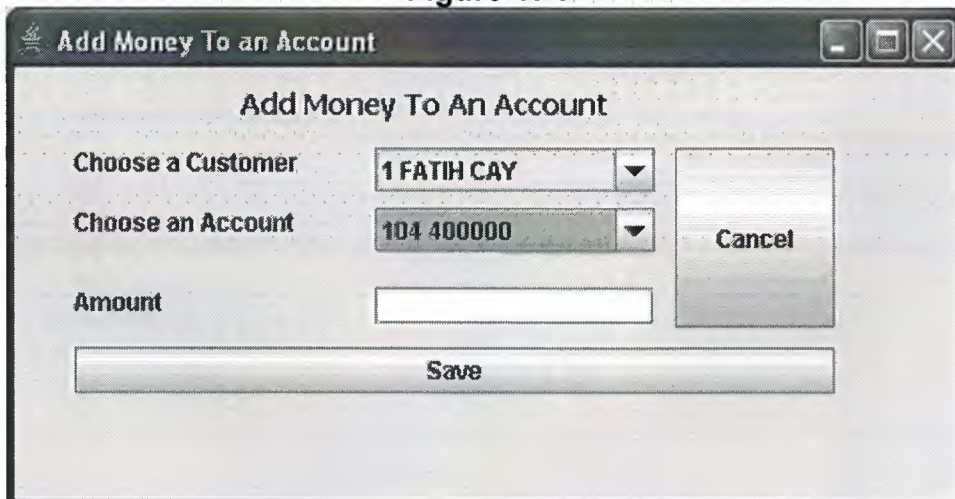
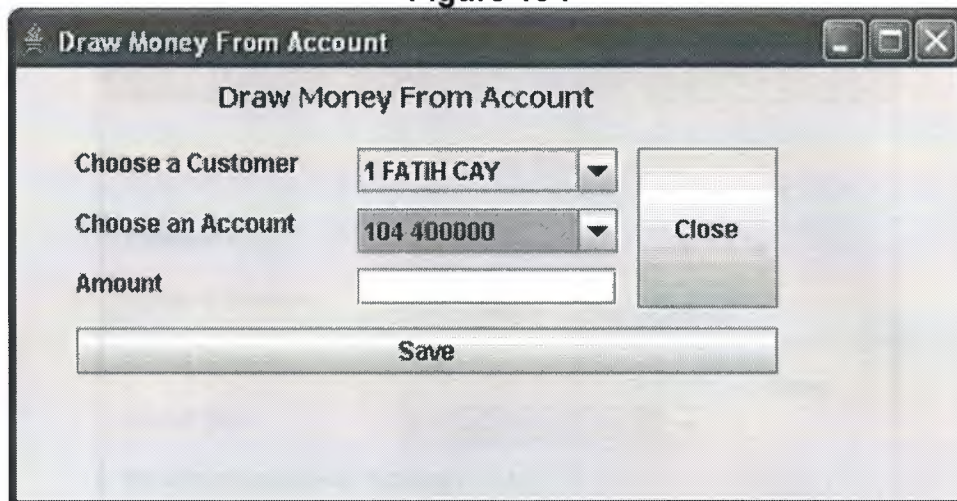
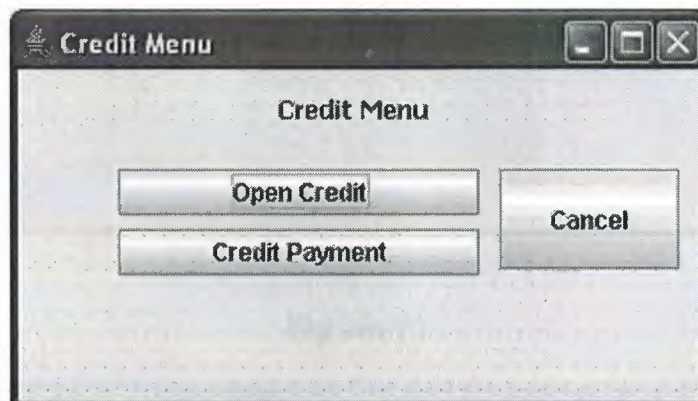


Figure 18 :



The screenshot shows a window titled "Draw Money From Account". Inside the window, there is a title bar with standard Windows controls (minimize, maximize, close). The main content area has the title "Draw Money From Account" at the top. Below the title, there are three input fields: "Choose a Customer" with a dropdown menu showing "1 FATIH CAY", "Choose an Account" with a dropdown menu showing "104 400000", and "Amount" with an empty text box. To the right of these fields is a "Close" button. At the bottom of the window is a wide "Save" button.

Figure 19 :



The screenshot shows a window titled "Credit Menu". Inside the window, there is a title bar with standard Windows controls (minimize, maximize, close). The main content area has the title "Credit Menu" at the top. Below the title, there are two buttons: "Open Credit" and "Credit Payment". To the right of these buttons is a "Cancel" button.

Figure 20 :



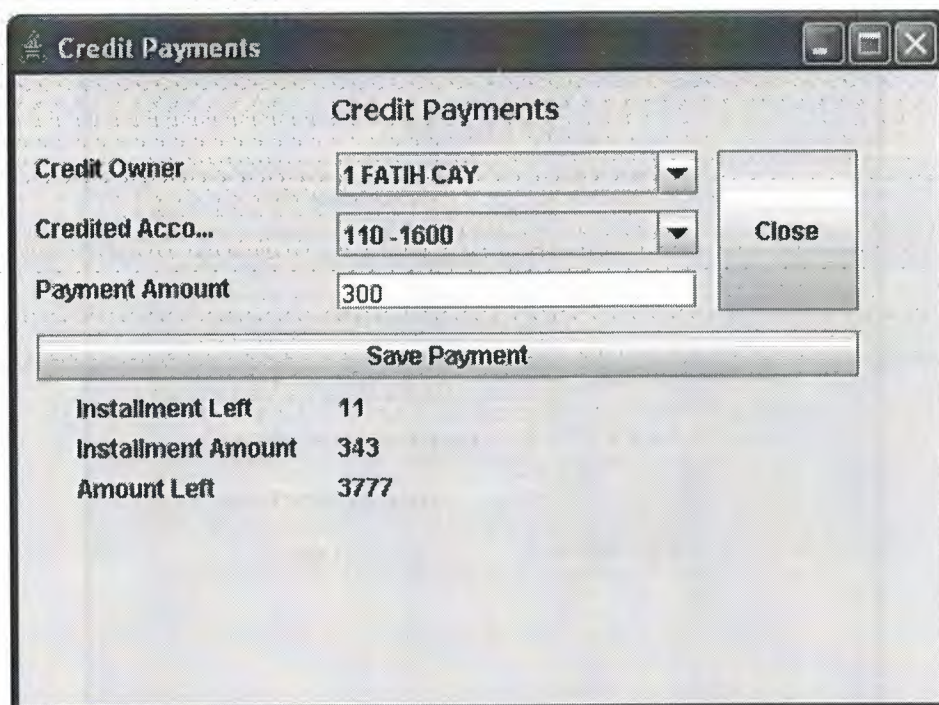
A screenshot of a software window titled "Credit Opening". The window contains several input fields and a "Close" button. The fields are: "Customer" with a dropdown menu showing "1 FATIH CAY", "Account Number" with a dropdown menu showing "104 400000", "Credit Amount" with a text box containing "500", and "Credit Type" with a dropdown menu showing "21 Car %3". Below these fields, there are three lines of text: "Maximum Instalment Number 12", "Monthly Payment 42", and "Total Payment 515". At the bottom of the window is a "Save" button.

Credit Opening	
Customer	1 FATIH CAY
Account Number	104 400000
Credit Amount	500
Credit Type	21 Car %3
Maximum Instalment Number	12
Monthly Payment	42
Total Payment	515

Close

Save

Figure 21 :



A screenshot of a software window titled "Credit Payments". The window contains several input fields and a "Close" button. The fields are: "Credit Owner" with a dropdown menu showing "1 FATIH CAY", "Credited Acco..." with a dropdown menu showing "110 -1600", and "Payment Amount" with a text box containing "300". Below these fields is a "Save Payment" button. At the bottom of the window, there are three lines of text: "Installment Left 11", "Installment Amount 343", and "Amount Left 3777".

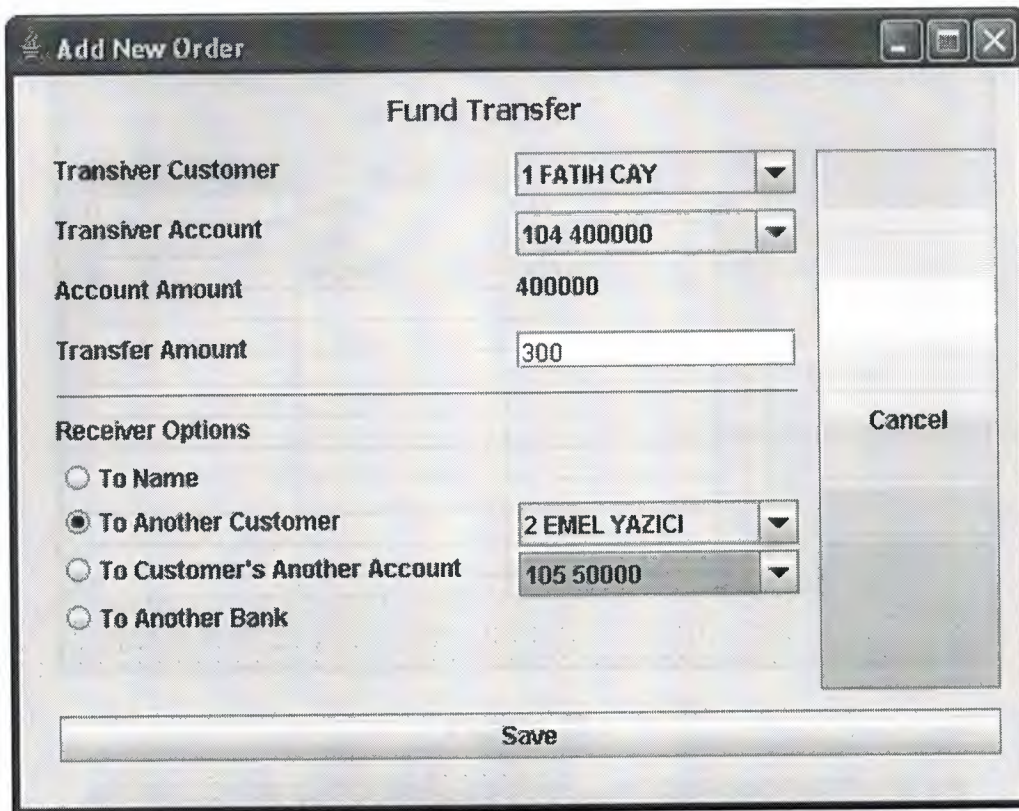
Credit Payments	
Credit Owner	1 FATIH CAY
Credited Acco...	110 -1600
Payment Amount	300

Close

Save Payment

Installment Left	11
Installment Amount	343
Amount Left	3777

Figure 22 :



The dialog box is titled "Add New Order" and contains a section for "Fund Transfer". It includes fields for "Transiver Customer" (1 FATIH CAY), "Transiver Account" (104 400000), "Account Amount" (400000), and "Transfer Amount" (300). Below these is a "Receiver Options" section with four radio buttons: "To Name", "To Another Customer" (selected), "To Customer's Another Account", and "To Another Bank". The "To Another Customer" option has associated fields for "2 EMEL YAZICI" and "105 50000". A "Cancel" button is on the right, and a "Save" button is at the bottom.

Add New Order

Fund Transfer

Transiver Customer: 1 FATIH CAY

Transiver Account: 104 400000

Account Amount: 400000

Transfer Amount: 300

Receiver Options

- ☐ To Name
- ☒ To Another Customer: 2 EMEL YAZICI
- ☐ To Customer's Another Account: 105 50000
- ☐ To Another Bank

Cancel

Save

Figure 23 :



The dialog box is titled "Report Menu" and contains a section for "Reports Menu". It lists seven report options as buttons: "All Customers", "Specific Customer Report", "Customer's Balance Report", "All Accounts", "Specific Account Report", "All Fund Transfers", and "Credit Reports". A "Cancel" button is on the right.

Report Menu

Reports Menu

- All Customers
- Specific Customer Report
- Customer's Balance Report
- All Accounts
- Specific Account Report
- All Fund Transfers
- Credit Reports

Cancel

Figure 24 :

The screenshot shows a window titled "All Customers". Inside, there is a table with three columns: "Customer Number", "Name", and "Surname". The table contains three rows of data. To the right of the table is a large "Cancel" button.

Customer Number	Name	Surname
1	FATIH	CAY
2	EMEL	YAZICI
3	NADIRE	CAVUS

Cancel

Figure 25 :

The screenshot shows a window titled "Specific Customer Search". It has a section titled "Customer Report" with three input fields: "Customer Name :", "Customer Surna...", and "Customer Social Security Numb...". The "Customer Name" field contains the text "FATIH". To the right of these fields is a "Cancel" button. Below the input fields is a "Search" button. Below the "Search" button is a table with three columns: "Customer Number", "Name", and "Surname". The table contains one row of data.

Customer Name : FATIH

Customer Surna...

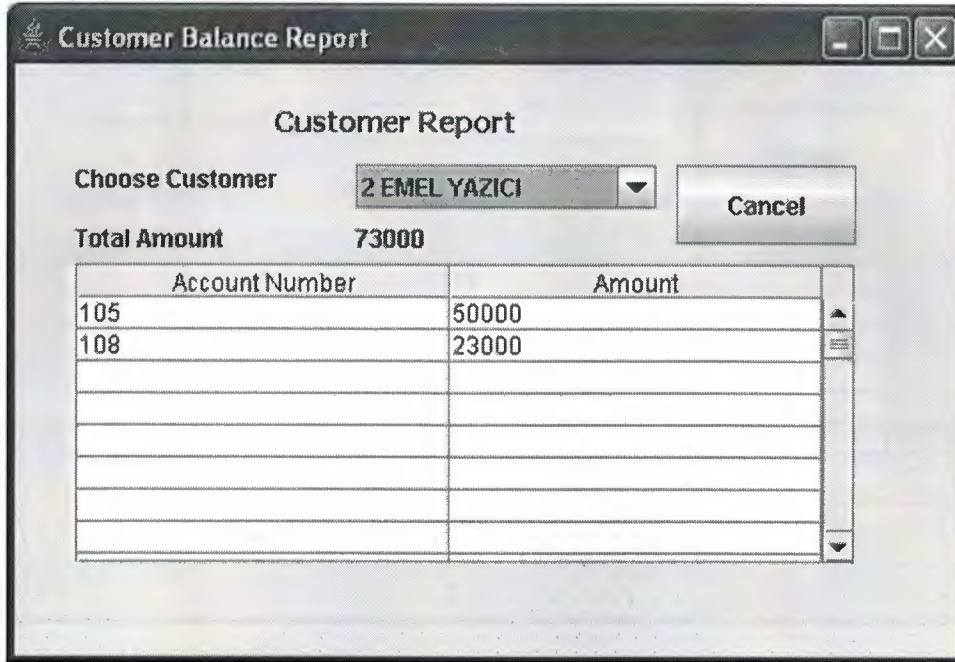
Customer Social Security Numb...

Cancel

Search

Customer Number	Name	Surname
1	FATIH	CAY

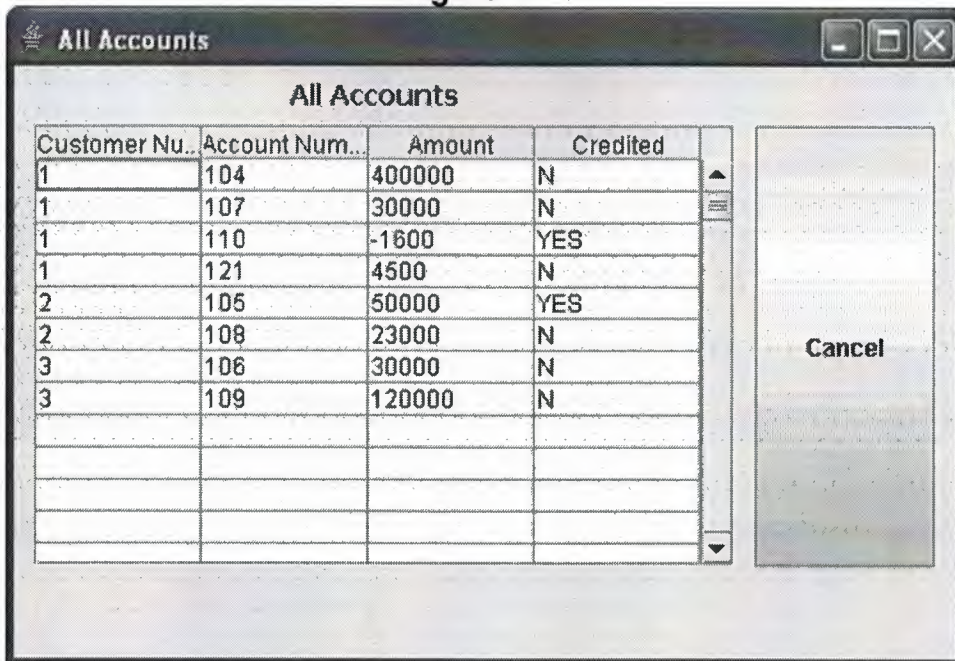
Figure 26 :



The 'Customer Balance Report' window displays a report for customer '2 EMEL YAZICI'. It shows a total amount of 73000. Below this, a table lists account numbers and their corresponding amounts.

Account Number	Amount
105	50000
108	23000

Figure 27 :



The 'All Accounts' window displays a table of all accounts. The table has four columns: Customer Nu., Account Num., Amount, and Credited. A 'Cancel' button is located on the right side of the window.

Customer Nu.	Account Num.	Amount	Credited
1	104	400000	N
1	107	30000	N
1	110	-1600	YES
1	121	4500	N
2	105	50000	YES
2	108	23000	N
3	106	30000	N
3	109	120000	N

Figure 28 :

Specific Account Search

Account Number

Search **Cancel**

Account Number	Customer Number	Amount	Credited

All Accounts

[illegible]

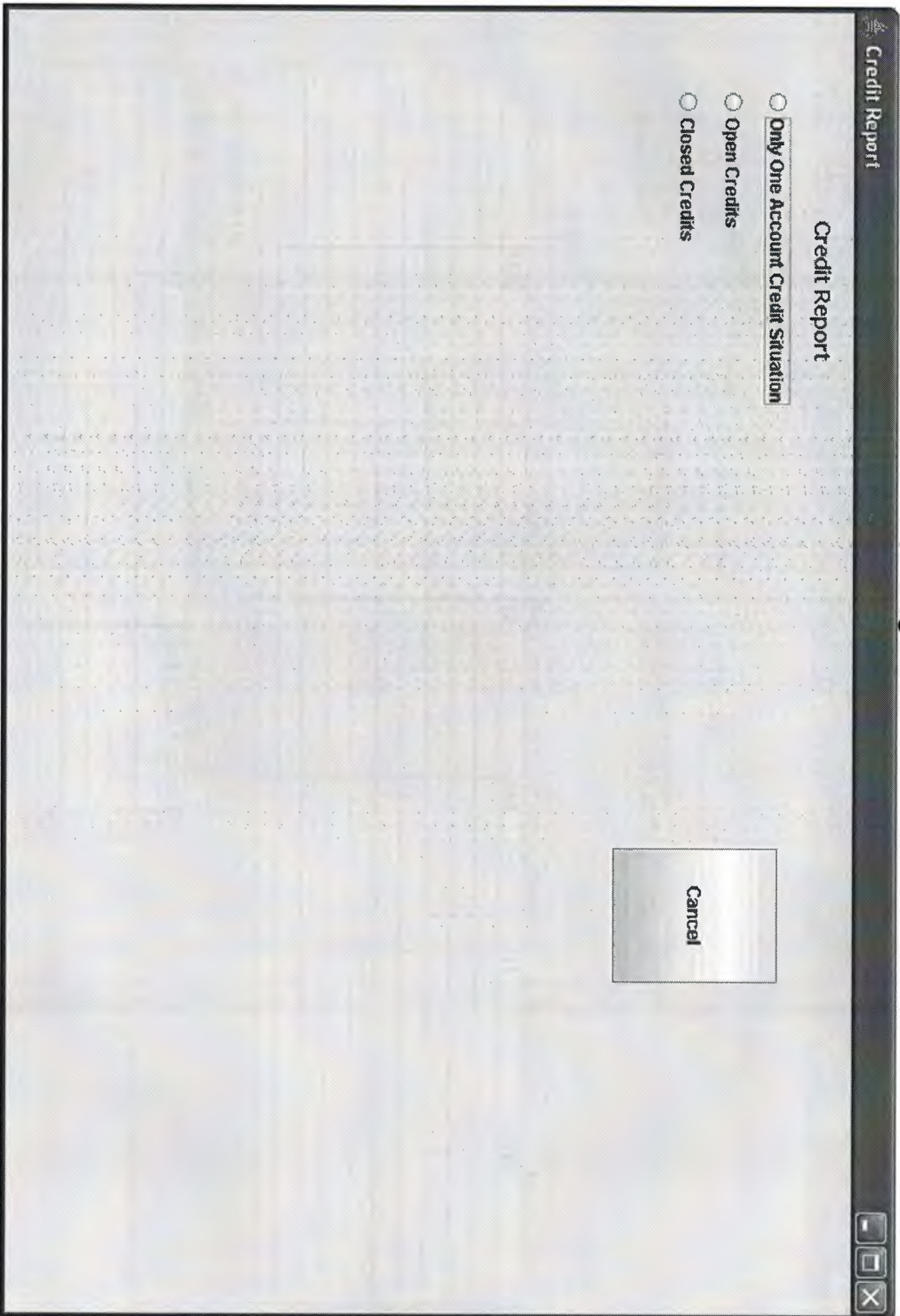


Figure 30 :

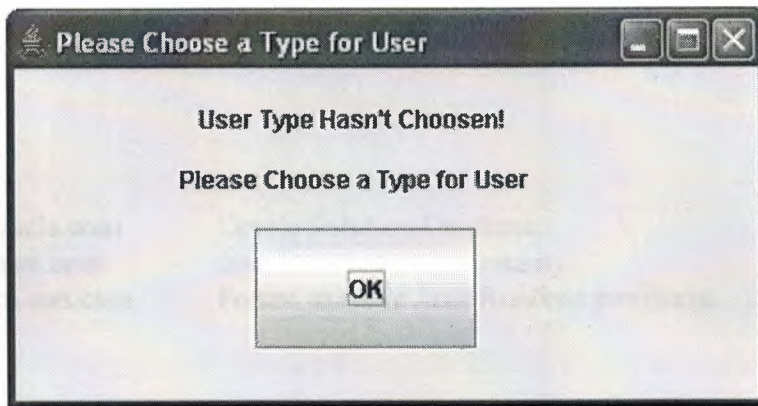
Credit Report

75

[illegible]

[illegible]

Figure 34 :



REFERENCES

Web Sites :

- | | |
|-------------------|---------------------------------------|
| 1) otn.oracle.com | Oracle Solution Database. |
| 2) Java.sun.com | Java Developer Community. |
| 3) Forum.sun.com | Forum to solve Java Runtime problems. |

E-Books :

- | | | |
|---------------------|-------------|--|
| 1) Thinking In Java | Bruce Eckel | www.bruceeckel.com |
| 2) Java Cookbook | Ian Darwin | O'Reilly Publishing June/2001 |

Book :

Database Programming With Oracle Nilesh Shah Prentice Hall October/2004

APPENDIX II

Source Codes

Main.Java

```
/*
 * Main.java
 *
 * Created on December 1, 2005, 9:14 PM
 *
 * To change this template, choose Tools | Options and locate the template under
 * the Source Creation and Management node. Right-click the template and choose
 * Open. You can then make changes to the template in the Source Editor.
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Main {

    /** Creates a new instance of Main */
    public Main() {
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        Usrpass nusrpas=new Usrpass();
        nusrpas.setTitle("Please Enter Your Username and Password");
        nusrpas.setSize(420,130);
        nusrpas.setVisible(true);

        /*
        Cimp x=new Cimp();
        x.setVisible(true);
        */

    }
}
```

Usrpass.Java

```
/*
 * Usrpass.java
 *
 * Created on December 4, 2005, 11:04 AM
 */

package c;
import java.sql.*;

/**
 *
 * @author Cheyenne
 */
public class Usrpass extends javax.swing.JFrame {
    private static String usr="";
    private String pass="";
    private String rank;

    /** Creates new form Usrpass */
    public Usrpass() {
        initComponents();
        jButton4.setVisible(false);
        this.setLocation(200, 200);
    }

    public static String getuser(){

        return usr;
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jTextField1 = new javax.swing.JTextField();
        jPasswordField1 = new javax.swing.JPasswordField();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();
    }
}
```

```

jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jLabel6 = new javax.swing.JLabel();
jButton2 = new javax.swing.JButton();
jPasswordField2 = new javax.swing.JPasswordField();
jPasswordField3 = new javax.swing.JPasswordField();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
jButton3 = new javax.swing.JButton();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setResizable(false);
getContentPane().add(jTextField1);
jTextField1.setBounds(120, 20, 90, 19);

getContentPane().add(jPasswordField1);
jPasswordField1.setBounds(120, 50, 90, 20);

jLabel1.setText("Username");
getContentPane().add(jLabel1);
jLabel1.setBounds(30, 20, 80, 14);

jLabel2.setText("Password");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 50, 70, 14);

jButton1.setText("Connect");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(220, 20, 90, 50);

jButton5.setText("EXIT");
jButton5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton5ActionPerformed(evt);
    }
});

getContentPane().add(jButton5);
jButton5.setBounds(320, 20, 90, 50);

jButton4.setText("Continue");
jButton4.addActionListener(new java.awt.event.ActionListener() {

```



```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton4ActionPerformed(evt);
        }
    });

```

```

getContentPane().add(jButton4);
jButton4.setBounds(220, 20, 90, 130);

```

```

jLabel3.setText("User Description :");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 100, 110, 14);

```

```

jLabel4.setText(" ");
getContentPane().add(jLabel4);
jLabel4.setBounds(160, 100, 140, 14);

```

```

jLabel5.setText("User Type      :");
getContentPane().add(jLabel5);
jLabel5.setBounds(30, 130, 110, 14);

```

```

jLabel6.setText(" ");
getContentPane().add(jLabel6);
jLabel6.setBounds(160, 130, 140, 14);

```

```

jButton2.setText("Change Password");
jButton2.setMaximumSize(new java.awt.Dimension(73, 23));
jButton2.setMinimumSize(new java.awt.Dimension(73, 23));
jButton2.setOpaque(false);
jButton2.setPreferredSize(new java.awt.Dimension(73, 23));
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

```

```

getContentPane().add(jButton2);
jButton2.setBounds(30, 153, 280, 50);

```

```

getContentPane().add(jPasswordField2);
jPasswordField2.setBounds(210, 220, 90, 20);

```

```

getContentPane().add(jPasswordField3);
jPasswordField3.setBounds(210, 250, 90, 20);

```

```

jLabel7.setText("New Password      :");
getContentPane().add(jLabel7);
jLabel7.setBounds(30, 220, 180, 14);

```

```

jLabel8.setText("Re enter New Password  :");
getContentPane().add(jLabel8);

```

```

jLabel8.setBounds(30, 250, 170, 14);

jButton3.setText("Apply Changes");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(30, 280, 280, 23);

pack();
}
// </editor-fold>

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    Connection con;
    String Ssql="";
    Statement stm;
    ResultSet sonuc;
    String ur="";
    String ursn="";
    String as="";
    String sa="";
    String u="";
    String us="";
    String newpass="";
    String newpass2="";
    String new1="";
    String new2="";
    int r;

    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }

    newpass=jPasswordField2.getText().toUpperCase();
    newpass2=jPasswordField3.getText().toUpperCase();
    new1=""+newpass+"";
    new2=""+usr+"";
    Ssql="update users set usrsnm="+new1+" where usrm= "+new2+" ";

```

```

if (newpass.equals(newpass2)){
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        stm.addBatch(Ssql);
        stm.executeBatch();
        stm.close();
        con.close();
        this.setSize(420, 360);
        this.setVisible(true);
        Ermes ermess=new Ermes("Password Changed", "Successful");
        ermess.setVisible(true);

        } catch(SQLException ex) {
        }
    }else{
        this.setSize(420, 360);
        this.setVisible(true);
        Ermes ermess=new Ermes("New Password Doesn't Match", "Wrong
Password");
        ermess.setVisible(true);

    }

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    this.setSize(420,340);
    jButton5.setSize(90,283);
    this.setVisible(true);
}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    System.exit(0);
}

private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    jButton5.setSize(55,120);
    System.out.println(rank);
    if (rank.equals("ADMIN")){

```



```

    Managermenu mmenu=new Managermenu();
    mmenu.setTitle("Main Menu");
    mmenu.setSize(400, 400);
    mmenu.setVisible(true);
}else{
    Menu anamenu=new Menu();
    anamenu.setTitle("Main Menu");
    anamenu.setSize(400, 250);
    anamenu.setVisible(true);
}
this.setVisible(false);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    Boolean usernamecheck=true;
    Boolean passwordcheck=true;
    Connection con;
    String Ssql="";
    Statement stm;
    ResultSet sonuc;
    String ur="";
    String ursn="";
    String as="";
    String sa="";
    String u="";
    String us="";
    int r;

    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    u=jTextField1.getText().toUpperCase();
    if (u==""){
        usernamecheck=false;
        Ermes ermess=new Ermes("Username is Missing", "Please Fill Username
Field");
        ermess.setVisible(true);
    }
    us=jPasswordField1.getText().toUpperCase();
    if (us==""){
        passwordcheck=false;
        Ermes ermess=new Ermes("Password is Missing", "Please Fill Password
Field");
    }
}

```



```

        ermess.setVisible(true);
    }
    jPasswordField1.setText("");
    ur=""+"u+"";
    ursn=""+"us+"";
    if ((usernamecheck==true)&&(passwordcheck==true)){
        Ssql="select * from users where usnm= "+ur+" and usrsnm= "+ursn+" order
by USRID";
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            r=0;
            while (sonuc.next()){
                r=r+1;
                as=sonuc.getString(4);
                jLabel4.setText(as);
                sa=sonuc.getString(5);
                rank=sa;
                jLabel6.setText(sa);
            }

            if (r!=0){
                this.setSize(420,240);
                jButton5.setSize(90,183);
                jButton1.setVisible(false);
                jButton4.setVisible(true);
                jLabel2.setVisible(false);
                jPasswordField1.setVisible(false);
                this.setVisible(true);
                usr=u;
                pass=us;
                stm.close();
                con.close();
            }else{
                Ermes ermess=new Ermes("User & Password Doesn't Match", "Wrong
Password");
                ermess.setVisible(true);
                stm.close();
                con.close();
            }
        } catch(SQLException ex) {
        }
    }
}
}

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Usrpass().setVisible(true);

        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JPasswordField jPasswordField1;
private javax.swing.JPasswordField jPasswordField2;
private javax.swing.JPasswordField jPasswordField3;
private javax.swing.JTextField jTextField1;
// End of variables declaration
}

```

Menu.Java

```
/*
 * Menu.java
 *
 * Created on January 28, 2006, 9:42 PM
 */
package c;

/**
 *
 * @author Cheyenne
 */
public class Menu extends javax.swing.JFrame {

    /** Creates new form Menu */
    public Menu() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton8 = new javax.swing.JButton();
        jButton11 = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        jButton2.setText("Exit");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });

        getContentPane().add(jButton2);
    }
}
```



```

jButton2.setBounds(210, 50, 80, 60);

jButton3.setText("Add Customer");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(20, 50, 180, 23);

jButton4.setText("Credit Menu");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

getContentPane().add(jButton4);
jButton4.setBounds(20, 110, 180, 23);

jButton8.setText("Fund Transfers");
jButton8.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton8ActionPerformed(evt);
    }
});

getContentPane().add(jButton8);
jButton8.setBounds(20, 140, 180, 23);

jButton11.setText("About");
jButton11.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton11ActionPerformed(evt);
    }
});

getContentPane().add(jButton11);
jButton11.setBounds(210, 120, 80, 70);

jLabel1.setFont(new java.awt.Font("Tahoma", 1, 18));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Menu");
getContentPane().add(jLabel1);
jLabel1.setBounds(110, 10, 130, 22);

jButton1.setText("Account Menu");
jButton1.addActionListener(new java.awt.event.ActionListener() {

```



```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(20, 80, 180, 23);

    jButton6.setText("Reports");
    jButton6.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton6ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton6);
    jButton6.setBounds(20, 170, 180, 20);

    pack();
}
// </editor-fold>

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reports nreports= new Reports();
    nreports.setTitle("Report Menu");
    nreports.setSize(400, 400);
    nreports.setVisible(true);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Accountmenu naccountmenu= new Accountmenu();
    naccountmenu.setTitle("Account Menu");
    naccountmenu.setSize(420, 210);
    naccountmenu.setVisible(true);
}

private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    About nabout= new About();
    nabout.setTitle("About");
    nabout.setSize(510, 550);
    nabout.setVisible(true);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Addcustomer ncustomer= new Addcustomer();
    ncustomer.setTitle("Add Customer");
}

```

```

        ncustomer.setSize(510, 400);
        ncustomer.setVisible(true);
    }

    private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        Addorder naddorder= new Addorder();
        naddorder.setTitle("Add New Order");
        naddorder.setSize(510, 400);
        naddorder.setVisible(true);
    }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        Creditmenu ncreditmenu=new Creditmenu();
        ncreditmenu.setTitle("Credit Menu");
        ncreditmenu.setSize(350, 200);
        ncreditmenu.setVisible(true);
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        System.exit(0);
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Menu().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton11;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JButton jButton6;
    private javax.swing.JButton jButton8;
    private javax.swing.JLabel jLabel1;
    // End of variables declaration

}

```

Managermenu.java

```
/*
 * Managermenu.java
 *
 * Created on February 13, 2006, 6:03 AM
 */

package c;
import javax.swing.*;
/**
 *
 * @author Cheyenne
 */
public class Managermenu extends javax.swing.JFrame {

    /** Creates new form Managermenu */
    public Managermenu() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();
        jButton7 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Managerial Menu");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(90, 10, 180, 17);

        jButton1.setText("Add New User");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {

```



```

        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(40, 60, 160, 23);

jButton2.setText("Add New Credit Type");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(40, 90, 160, 23);

jButton3.setText("Add New Bank");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(40, 120, 160, 23);

jButton4.setText("Add New Country");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

getContentPane().add(jButton4);
jButton4.setBounds(40, 150, 160, 23);

jButton5.setText("Switch To User Menu");
jButton5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton5ActionPerformed(evt);
    }
});

getContentPane().add(jButton5);
jButton5.setBounds(40, 210, 160, 50);

jButton6.setText("Exit");
jButton6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```



```

        jButton6ActionPerformed(evt);
    }
});

getContentPane().add(jButton6);
jButton6.setBounds(210, 60, 80, 200);

jButton7.setText("Managerial Reports");
jButton7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton7ActionPerformed(evt);
    }
});

getContentPane().add(jButton7);
jButton7.setBounds(40, 180, 160, 23);

pack();
}
// </editor-fold>

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Managerreport nmanagereport=new Managerreport();
    nmanagereport.setTitle("Managerial Reports");
    nmanagereport.setSize(400, 200);
    nmanagereport.setVisible(true);
}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Menu anamenu=new Menu();
    anamenu.setTitle("Main Menu");
    anamenu.setSize(400, 250);
    anamenu.setVisible(true);
}

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    System.exit(0);
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Country ccountry=new Country();
    ccountry.setTitle("Countries");
    ccountry.setSize(400, 200);
    ccountry.setVisible(true);
}

```

```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Banks nbanks=new Banks();
    nbanks.setTitle("Banks");
    nbanks.setSize(420, 245);
    nbanks.setVisible(true);
}

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Addcreditttype ctype=new Addcreditttype();
    ctype.setTitle("Create New credit type");
    ctype.setSize(410, 300);
    ctype.setVisible(true);
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Users nusers= new Users();
    nusers.setTitle("Users");
    nusers.setSize(380, 290);
    nusers.setVisible(true);
}

```

```

/**
 * @param args the command line arguments
 */

```

```

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Managermenu().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JButton jButton6;
private javax.swing.JButton jButton7;
private javax.swing.JLabel jLabel1;
// End of variables declaration

```

```

}

```

Allaccounts.Java

```
/*
 * Allaccounts.java
 *
 * Created on February 14, 2006, 6:13 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Allaccounts extends javax.swing.JFrame {

    /** Creates new form Allaccounts */
    public Allaccounts() {
        initComponents();
        this.setLocation(200,200);
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        jTable1.setValueAt("Name", 0, 1);
        jTable1.setValueAt("Surname", 0, 2);

        Ssql = ("Select ACUSTNUM,ACCOUNTNUM,AAMOUNT,acredited from
accounts order by acustnum,accountnum");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            int row=0;
            while (sonuc.next()){
                //s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                int column=0;
                jTable1.setValueAt(sonuc.getString(1),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(2),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(3),row, column);
            }
        }
    }
}
```



```

        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        row++;
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel1.setText("All Accounts");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(100, 4, 160, 20);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
        }
    ));

```



```
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
{null, null, null, null},  
}  
new String [] {  
    "Customer Number", "Account Number", "Amount", "Credited"  
}  
));  
jScrollPane1.setViewportView(jTable1);  
  
getContentPane().add(jScrollPane1);  
jScrollPane1.setBounds(10, 30, 350, 220);  
  
jButton1.setText("Cancel");  
jButton1.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        jButton1ActionPerformed(evt);
```



```

    }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(370, 30, 90, 220);

    pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Allaccounts().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration

}

```

Addcredittype.Java

```
/*
 * Addcredittype.java
 *
 * Created on January 28, 2006, 9:12 PM
 */

package c;
import javax.swing.*;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Addcredittype extends javax.swing.JFrame {
    String user=Usrpass.getuser();

    /** Creates new form Addcredittype */
    public Addcredittype() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jTextField3 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Adding New Credit Type");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(110, 20, 230, 17);
    }
}
```

```

jLabel2.setText("Type");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 70, 70, 14);

jLabel3.setText("Monthly Ratio");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 100, 120, 14);

jLabel4.setText("Max Installment Number");
getContentPane().add(jLabel4);
jLabel4.setBounds(30, 130, 140, 14);

getContentPane().add(jTextField1);
jTextField1.setBounds(170, 70, 130, 19);

getContentPane().add(jTextField2);
jTextField2.setBounds(170, 100, 130, 19);

getContentPane().add(jTextField3);
jTextField3.setBounds(170, 130, 130, 19);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 160, 360, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(310, 70, 80, 80);

pack();
}
// </editor-fold>
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
}

```



```

    }
    return true; // valid
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Boolean ratiocheck=true;
    Boolean maxcheck=true;
    Boolean typecheck=true;
    String type;
    String ratio;
    String max;

    type=jTextField1.getText();
    if (isNumber(type)==false){
        typecheck=true;
    }else{
        typecheck=false;
        Ermes ermess=new Ermes("Credit Type Wrong","Please Enter Character
Value For Credit Type !");
        ermess.setVisible(true);
    }
    ratio=jTextField2.getText();
    if (isNumber(ratio)==true){
        ratiocheck=true;
    }else{
        ratiocheck=false;
        Ermes ermess=new Ermes("Ratio Wrong","Please Enter Numerical Value
For Ratio !");
        ermess.setVisible(true);
    }
    max=jTextField3.getText();
    if (isNumber(max)==true){
        maxcheck=true;
    }else{
        maxcheck=false;
        Ermes ermess=new Ermes("Max Installment Number Wrong","Please Enter
Numerical Value For Max Installment Number !");
        ermess.setVisible(true);
    }
    if (type==""){
        Ermes ermess=new Ermes("Credit Type Name is Missing,Please Fill It!");
        ermess.setVisible(true);
    }else{
        if (ratio==""){
            Ermes ermess=new Ermes("Credit Ratio is Missing,Please Fill It!");
            ermess.setVisible(true);
        }else{
            if (max==""){
                Ermes ermess=new Ermes("Maximum Insallment Number is
Missing,Please Fill It!");

```



```

    ermess.setVisible(true);
}else{

    if ((typecheck==true) && (ratiocheck==true) && (maxcheck==true)){
        //Creating a Credit Type
        Connection con;
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            PreparedStatement pstmt=con.prepareStatement(
                "INSERT INTO credittypes(type,ratio,maxinsnumber,usr)
VALUES(?,?,?,?)");
            String one=type;
            String two=ratio;
            String three=max;
            pstmt.setString(1,one);
            pstmt.setString(2,two);
            pstmt.setString(3,three);
            pstmt.setString(4,user);
            pstmt.executeUpdate();

            pstmt.close();
            con.close();
            jTextField1.setText("");

            Ermes ermess=new Ermes("Credit Type Created Successfully ! ",
                "Type Created");
            ermess.setVisible(true);
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
            Ermes ermess=new Ermes(type,type);
            ermess.setVisible(true);
        }
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);// TODO add your handling code here:
}

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Addcredittype().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration

```

```

}

```

Reportaccount.Java

```
/*
 * Reportaccount.java
 *
 * Created on February 14, 2006, 6:26 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Reportaccount extends javax.swing.JFrame {

    /** Creates new form Reportaccount */
    public Reportaccount() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
        jButton2 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Specific Account Search");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(120, 10, 210, 14);

        jLabel2.setText("Account Number");
        getContentPane().add(jLabel2);
        jLabel2.setBounds(30, 40, 120, 14);
    }
}
```

```
getContentPane().add(jTextField1);
jTextField1.setBounds(170, 40, 160, 19);
```

```
jButton1.setText("Search");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton1);
jButton1.setBounds(20, 70, 310, 23);
```

[illegible]


```

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

String account="";
account=jTextField1.getText();

if (account.equals("")){
    Ermes ermess=new Ermes("Search Failed!","Please Fill Account Number!");
    ermess.setVisible(true);
}else{
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;
    Ssql = ("Select ACCOUNTNUM,ACUSTNUM,AAMOUNT,acredited from
accounts where accountnum="+account+" ");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        int row=0;
        int rowcount=jTable1.getRowCount();
        int columncount=jTable1.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable1.setValueAt("", rown,columnn);
            }
        }
        while (sonuc.next()){
            int column=0;
            jTable1.setValueAt(sonuc.getString(1),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(2),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(3),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(4),row, column);
            row++;
        }
    }
}

```

```

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
}
}

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Reportaccount().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
// End of variables declaration
}

```

Addorder.Java

```
/*
 * Addorder.java
 *
 * Created on February 9, 2006, 10:27 PM
 */
package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
import java.io.*;
/**
 *
 * @author Cheyenne
 */
public class Addorder extends javax.swing.JFrame implements ItemListener {
    int flag=0;
    String tcustomer="";
    String taccount="";
    String rcustomer="";
    String raccount="";
    String bank="";
    String name="";
    String amount="";
    String accountamount="";
    String user=Usrpass.getuser();
    Boolean amountcheck=true;
    Boolean namecheck=true;
    /** Creates new form Addorder */
    public Addorder() {
        initComponents();
        this.setLocation(200, 200);
        jTextField1.setVisible(false);
        jTextField3.setVisible(false);
        jLabel6.setVisible(false);
        jComboBox5.setVisible(false);
        jComboBox3.setVisible(false);
        jComboBox4.setVisible(false);
        jComboBox1.addItem("Choose a Customer");
        jComboBox3.addItem("Choose a Customer");
        jComboBox2.addItem("Choose an Account");
        jComboBox4.addItem("Choose an Account");
        buttonGroup1.add(jRadioButton1);
        buttonGroup1.add(jRadioButton2);
        buttonGroup1.add(jRadioButton3);
        buttonGroup1.add(jRadioButton4);
        Connection con;
        Statement stm;
```



```

ResultSet sonuc;
String Ssql;

Ssql = ("Select custnum,custnm,custsnm from customers");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
        jComboBox1.addItem(s);
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
jComboBox2.addItemListener(this);
jComboBox3.addItemListener(this);
jComboBox4.addItemListener(this);
jRadioButton1.addItemListener(this);
jRadioButton2.addItemListener(this);
jRadioButton3.addItemListener(this);
jRadioButton4.addItemListener(this);
}
public void itemStateChanged( ItemEvent event ) {

    if( event.getSource() == jRadioButton1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
        jTextField1.setVisible(true);
        jLabel6.setVisible(false);
        jComboBox3.setVisible(false);
        jComboBox4.setVisible(false);
        jTextField3.setVisible(false);
        jComboBox5.setVisible(false);
        flag=1;
    }
    if ( event.getSource() == jRadioButton2&& event.getStateChange() ==
ItemEvent.SELECTED ) {

```



```

jTextField1.setVisible(false);
jTextField3.setVisible(false);
jLabel6.setVisible(false);
jComboBox5.setVisible(false);
jComboBox3.removeItemListener(this);
jComboBox3.removeAllItems();
jComboBox3.addItem("Choose an Account");
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;

Ssql = ("Select custnum,custnm,custsnm from customers");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}

try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
        jComboBox3.addItem(s);
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}

jComboBox3.addItemListener(this);
jComboBox3.setVisible(true);
jComboBox4.removeItemListener(this);
jComboBox4.removeAllItems();
jComboBox4.addItem("Choose an Account");
jComboBox4.addItemListener(this);
jComboBox4.setVisible(true);
flag=2;
}

if ( event.getSource() == jRadioButton3&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    jTextField1.setVisible(false);
    jTextField3.setVisible(false);
    jLabel6.setVisible(false);

```

```

jComboBox5.setVisible(false);
jComboBox3.setVisible(false);
jComboBox4.removeItemListener(this);
jComboBox4.removeAllItems();
jComboBox4.addItem("Choose an Account");

System.out.println("custo "+tcustomer);
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;

Ssql = ("Select accountnum,aamount from accounts where
acustnum="+tcustomer+" ");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=sonuc.getString(1);
        accountamount=sonuc.getString(2);
        jComboBox4.addItem(s+" "+accountamount);
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox4.addItemListener(this);
jComboBox4.setVisible(true);
flag=3;
}

if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    tcustomer=jComboBox1.getSelectedItem().toString();
    System.out.println(tcustomer);
    jComboBox2.removeItemListener(this);
    jComboBox2.removeAllItems();
    jComboBox2.addItem("Choose an Account");
    jComboBox2.addItemListener(this);
}

```

```

    if (tcustomer=="Choose a Customer"){
        Owner First!";
        Ermes ermess=new Ermes("Customer is Missing","Please Choose an
        ermess.setLocation(210,210);
        ermess.setVisible(true);

    }else{
        tcustomer=tcustomer.substring(0,(tcustomer.indexOf(" ")));
        System.out.println("cust "+tcustomer);
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select accountnum,aamount from accounts where
        acustnum="+tcustomer+" ");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2));
                jComboBox2.addItem(s);
            }
            stm.close();
            con.close();
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
        }
    }
}

if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    taccount=jComboBox2.getSelectedItem().toString();
    System.out.println(taccount);
    if (taccount=="Choose an Account"){
        Account First!";
        Ermes ermess=new Ermes("Account is Missing","Please Choose an
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}

```



```

}else{
    taccount=taccount.substring(0,(taccount.indexOf(" ")));
    System.out.println("account combo2 listener: "+taccount);
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;

    Ssql = ("Select aamount from accounts where accountnum="+taccount+"
");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String s="";
        while (sonuc.next()){
            s=(sonuc.getString(1));
            jLabel8.setText(s);
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
}

if( event.getSource() == jComboBox3&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    rcustomer=jComboBox3.getSelectedItem().toString();
    //rcustomer=tcustomer;
    jComboBox4.removeItemListener(this);
    jComboBox4.removeAllItems();
    jComboBox4.addItem("Choose an Account");
    jComboBox4.addItemListener(this);
    if (rcustomer=="Choose a Customer"){
        Ermes ermess=new Ermes("Customer is Missing","Please Choose an
Owner First!");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}else{
    rcustomer=rcustomer.substring(0,(rcustomer.indexOf(" ")));

```



```

Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;

Ssql = ("Select accountnum,aamount from accounts where
acustnum="+rcustomer+" ");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=(sonuc.getString(1)+" "+sonuc.getString(2));
        jComboBox4.addItem(s);
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
}

if ( event.getSource() == jRadioButton4&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    jTextField1.setVisible(false);
    jComboBox3.setVisible(false);
    jComboBox4.setVisible(false);
    jLabel6.setVisible(true);
    jTextField3.setVisible(true);
    jComboBox5.removeItemListener(this);
    jComboBox5.removeAllItems();
    jComboBox5.addItem("Choose a Bank");
    jComboBox5.addItemListener(this);
    jComboBox5.setVisible(true);
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;

    Ssql = ("Select bankcode,bankname,country from banks");
    try {

```



```

jTextField2 = new javax.swing.JTextField();
jRadioButton4 = new javax.swing.JRadioButton();
jComboBox5 = new javax.swing.JComboBox();
jTextField3 = new javax.swing.JTextField();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Fund Transfer");
getContentPane().add(jLabel1);
jLabel1.setBounds(120, 10, 230, 17);

jLabel2.setText("Transiver Customer");
getContentPane().add(jLabel2);
jLabel2.setBounds(20, 40, 140, 14);

jLabel3.setText("Transiver Account");
getContentPane().add(jLabel3);
jLabel3.setBounds(20, 70, 140, 14);

jLabel4.setText("Receiver Options");
getContentPane().add(jLabel4);
jLabel4.setBounds(20, 170, 110, 14);

jRadioButton1.setText("To Name");
getContentPane().add(jRadioButton1);
jRadioButton1.setBounds(20, 190, 120, 23);

jRadioButton2.setText("To Another Customer");
getContentPane().add(jRadioButton2);
jRadioButton2.setBounds(20, 212, 180, 23);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(250, 40, 140, 22);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(250, 70, 140, 22);

getContentPane().add(jSeparator1);
jSeparator1.setBounds(20, 160, 370, 10);

getContentPane().add(jTextField1);
jTextField1.setBounds(250, 190, 140, 19);

getContentPane().add(jComboBox3);

```



```

jComboBox3.setBounds(250, 215, 140, 22);

jRadioButton3.setText("To Customer's Another Account");
getContentPane().add(jRadioButton3);
jRadioButton3.setBounds(20, 236, 210, 23);

getContentPane().add(jComboBox4);
jComboBox4.setBounds(250, 240, 140, 22);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(20, 320, 470, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(400, 40, 90, 270);

jLabel5.setText("Transfer Amount");
getContentPane().add(jLabel5);
jLabel5.setBounds(20, 130, 130, 14);

getContentPane().add(jTextField2);
jTextField2.setBounds(250, 130, 140, 19);

jRadioButton4.setText("To Another Bank");
getContentPane().add(jRadioButton4);
jRadioButton4.setBounds(20, 260, 220, 23);

getContentPane().add(jComboBox5);
jComboBox5.setBounds(250, 265, 140, 22);

getContentPane().add(jTextField3);
jTextField3.setBounds(250, 290, 140, 19);

jLabel6.setText("Receiver Name");
getContentPane().add(jLabel6);
jLabel6.setBounds(20, 290, 180, 14);

```



```

jLabel7.setText("Account Amount");
getContentPane().add(jLabel7);
jLabel7.setBounds(20, 100, 210, 14);

jLabel8.setText("Amount");
getContentPane().add(jLabel8);
jLabel8.setBounds(250, 100, 140, 14);

pack();
}
// </editor-fold>
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    if (amount==""){

    }else{
        if (isNumber(amount)==true){
            amountcheck=true;
            jTextField2.setText(amount);
        }else{
            amountcheck=false;
            Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive
Number For Amount !");
            ermess.setLocation(210,210);
            ermess.setVisible(true);
        }
    }
    if (amountcheck==true){
        if(Integer.parseInt(accountamount)>=Integer.parseInt(amount)){
            if (flag==1){
                name=jTextField1.getText().toUpperCase();
                if (name==""){

                }else{
                    if (isNumber(name)==true){
                        namecheck=true;
                    }else{
                        namecheck=false;
                        Ermes ermess=new Ermes("Name Wrong","Please Fill Receiver
Name !");
                        ermess.setLocation(210,210);

```

```

        ermess.setVisible(true);
    }
}
amount=jTextField2.getText();
System.out.println("TACCOUN :"+taccount);

if (taccount!=""){
    System.out.println("Account hasnt choosen");
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
fundtransfers(ftcustnum,ftaccountnum,frcustnm,famount,usr) VALUES(?,?,?,?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,taccount);
        pstmt.setString(3,name);
        pstmt.setString(4,amount);
        pstmt.setString(5,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");
        ermess.setVisible(true);
    }
}
//*****
*****

//Substracting amount from account
Statement stm;
String Ssql;
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

```

```

    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }

    try{
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        String pos=""+"YES"+"";
        System.out.println("ac :"+taccount);
        Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+taccount+" ";
        System.out.println(Ssql);
        stm.addBatch(Ssql);
        stm.executeBatch();

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
}

/*****
*****/

//Adding revenue to bank revenues
Connection connec;
String revenue;
if (Integer.parseInt(amount)<500){
    revenue="5";
}else{
    revenue="10";
}
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    PreparedStatement pstmts=connec.prepareStatement(
        "INSERT INTO
bankrevenues(revcustomer,revaccount,revamout,revdesc) VALUES(?,?,?,?)");
    pstmts.setString(1,tcustomer);
    pstmts.setString(2,taccount);

```



```

        pstmt.setString(3,revenue);
        pstmt.setString(4,"Fund Transfer revenue");
        pstmt.executeUpdate();
        pstmt.close();
        connec.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Creditedd","Crediting
Failed");

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
} else{
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        System.out.println("flag 1 -- nonaccount");
        System.out.println("ftcustnum :"+tcustomer);
        System.out.println("ftcustnm :"+name);
        System.out.println("famount :"+amount);
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO fundtransfers(ftcustnum,fcustnm,famount,usr)
VALUES(?,?,?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,name);
        pstmt.setString(3,amount);
        pstmt.setString(4, user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");

```



```

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }

/*****
*****/

    //Adding revenue to bank revenues
    Connection connec;
    String revenue;
    if (Integer.parseInt(amount)<500){
        revenue="5";
    }else{
        revenue="10";
    }
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmts=connec.prepareStatement(
            "INSERT INTO bankrevenues(revcustomer,revamout,revdesc)
VALUES(?,?,?)" );
        pstmts.setString(1,tcustomer);
        pstmts.setString(2,revenue);
        pstmts.setString(3,"Fund Transfer revenue");
        pstmts.executeUpdate();
        pstmts.close();
        connec.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Creditedd","Crediting
Failed");

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}

}

/*****
*****/

    if (flag==2){
        System.out.println(" ilk raccount :"+raccount);
        //rcustomer=jComboBox3.getSelectedItem().toString();
        raccount=jComboBox4.getSelectedItem().toString();
        //rcustomer=rcustomer.substring(0,(rcustomer.indexOf(" ")));

```

```

raccount=raccount.substring(0,(raccount.indexOf(" ")));
amount=jTextField2.getText();
System.out.println("flag 2");
System.out.println("tcustomer :"+tcustomer);
System.out.println("taccount :"+taccount);
System.out.println("rcustomer :"+rcustomer);
System.out.println("raccount :"+raccount);
System.out.println("amount :"+amount);
if (taccount!=""){
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
fundtransfers(ftcustnum,ftaccountnum,frcustnum,fraccountnum,famount,usr)
VALUES(?,?,?,?,?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,taccount);
        pstmt.setString(3,rcustomer);
        pstmt.setString(4,raccount);
        pstmt.setString(5,amount);
        pstmt.setString(6,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}
//*****
*****

//Subtracting amount from account

```

```

Statement stm;
String Ssql;
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}

try{
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    String pos=""+"YES"+"";

    Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+taccount+" ";
    stm.addBatch(Ssql);
    stm.executeBatch();

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}

//*****
*****

// Adding amount to Receiver's Account
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}

try{
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+raccount+" ";
    stm.addBatch(Ssql);
    stm.executeBatch();

    stm.close();
    con.close();
}

```



```

    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }

/*****
*****/

//Adding revenue to bank revenues
Connection connec;
String revenue;
if (Integer.parseInt(amount)<500){
    revenue="5";
}else{
    revenue="10";
}
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmts=connec.prepareStatement(
            "INSERT INTO
bankrevenues(revcustomer,revaccount,revamout,revdesc) VALUES(?,?,?,?)");
        pstmts.setString(1,tcustomer);
        pstmts.setString(2,taccount);
        pstmts.setString(3,revenue);
        pstmts.setString(4,"Fund Transfer revenue");
        pstmts.executeUpdate();
        pstmts.close();
        connec.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Creditedd","Crediting
Failed");

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}else{
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");

```



```

        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
fundtransfers(ftcustnum,frcustnum,fraccountnum,famount,usr) VALUES(?,?,?,?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,rcustomer);
        pstmt.setString(3,raccount);
        pstmt.setString(4,amount);
        pstmt.setString(5,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
//*****
*****

    // Adding amount to Receiver's Account
    Statement stm;
    String Ssql;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try{
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        Ssql="update accounts set aamount=aamount+" +amount+" where
accountnum="+raccount+" ";
        stm.addBatch(Ssql);

```

```

        stm.executeBatch();

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }

/*****
*****/

//Adding revenue to bank revenues
Connection connec;
String revenue;
if (Integer.parseInt(amount)<500){
    revenue="5";
}else{
    revenue="10";
}
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmts=connec.prepareStatement(
            "INSERT INTO bankrevenues(revcustomer,revamout,revdesc)
VALUES(?,?,?)");
        pstmts.setString(1,tcustomer);
        pstmts.setString(2,revenue);
        pstmts.setString(3,"Fund Transfer revenue");
        pstmts.executeUpdate();
        pstmts.close();
        connec.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Creditedd","Crediting
Failed");

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}
}
if (flag==3){
    rcustomer=tcustomer;
    raccout=jComboBox4.getSelectedItem().toString();
}

```

```

raccount=raccount.substring(0,(raccount.indexOf(" ")));
amount=jTextField2.getText();
if (taccount!=""){
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
fundtransfers(ftcustnum,ftaccountnum,frcustnum,fraccountnum,famount,usr)
VALUES(?,?,?,?,?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,taccount);
        pstmt.setString(3,tcustomer);
        pstmt.setString(4,raccount);
        pstmt.setString(5,amount);
        pstmt.setString(6,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}

//*****
*****

//Subtracting amount from account
Statement stm;
String Ssql;
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

```



```

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try{
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            String pos=""+"YES"+"";

            Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+taccount+" ";
            stm.addBatch(Ssql);
            stm.executeBatch();

            stm.close();
            con.close();
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
        }
    }
    /*******
    *****/
    // Adding amount to Receiver's Account
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try{
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+raccount+" ";
        stm.addBatch(Ssql);
        stm.executeBatch();

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    }else{
        Ermes ermess=new Ermes("Transiver Account is Missing","Fund
Transfer Failed");
    }

```



```

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}
if (flag==4){
    bank=jComboBox5.getSelectedItem().toString();
    bank=bank.substring(0,(bank.indexOf(" ")));
    name=jTextField3.getText().toUpperCase();
    amount=jTextField2.getText();
    String
filename="EMFABANK_swift_message_to_bank_code_"+bank+".txt";
    System.out.println("bank :"+bank);
    System.out.println("name :"+name);
    System.out.println("amount :"+amount);

    if (taccount!=""){
        Connection con;
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            PreparedStatement pstmt=con.prepareStatement(
                "INSERT INTO
fundtransfers(ftcustnum,ftaccountnum,frbank,frcustnm,famount,usr)
VALUES(?,?,?,?,?,?)");
            pstmt.setString(1,tcustomer);
            pstmt.setString(2,taccount);
            pstmt.setString(3,bank);
            pstmt.setString(4,name);
            pstmt.setString(5,amount);
            pstmt.setString(6,user);
            pstmt.executeUpdate();

            pstmt.close();
            con.close();
            jTextField1.setText("");
//*****
//Creating Swift's Text File
try {
    File swiftfile =new File(filename);
    FileOutputStream out = new FileOutputStream(filename);
    BufferedOutputStream bout = new BufferedOutputStream(out);
    PrintStream pout = new PrintStream(bout);
    pout.println("EMFA Bank");

```

```

        pout.println("Customer :"+tcustomer);
        pout.println("Account :"+taccount);
        pout.println("*****");
        pout.println("Receiver Bank Code:"+bank);
        pout.println("Receiver Customer:"+name);
        pout.println("Amount :"+amount+"TL");
        pout.close();
    } catch(IOException ioe) {
        System.out.println(ioe.toString());
        System.out.println("fdsfsdfs");
    }
}
//*****
        Ermes ermess=new Ermes("Fund Transferred Successfully ! ",
"Fund Transferred");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund
Transfer Failed");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}
//*****
*****
        //Subtracting amount from account
        Statement stm;
        String Ssql;
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try{
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            String pos="""+"YES"+""";

            Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+taccount+" ";
            stm.addBatch(Ssql);
            stm.executeBatch();

            stm.close();
            con.close();

```

```

    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }

/*****
*****/

//Adding revenue to bank revenues
Connection connec;
String revenue;
if (Integer.parseInt(amount)<500){
    revenue="5";
}else{
    revenue="10";
}
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmts=connec.prepareStatement(
            "INSERT INTO
bankrevenues(revcustomer,revaccount,revamout,revdesc) VALUES(?,?,?,?)");
        pstmts.setString(1,tcustomer);
        pstmts.setString(2,taccount);
        pstmts.setString(3,revenue);
        pstmts.setString(4,"Fund Transfer revenue");
        pstmts.executeUpdate();
        pstmts.close();
        connec.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Credited","Crediting
Failed");

        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}else{
    Connection con;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");

```



```

        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
fundtransfers(ftcustnum,frbank,frcustnm,famount,usr) VALUES(?,?,?, ?,?)");
        pstmt.setString(1,tcustomer);
        pstmt.setString(2,bank);
        pstmt.setString(3,name);
        pstmt.setString(4,amount);
        pstmt.setString(5,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");
    }
    /*******
    //Creating Swift's Text File
    try {
        File swiftfile =new File(filename);
        FileOutputStream out = new FileOutputStream(filename);
        BufferedOutputStream bout = new BufferedOutputStream(out);
        PrintStream pout = new PrintStream(bout);
        pout.println("EMFA Bank");
        pout.println("Customer :"+tcustomer);
        pout.println("*****");
        pout.println("Receiver Bank Code:"+bank);
        pout.println("Receiver Customer:"+name);
        pout.println("Amount :"+amount+"TL");
        pout.close();
    } catch(IOException ioe) {
        System.out.println(ioe.toString());
        System.out.println("fdsfsdfs");
    }

    /*******
    *****/
    //Adding revenue to bank revenues
    Connection connec;
    String revenue;
    if (Integer.parseInt(amount)<500){
        revenue="5";
    }else{
        revenue="10";
    }
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

```



```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Addorder().setVisible(true);
        }
    });
}
import java

```

```

// Variables declaration - do not modify
private javax.swing.ButtonGroup buttonGroup1;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JComboBox jComboBox3;
private javax.swing.JComboBox jComboBox4;
private javax.swing.JComboBox jComboBox5;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JRadioButton jRadioButton1;
private javax.swing.JRadioButton jRadioButton2;
private javax.swing.JRadioButton jRadioButton3;
private javax.swing.JRadioButton jRadioButton4;
private javax.swing.JSeparator jSeparator1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration

```

```

}

```

Country.Java

```
/*
 * Country.java
 *
 * Created on February 9, 2006, 8:27 PM
 */

package c;
import java.sql.*;
import javax.swing.*;

/**
 *
 * @author Cheyenne
 */
public class Country extends javax.swing.JFrame {
    String user=Usrpass.getuser();
    String id="";
    String name="";
    Boolean cidcheck=true;
    Boolean cnamecheck=true;
    int cid=0;
    int cname=0;
    /** Creates new form Country */
    public Country() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    }
}
```



```

jLabel1.setText("Create Country");
getContentPane().add(jLabel1);
jLabel1.setBounds(120, 14, 150, 20);

jLabel2.setText("Country ID");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 50, 70, 14);

jLabel3.setText("Country Name");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 80, 90, 14);

getContentPane().add(jTextField1);
jTextField1.setBounds(140, 50, 140, 19);

getContentPane().add(jTextField2);
jTextField2.setBounds(140, 80, 140, 19);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 110, 340, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(290, 50, 80, 50);

pack();
}
// </editor-fold>
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

    id=jTextField1.getText().toUpperCase();
    if (id==""){

    }else{
        if (isNumber(id)==true){
            cidcheck=true;
        }else{
            cidcheck=false;
            Ermes ermess=new Ermes("Country ID Wrong","Please Enter 2 Character
For Country ID !");
            ermess.setVisible(true);
        }
    }
    name=jTextField2.getText().toUpperCase();
    if (name==""){

    }else{
        if (isNumber(id)==true){
            cnamecheck=true;
        }else{
            cnamecheck=false;
            Ermes ermess=new Ermes("Country ID Wrong","Please Enter 2 Character
For Country ID !");
            ermess.setVisible(true);
        }
    }
    if((cidcheck==true)&&(cnamecheck==true)){
        Connection con;
        Connection ccon;
        Statement stm;
        ResultSet sonuc;
        String Ssql="";
        Ssql = ("Select countryid,countryname from country");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {

```

```

con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
stm = con.createStatement();
sonuc = stm.executeQuery(Ssql);
String idcheck="";
String namecheck="";

while (sonuc.next()){
    idcheck=sonuc.getString(1);
    namecheck=sonuc.getString(2);
    if (id.equals(idcheck)){
        cid=cid+1;
    }
    if (name.equals(namecheck)){
        cname=cname+1;
    }
}
stm.close();
con.close();
System.out.println(cid);
System.out.println(cname);
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
if (id!=""){
    if (name!=""){
        if (cid==0){
            if (cname==0){
                try {
                    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
                }

                catch(java.lang.ClassNotFoundException e) {
                    System.err.print("ClassNotFoundException: ");
                    System.err.print(e.getMessage());
                }
                try {
                    ccon = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                    PreparedStatement pstmt=ccon.prepareStatement(
                        "INSERT INTO country(countryid,countryname,usr)
VALUES(?,?,?)");
                    pstmt.setString(1,id);
                    pstmt.setString(2,name);
                    pstmt.setString(3,user);
                    pstmt.executeUpdate();

                    pstmt.close();
                    ccon.close();
                    jTextField1.setText("");

```



```

        Ermes ermess=new Ermes("Country Recorded Successfully ! ",
"Country Recorded");
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Bank Creation Failed", "Bank
Creation Failed");
        ermess.setVisible(true);
    }
    }else{
        Ermes ermess=new Ermes("Country Creation Failed", "Country
Name Has Recorded");
        ermess.setVisible(true);
    }
    }else{
        Ermes ermess=new Ermes("Country Creation Failed", "Country ID
Has Recorded");
        ermess.setVisible(true);
    }
    }else{
        Ermes ermess=new Ermes("Country Creation Failed", "Country Name is
Missing");
        ermess.setVisible(true);
    }
    }else{
        Ermes ermess=new Ermes("Country Creation Failed", "Country ID is
Missing");
        ermess.setVisible(true);
    }
    }
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Country().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JTextField jTextField1;

```

```

private javax.swing.JTextField jTextField2;
// End of variables declaration
* AddMoney.java
}

```

```

* Created on Feb 1, 2004
*/

```

```

package c;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
import java.util.*;

```

```

* @author
*/

```

```

public

```

```

    static

```

```

    static

```

```

    static

```

```

    static

```

```

    static

```

```

try {

```

```

    try {

```

```

        try {

```

```

            try {

```

```

                try {

```

```

                    try {

```

```

                        try {

```

Addmoney.Java

```
/*
 * Addmoney.java
 *
 * Created on February 12, 2006, 5:29 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Addmoney extends javax.swing.JFrame implements ItemListener {
    String customer="";
    String account="";
    String amount="";
    Boolean amountcheck=true;
    /** Creates new form Addmoney */
    public Addmoney() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Customer");
        jComboBox2.addItem("Choose an Account");
        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                jComboBox1.addItem(s);
            }
        }
    }
}
```



```

    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
jComboBox2.addItemListener(this);
}

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Customer Selection
        customer=jComboBox1.getSelectedItem().toString();
        jComboBox2.removeItemListener(this);
        jComboBox2.removeAllItems();
        jComboBox2.addItem("Choose an Account");
        jComboBox2.addItemListener(this);
        if (customer=="Choose a Customer"){
            Ermes ermess=new Ermes("Customer is Missing","Please Choose an
Owner First!");
            ermess.setVisible(true);

        }else{
            customer=customer.substring(0,(customer.indexOf(" ")));
            System.out.println("customer"+ customer);
            //Account Selection
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;
            String check="""+"YES"+""";
            System.out.println("customer"+customer);
            Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+" ");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stm = con.createStatement();
                sonuc = stm.executeQuery(Ssql);
                String s="";

```

```

        int count=0;
        while (sonuc.next()){
            s=(sonuc.getString(1)+" "+sonuc.getString(2));
            jComboBox2.addItem(s);
            count=count+1;
        }
        if (count==0){
            Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Customer!");
            ermess.setVisible(true);
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Account!");
        ermess.setVisible(true);
    }
}
}
if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    account=jComboBox2.getSelectedItem().toString();
    if (account=="Choose an Account"){
        Ermes ermess=new Ermes("Account Has not Chooosen", "Please Choose
an Account!");
        ermess.setVisible(true);
    }else{
        account=account.substring(0,(account.indexOf(" ")));
    }
}
}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

```

```

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jComboBox2 = new javax.swing.JComboBox();
    jLabel4 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
}

```

```

jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Add Money To An Account");
getContentPane().add(jLabel1);
jLabel1.setBounds(60, 10, 290, 17);

jLabel2.setText("Choose a Customer");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 40, 150, 14);

jLabel3.setText("Choose an Account");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 70, 140, 14);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(180, 40, 140, 22);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(180, 70, 140, 22);

jLabel4.setText("Amount");
getContentPane().add(jLabel4);
jLabel4.setBounds(30, 110, 140, 14);

getContentPane().add(jTextField1);
jTextField1.setBounds(180, 110, 140, 19);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 140, 380, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

```



```

getContentPane().add(jButton2);
jButton2.setBounds(330, 40, 80, 90);

pack();
}
// </editor-fold>
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    String user=Usrpass.getuser();
    amount=jTextField1.getText();
    if (amount==""){

    }else{
        if (isNumber(amount)==true){
            amountcheck=true;
        }else{
            amountcheck=false;
            Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive
Number For Amount !");
            ermess.setVisible(true);
        }
    }
    if (amountcheck==true){
        Statement stm;
        Connection con;
        String Ssql;
        if (account!=""){
            if (amount!=""){
                try {
                    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

                }

                catch(java.lang.ClassNotFoundException e) {
                    System.err.print("ClassNotFoundException: ");
                    System.err.print(e.getMessage());
                }
            }
            try{
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stm = con.createStatement();
            }
        }
    }
}

```

```

        Ssql="update accounts set aamount=aamount+" +amount+" where
accountnum="+account+" ";
        stm.addBatch(Ssql);
        stm.executeBatch();
        Ermes ermess=new Ermes("Amount Added To Account","Amount
Added!");
        ermess.setVisible(true);

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    //Adding Transaction To Account transactions
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
accounttransactions(atraccount,atamount,atype,usr) VALUES(?,?,?,?)");
        pstmt.setString(1,account);
        pstmt.setString(2,amount);
        pstmt.setString(3,"ADD");
        pstmt.setString(4, user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund Transfer
Failed");
        ermess.setVisible(true);
    }
    }else{
        Ermes ermess=new Ermes("Amount Has not Given","Please Give
Amount!");
        ermess.setVisible(true);
    }
    }else{

```

```

        Ermes ermess=new Ermes("Account Has not Chosen","Please Choose
an Account!");
        ermess.setVisible(true);

```

```

    }
}

}

package c;

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Addmoney().setVisible(true);
        }
    });
}
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JTextField jTextField1;
// End of variables declaration

```

```

}

```


Drawmoney.Java

```
/*
 * Drawmoney.java
 *
 * Created on February 12, 2006, 6:12 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Drawmoney extends javax.swing.JFrame implements ItemListener{
    String customer="";
    String account="";
    String amount="";
    Boolean amountcheck=true;
    /** Creates new form Drawmoney */
    public Drawmoney() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Customer");
        jComboBox2.addItem("Choose an Account");
        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
            }
        }
    }
}
```

```

        jComboBox1.addItem(s);
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
jComboBox2.addItemListener(this);
}

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Customer Selection
        customer=jComboBox1.getSelectedItem().toString();
        jComboBox2.removeItemListener(this);
        jComboBox2.removeAllItems();
        jComboBox2.addItem("Choose an Account");
        jComboBox2.addItemListener(this);
        if (customer=="Choose a Customer"){
            Ermes ermess=new Ermes("Customer is Missing","Please Choose an
Owner First!");
            ermess.setVisible(true);

        }else{
            customer=customer.substring(0,(customer.indexOf(" ")));
            System.out.println("customer"+ customer);
            //Account Selection
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;
            String check="""+"YES"+"";
            System.out.println("customer"+customer);
            Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+" ");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stm = con.createStatement();
                sonuc = stm.executeQuery(Ssql);
                String s="";

```

```

        int count=0;
        while (sonuc.next()){
            s=(sonuc.getString(1)+" "+sonuc.getString(2));
            jComboBox2.addItem(s);
            count=count+1;
        }
        if (count==0){
            Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Customer!");
            ermess.setVisible(true);
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Account!");
        ermess.setVisible(true);
    }
}

if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    account=jComboBox2.getSelectedItem().toString();
    if (account=="Choose an Account"){
        Ermes ermess=new Ermes("Account Has not Chosen", "Please Choose
an Account!");
        ermess.setVisible(true);
    }
    else{
        account=account.substring(0,(account.indexOf(" ")));
    }
}

}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jComboBox1 = new javax.swing.JComboBox();
    jComboBox2 = new javax.swing.JComboBox();
    jTextField1 = new javax.swing.JTextField();

```



```

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Draw Money From Account");
getContentPane().add(jLabel1);
jLabel1.setBounds(60, 4, 270, 20);

jLabel2.setText("Choose a Customer");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 40, 120, 14);

jLabel3.setText("Choose an Account");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 70, 130, 14);

jLabel4.setText("Amount");
getContentPane().add(jLabel4);
jLabel4.setBounds(30, 100, 120, 14);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 130, 350, 23);

jButton2.setText("Close");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(310, 40, 70, 80);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(170, 40, 130, 22);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(170, 70, 130, 22);

getContentPane().add(jTextField1);
jTextField1.setBounds(170, 100, 130, 19);

```

```

    pack();
}
// </editor-fold>

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    String user;
    user=Usrpass.getuser();
    System.out.println("user :"+user);
    amount=jTextField1.getText();
    if (amount==""){

    }else{
        if (isNumber(amount)==true){
            amountcheck=true;
        }else{
            amountcheck=false;
            Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive
Number For Amount !");
            ermess.setVisible(true);
        }
    }
    if (amountcheck==true){
        Statement stm;
        Connection con;
        String Ssql;
        if (account!=""){
            if (amount!=""){
                try {
                    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

                }

                catch(java.lang.ClassNotFoundException e) {
                    System.err.print("ClassNotFoundException: ");
                    System.err.print(e.getMessage());

```

```

    }
    try{
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        Ssql="update accounts set aamount=aamount-"+amount+" where
accountnum="+account+" ";
        stm.addBatch(Ssql);
        stm.executeBatch();
        Ermes ermess=new Ermes("Amount Draw From Account","Amount
Draw!");
        ermess.setVisible(true);

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    //Adding Transaction To Account transactions
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO
accounttransactions(atraccount,atamount,atype,usr) VALUES(?,?,?,?)");
        pstmt.setString(1,account);
        pstmt.setString(2,amount);
        pstmt.setString(3,"DRAW");
        pstmt.setString(4,user);
        pstmt.executeUpdate();

        pstmt.close();
        con.close();
        jTextField1.setText("");
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Fund Hasn't Transferred","Fund Transfer
Failed");
        ermess.setVisible(true);
    }
} else{
    Ermes ermess=new Ermes("Amount Has not Given","Please Give
Amount!");

```



```

        ermess.setVisible(true);

    }
    }else{
        Ermes ermess=new Ermes("Account Has not Chooosen","Please Choose
an Account!");
        ermess.setVisible(true);

    }
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Drawmoney().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JTextField jTextField1;
// End of variables declaration

}

```

Allrevenues.Java

```
/*
 * Allrevenues.java
 *
 * Created on February 15, 2006, 12:06 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Allrevenues extends javax.swing.JFrame {

    /** Creates new form Allrevenues */
    public Allrevenues() {
        initComponents();
        this.setLocation(50, 50);
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        //jTable1.setValueAt("Name", 0, 1);
        //jTable1.setValueAt("Surname", 0, 2);

        Ssql = ("Select  revenuedate,revcustomer,revaccount,revamout,revdesc from
bankrevenues order by revenuedate");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            int row=0;
            while (sonuc.next()){
                int column=0;
                jTable1.setValueAt(sonuc.getString(1),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(2),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(3),row, column);
                column++;
            }
        }
    }
}
```

```

        jTable1.setValueAt(sonuc.getString(4),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(5),row, column);
        row++;
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setText("All Bank Revenues");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(220, 10, 220, 17);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
            {null, null, null, null, null},
        }
    ));

```



```

getContentPane().add(jScrollPane1);
jScrollPane1.setBounds(10, 40, 570, 390);

jButton1.setText("Cancel");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(590, 40, 100, 390);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Allrevenues().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration
}

```

Revenuedate.Java

```
/*
 * Revenuedate.java
 *
 * Created on February 17, 2006, 12:29 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Revenuedate extends javax.swing.JFrame implements ItemListener{
    String day;
    String month;
    String year;
    String eday;
    String emonth;
    String eyear;
    String startdate;
    String enddate;
    /** Creates new form Revenuedate */
    public Revenuedate() {
        initComponents();
        this.setLocation(200, 200);
        //Start Date ilk Ay Combo dolumu
        jComboBox2.addItem("January");
        jComboBox2.addItem("February");
        jComboBox2.addItem("March");
        jComboBox2.addItem("April");
        jComboBox2.addItem("May");
        jComboBox2.addItem("Jun");
        jComboBox2.addItem("July");
        jComboBox2.addItem("August");
        jComboBox2.addItem("September");
        jComboBox2.addItem("October");
        jComboBox2.addItem("November");
        jComboBox2.addItem("December");
        //ilk Ocak dolumu ve seçimi
        for( int i = 1; i <= 31; i++ )
            jComboBox1.addItem( i );
        for( int i = 1900; i <= 2006; i++ )
            jComboBox3.addItem( i );
        day=jComboBox1.getSelectedItem().toString();
        month=jComboBox2.getSelectedItem().toString();
    }
}
```



```

year=jComboBox3.getSelectedItem().toString();
//Start Date ilk Ay Combo dolumu
jComboBox4.addItem("January");
jComboBox4.addItem("February");
jComboBox4.addItem("March");
jComboBox4.addItem("April");
jComboBox4.addItem("May");
jComboBox4.addItem("Jun");
jComboBox4.addItem("July");
jComboBox4.addItem("August");
jComboBox4.addItem("September");
jComboBox4.addItem("October");
jComboBox4.addItem("November");
jComboBox4.addItem("December");
//ilk Ocak dolumu ve seçimi
for( int i = 1; i <= 31; i++ )
    jComboBox5.addItem( i );
for( int i = 1900; i <= 2006; i++ )
    jComboBox6.addItem( i );
eday=jComboBox5.getSelectedItem().toString();
emonth=jComboBox4.getSelectedItem().toString();
eyear=jComboBox6.getSelectedItem().toString();
// Değişiklikleri izlemek
jComboBox1.addItemListener(this);
jComboBox2.addItemListener(this);
jComboBox3.addItemListener(this);
jComboBox4.addItemListener(this);
jComboBox5.addItemListener(this);
jComboBox6.addItemListener(this);
}

```

/** This method is called from within the constructor to

* initialize the form.

* WARNING: Do NOT modify this code. The content of this method is

* always regenerated by the Form Editor.

*/

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jComboBox2 = new javax.swing.JComboBox();
    jComboBox3 = new javax.swing.JComboBox();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jComboBox4 = new javax.swing.JComboBox();
    jComboBox5 = new javax.swing.JComboBox();
    jComboBox6 = new javax.swing.JComboBox();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();

```

```

jTable1 = new javax.swing.JTable();
//
getContentPane().setLayout(null);
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setText("Date Based Revenue Report");
getContentPane().add(jLabel1);
jLabel1.setBounds(100, 10, 250, 17);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(210, 40, 50, 22);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(100, 40, 110, 22);

getContentPane().add(jComboBox3);
jComboBox3.setBounds(260, 40, 60, 22);

jLabel2.setText("Starting Date");
getContentPane().add(jLabel2);
jLabel2.setBounds(10, 40, 90, 14);

jLabel3.setText("Ending Date");
getContentPane().add(jLabel3);
jLabel3.setBounds(10, 80, 90, 14);

getContentPane().add(jComboBox4);
jComboBox4.setBounds(100, 80, 110, 22);

getContentPane().add(jComboBox5);
jComboBox5.setBounds(210, 80, 50, 22);

getContentPane().add(jComboBox6);
jComboBox6.setBounds(260, 80, 60, 22);

jButton1.setText("Search");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(10, 110, 310, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

```

```
getContentPane().add(jButton2);
jButton2.setBounds(330, 40, 80, 90);
```

174

[illegible]


```

        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "Revenue Date", "Customer Number", "Account Number", "Revenue
Amount", "Type"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane1.setViewportView(jTable1);

getContentPane().add(jScrollPane1);
jScrollPane1.setBounds(12, 142, 490, 130);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
//Searching with Date
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;
Ssql = ("Select revenuedate,revcustomer,revaccount,revamout,revdesc from
bankrevenues where revenuedate>='"+startdate+"' and revenuedate<='"+enddate+"
");
System.out.println(Ssql);
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
}

```

```

try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    int row=0;
    int rowcount=jTable1.getRowCount();
    int columncount=jTable1.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable1.setValueAt("", rown,columnn);
        }
    }
    while (sonuc.next()){
        int column=0;
        jTable1.setValueAt(sonuc.getString(1),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(2),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(3),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(5),row, column);
        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}

}

public void itemStateChanged( ItemEvent event ) {
    //Start Date Combo Dolumu
    if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
        jComboBox1.removeItemListener(this);
        jComboBox1.removeAll();
        jComboBox1.repaint();
        jComboBox1.addItemListener(this);
        //Month days selection
        month=jComboBox2.getSelectedItem().toString();
        if
        ((month=="January")||(month=="March")||(month=="May")||(month=="July")||(month=
=="August")||(month=="October")||(month=="December")){
            for( int i = 1; i <= 31; i++ )
                jComboBox1.addItem( i );
        }
        if
        ((month=="April")||(month=="Jun")||(month=="September")||(month=="November")){
            for( int i = 1; i <= 30; i++ )

```

```

        jComboBox1.addItem( i );
    }
    if (month=="February"){
        for( int i = 1; i <= 28; i++ )
            jComboBox1.addItem( i );
    }
    //Years
    for( int i = 1900; i <= 2006; i++ )
        jComboBox3.addItem( i );
}
//Gun
if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    day=jComboBox1.getSelectedItem().toString();
}
//Start Date
if( event.getSource() == jComboBox3&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    year=jComboBox3.getSelectedItem().toString();
    startdate=day+"-"+month.substring(0, 3)+"-"+year;
}
if( event.getSource() == jComboBox4&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    jComboBox5.removeItemListener(this);
    jComboBox5.removeAll();
    jComboBox5.repaint();
    jComboBox5.addItemListener(this);
    //Month days selection
    emonth=jComboBox4.getSelectedItem().toString();
    if
((emonth=="January")||(emonth=="March")||(emonth=="May")||(emonth=="July")||(em
onth=="August")||(emonth=="October")||(emonth=="December")){
        for( int i = 1; i <= 31; i++ )
            jComboBox5.addItem( i );
    }
    if
((emonth=="April")||(emonth=="Jun")||(emonth=="September")||(emonth=="November
")){
        for( int i = 1; i <= 30; i++ )
            jComboBox5.addItem( i );
        }
    if (emonth=="February"){
        for( int i = 1; i <= 28; i++ )
            jComboBox5.addItem( i );
    }
    //Years
    for( int i = 1900; i <= 2006; i++ )
        jComboBox6.addItem( i );
}
//Gun

```



```

        if( event.getSource() == jComboBox5&& event.getStateChange() ==
ItemEvent.SELECTED ) {
            eday=jComboBox5.getSelectedItem().toString();
        }
        //Start Date
        if( event.getSource() == jComboBox6&& event.getStateChange() ==
ItemEvent.SELECTED ) {
            eyear=jComboBox6.getSelectedItem().toString();
            enddate=eday+"-"+emonth.substring(0, 3)+"-"+eyear;
        }
        System.out.println(startdate);
        System.out.println(enddate);
    }
    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        this.setVisible(false);
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Revenuedate().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JComboBox jComboBox1;
    private javax.swing.JComboBox jComboBox2;
    private javax.swing.JComboBox jComboBox3;
    private javax.swing.JComboBox jComboBox4;
    private javax.swing.JComboBox jComboBox5;
    private javax.swing.JComboBox jComboBox6;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTable jTable1;
    // End of variables declaration

```

Allcustomers.Java

```
/*
 * Allcustomers.java
 *
 * Created on February 14, 2006, 5:42 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Allcustomers extends javax.swing.JFrame {

    /** Creates new form Allcustomers */
    public Allcustomers() {
        initComponents();
        this.setLocation(200, 200);

        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        //jTable1.setValueAt("Name", 0, 1);
        //jTable1.setValueAt("Surname", 0, 2);

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC", "ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            int row=0;
            while (sonuc.next()){
                int column=0;
                jTable1.setValueAt(sonuc.getString(1),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(2),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(3),row, column);
                row++;
            }
        }
    }
}
```

```

    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

```

```

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

```

```

    getContentPane().setLayout(null);

```

```

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel1.setText("All Customers");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(110, 10, 170, 17);

```

```

    jTable1.setModel(new javax.swing.table.DefaultTableModel(

```

```

        new Object [][] {

```

```

            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null},

```


[illegible]


```

jButton1.setText("Cancel");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(350, 40, 80, 220);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Allcustomers().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration

}

```

Addcustomer.Java

```
/*
 * Addcustomer.java
 *
 * Created on January 2, 2006, 10:10 PM
 */

package c;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
import java.sql.*;

/**
 *
 * @author Cheyenne
 */
public class Addcustomer extends javax.swing.JFrame implements ItemListener{
    String num="";
    String name="";
    String surname="";
    String month="";
    String day="";
    String year="";
    String gender="";
    String phone="";
    String address="";
    String birthdate="";
    String user=Usrpass.getuser();
    /** Creates new form Addcustomer */
    public Addcustomer() {
        initComponents();
        this.setLocation(200, 200);
        //ilk Ay Combo dolumu

        jComboBox2.addItem("January");
        jComboBox2.addItem("February");
        jComboBox2.addItem("March");
        jComboBox2.addItem("April");
        jComboBox2.addItem("May");
        jComboBox2.addItem("Jun");
        jComboBox2.addItem("July");
        jComboBox2.addItem("August");
        jComboBox2.addItem("September");
        jComboBox2.addItem("October");
        jComboBox2.addItem("November");
        jComboBox2.addItem("December");
        jComboBox4.addItem("Male");
```



```

jComboBox4.addItem("Female");

//ilk Ocak dolumu ve seçimi
for( int i = 1; i <= 31; i++ )
    jComboBox1.addItem( i );
for( int i = 1900; i <= 1993; i++ )
    jComboBox3.addItem( i );
day=jComboBox1.getSelectedItem().toString();
month=jComboBox2.getSelectedItem().toString();
year=jComboBox3.getSelectedItem().toString();
birthdate=day+"-"+month.substring(0, 3)+"-"+year;

// Değişiklikleri izlemek
jComboBox2.addItemListener(this);
jComboBox4.addItemListener(this);
jComboBox3.addItemListener(this);
jComboBox1.addItemListener(this);
}

public void itemStateChanged( ItemEvent event ) {
    //Birthdate Combo Dolumu
    if( event.getSource() == jComboBox2 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Month days selection
        month=jComboBox2.getSelectedItem().toString();
        if
((month=="January")||(month=="March")||(month=="May")||(month=="July")||(month=
=="August")||(month=="October")||(month=="December")){
            jComboBox1.removeItemListener(this);
            jComboBox1.removeAllItems();
            for( int i = 1; i <= 31; i++ )
                jComboBox1.addItem( i );
            jComboBox1.addItemListener(this);
        }
        if
((month=="April")||(month=="Jun")||(month=="September")||(month=="November")){
            jComboBox1.removeItemListener(this);
            jComboBox1.removeAllItems();
            for( int i = 1; i <= 30; i++ )
                jComboBox1.addItem( i );
            jComboBox1.addItemListener(this);
        }
        if (month=="February"){
            jComboBox1.removeItemListener(this);
            jComboBox1.removeAllItems();
            for( int i = 1; i <= 28; i++ )
                jComboBox1.addItem( i );
            jComboBox1.addItemListener(this);
        }
    }
}

```

```

        //Years
        for( int i = 1900; i <= 1993; i++ )
            jComboBox3.addItem( i );
    }
    //Gun
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        day=jComboBox1.getSelectedItem().toString();
    }
    //Birthdate
    if( event.getSource() == jComboBox3 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        year=jComboBox3.getSelectedItem().toString();
        birthdate=day+"-"+month.substring(0, 3)+"-"+year;
        System.out.println(birthdate);
    }
    //Gender
    if( event.getSource() == jComboBox4 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        gender=jComboBox4.getSelectedItem().toString();
    }
}

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    jTextField3 = new javax.swing.JTextField();
    jTextField4 = new javax.swing.JTextField();
    jComboBox1 = new javax.swing.JComboBox();
    jComboBox2 = new javax.swing.JComboBox();
    jComboBox3 = new javax.swing.JComboBox();
    jButton1 = new javax.swing.JButton();
    jLabel7 = new javax.swing.JLabel();
    jComboBox4 = new javax.swing.JComboBox();
    jTextArea1 = new javax.swing.JTextArea();
    jButton2 = new javax.swing.JButton();
}

```

```

jLabel8 = new javax.swing.JLabel();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setText("Name");
getContentPane().add(jLabel1);
jLabel1.setBounds(10, 90, 80, 14);

jLabel2.setText("Surname");
getContentPane().add(jLabel2);
jLabel2.setBounds(10, 120, 70, 14);

jLabel3.setText("Number");
getContentPane().add(jLabel3);
jLabel3.setBounds(10, 60, 80, 14);

jLabel4.setText("Birthdate");
getContentPane().add(jLabel4);
jLabel4.setBounds(10, 180, 70, 14);

jLabel5.setText("Phone ");
getContentPane().add(jLabel5);
jLabel5.setBounds(10, 210, 50, 14);

jLabel6.setText("Address");
getContentPane().add(jLabel6);
jLabel6.setBounds(10, 240, 70, 14);

getContentPane().add(jTextField1);
jTextField1.setBounds(150, 60, 200, 19);

getContentPane().add(jTextField2);
jTextField2.setBounds(150, 90, 200, 19);

getContentPane().add(jTextField3);
jTextField3.setBounds(150, 120, 200, 19);

getContentPane().add(jTextField4);
jTextField4.setBounds(150, 210, 200, 19);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(250, 180, 40, 22);

jComboBox2.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(java.awt.event.ItemEvent evt) {
        jComboBox2.itemStateChanged(evt);
    }
});
jComboBox2.addActionListener(new java.awt.event.ActionListener() {

```



```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jComboBox2ActionPerformed(evt);
        }
    });
    jComboBox2.addPropertyChangeListener(new
java.beans.PropertyChangeListener() {
        public void propertyChange(java.beans.PropertyChangeEvent evt) {
            jComboBox2PropertyChange(evt);
        }
    });

    getContentPane().add(jComboBox2);
    jComboBox2.setBounds(150, 180, 100, 22);

    getContentPane().add(jComboBox3);
    jComboBox3.setBounds(290, 180, 60, 22);

    jButton1.setText("Save");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(10, 320, 420, 23);

    jLabel7.setText("Gender");
    getContentPane().add(jLabel7);
    jLabel7.setBounds(10, 150, 70, 14);

    getContentPane().add(jComboBox4);
    jComboBox4.setBounds(150, 150, 200, 22);

    getContentPane().add(jTextArea1);
    jTextArea1.setBounds(150, 240, 200, 70);

    jButton2.setText("Cancel");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton2);
    jButton2.setBounds(360, 60, 73, 250);

    jLabel8.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel8.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel8.setText("New Customer");

```

```

getContentPane().add(jLabel8);
jLabel8.setBounds(60, 20, 290, 17);

pack();
}
// </editor-fold>

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
this.setVisible(false);
}

private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Boolean numcheck=true;
    Boolean namecheck=true;
    Boolean surnamecheck=true;
    Boolean phonecheck=true;
    num=jTextField1.getText();
    if (isNumber(num)==true){
        numcheck=true;
    }else{
        numcheck=false;
        Ermes ermess=new Ermes("Customer Number Wrong","Please Enter
Numerical Value For Customer Number !");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
    name=jTextField2.getText().toUpperCase();
    if (isNumber(name)==false){
        namecheck=true;
    }else{
        namecheck=false;
        Ermes ermess=new Ermes("Customer Name Wrong","Please Enter
Character Value For Customer Name !");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
    surname=jTextField3.getText().toUpperCase();
    if (isNumber(surname)==false){
        surnamecheck=true;
    }
}

```

```

}else{
    surnamecheck=false;
    Ermes ermess=new Ermes("Customer Surname Wrong","Please Enter
Character Value For Customer Surname !");
    ermess.setLocation(210,210);
    ermess.setVisible(true);
}
phone=jTextField4.getText();
if (isNumber(phone)==true){
    phonecheck=true;
}else{
    phonecheck=false;
    Ermes ermess=new Ermes("Customer Phone Wrong","Please Enter
Numerical Value For Customer Phone !");
    ermess.setLocation(210,210);
    ermess.setVisible(true);
}
address=jTextArea1.getText().toUpperCase();
if ((numcheck==true) && (namecheck==true) && (surnamecheck==true) &&
(phonecheck==true)){
    //Querying Existent Customers
    Connection con;
    String Ssql="";
    Statement stm;
    ResultSet sonuc;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    Ssql="select custnum from customers where custnum="+num+" ";
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        int r=0;
        while (sonuc.next()){
            r=r+1;
        }
        if (r!=0){
            String message = "This Customer Number Has Been Recorded
Before\nPlease Check The Customer Number ";
            Ermes ermess=new Ermes(message);
            ermess.setLocation(210,210);
            ermess.setVisible(true);
            stm.close();
            con.close();

```



```

    }else{
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            PreparedStatement pstmt=con.prepareStatement(
                "INSERT INTO
customers(custnum,custnm,custsnm,custgender,custbirthdate,custphone,custaddres
s,usr) VALUES(?,?,?,?,?,?,?,?)");
            String one=jTextField1.getText();
            String two=jTextField2.getText().toUpperCase();
            String three=jTextField3.getText().toUpperCase();
            String four=jComboBox4.getSelectedItem().toString();
            String five=birthdate;
            String six=jTextField4.getText();
            String seven=jTextArea1.getText().toUpperCase();
            pstmt.setString(1,one);
            pstmt.setString(2,two);
            pstmt.setString(3, three);
            pstmt.setString(4,four);
            pstmt.setString(5,five);
            pstmt.setString(6,six);
            pstmt.setString(7,seven);
            pstmt.setString(8,user);
            pstmt.executeUpdate();

            pstmt.close();
            con.close();
            jTextField1.setText("");
            jTextField2.setText("");
            jTextField3.setText("");
            jTextField4.setText("");
            jTextArea1.setText("");
            String message = "This Customer Added Successfully ! ";
            String title= two+" "+three;
            Ermes ermess=new Ermes(message, title);
            ermess.setLocation(210,210);
            ermess.setVisible(true);
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
            Ermes ermess=new Ermes("Customer Could Not Created","Please
Check All Fields !");
            ermess.setLocation(210,210);
            ermess.setVisible(true);
        }
    }
} catch(SQLException ex) {
    Ermes ermess=new Ermes("Customer Could Not Created","Please Check
All Fields !");
    ermess.setLocation(210,210);
    ermess.setVisible(true);
}
}

```

```

    }else{
        Ermes ermess=new Ermes("Customer Could Not Created","Please Check All
Fields !");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}

```

```

private void jComboBox2PropertyChange(java.beans.PropertyChangeEvent evt) {
// TODO add your handling code here:
}

```

```

private void jComboBox2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
}

```

```

private void jComboBox2ItemStateChanged(java.awt.event.ItemEvent evt) {
// TODO add your handling code here:
}

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Addcustomer().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JComboBox jComboBox3;
private javax.swing.JComboBox jComboBox4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;

```

```

private javax.swing.JTextArea jTextArea1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
// End of variables declaration

```

```

}

```

```

try {

```

```

    con = DriverManager

```

```

    .getConnection(

```

```

        "jdbc:odbc:Driver={Microsoft

```

```

        Access Driver (*.mdb);

```

```

        DataSourceName=jdbc:odbc:Driver={Microsoft

```

```

        Access Driver (*.mdb);

```

```

        DataSourceName=jdbc:odbc:Driver={Microsoft

```


Customerreport.java

```
/*
 * Customerreport.java
 *
 * Created on February 14, 2006, 6:36 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Customerreport extends javax.swing.JFrame implements ItemListener {

    /** Creates new form Customerreport */
    public Customerreport() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Customer");
        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                jComboBox1.addItem(s);
            }

            stm.close();
        }
    }
}
```

```

        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    jLabel3.setVisible(false);
    jLabel4.setVisible(false);
    jComboBox1.addItemListener(this);
}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

```

```

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jButton1 = new javax.swing.JButton();

```

```

    getContentPane().setLayout(null);

```

```

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setText("Customer Report");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(130, 20, 170, 17);

```

```

    jLabel2.setText("Choose Customer");
    getContentPane().add(jLabel2);
    jLabel2.setBounds(30, 50, 120, 14);

```

```

    getContentPane().add(jComboBox1);
    jComboBox1.setBounds(170, 50, 150, 22);

```

```

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null},
            {null, null},
            {null, null},
            {null, null},
            {null, null},
            {null, null},
            {null, null},
            {null, null},

```

[illegible]

[illegible]

[illegible]


```

        public boolean isCellEditable(int rowIndex, int columnIndex) {
            return canEdit [columnIndex];
        }
    });
    jScrollPane1.setViewportView(jTable1);

    getContentPane().add(jScrollPane1);
    jScrollPane1.setBounds(30, 100, 390, 150);

    jLabel3.setText("Total Amount ");
    getContentPane().add(jLabel3);
    jLabel3.setBounds(30, 80, 130, 14);

    jLabel4.setText("jLabel4");
    getContentPane().add(jLabel4);
    jLabel4.setBounds(170, 80, 120, 14);

    jButton1.setText("Cancel");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(330, 50, 90, 40);

    pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}
public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Customer Selection
        String customer="";
        customer=jComboBox1.getSelectedItem().toString();
        if (customer=="Choose a Customer"){
            Ermes ermess=new Ermes("Customer is Missing","Please Choose an
Owner First!");
            ermess.setVisible(true);
        }else{
            customer=customer.substring(0,(customer.indexOf(" ")));
            System.out.println("customer"+ customer);
        }
    }
}

```

```

//Account Selection
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;
Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+" order by accountnum ");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    int count=0;
    int c=0;
    int amount=0;
    int rowcount=jTable1.getRowCount();
    int columncount=jTable1.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable1.setValueAt("", rown,columnn);
        }
    }
    int row=0;
    while (sonuc.next()){
        int column=0;
        jTable1.setValueAt(sonuc.getString(1),row, column);
        column++;
        count=Integer.parseInt(sonuc.getString(2));
        jTable1.setValueAt(String.valueOf(count),row, column);
        row++;
        amount=amount+count;
        c++;
    }
    jLabel3.setVisible(true);
    jLabel4.setVisible(true);
    jLabel4.setText(String.valueOf(amount));
    if (c==0){
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!","Please Check The Customer!");
        ermess.setVisible(true);
    }
    stm.close();

```

```

        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Account!");
        ermess.setVisible(true);
    }
}
}
}
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Customerreport().setVisible(true);
        }
    });
}

```

// Variables declaration - do not modify

```

private javax.swing.JButton jButton1;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration

```

```

}

```


Reportcustomer.Java

```
/*
 * Reportcustomer.java
 *
 * Created on February 14, 2006, 1:07 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Reportcustomer extends javax.swing.JFrame {

    /** Creates new form Reportcustomer */
    public Reportcustomer() {
        // setLocation((Toolkit.getDefaultToolkit().getScreenSize().width - getWidth()) /
2,
        // (Toolkit.getDefaultToolkit().getScreenSize().height - getHeight()) / 2);
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jTextField3 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Customer Report");
        getContentPane().add(jLabel1);
    }
}
```

```

jLabel1.setBounds(120, 4, 180, 20);

jLabel2.setText("Customer Name :");
getContentPane().add(jLabel2);
jLabel2.setBounds(10, 40, 100, 14);

jLabel3.setText("Customer Surname :");
getContentPane().add(jLabel3);
jLabel3.setBounds(10, 70, 110, 14);

jLabel4.setText("Customer Social Security Number");
getContentPane().add(jLabel4);
jLabel4.setBounds(10, 100, 190, 14);

getContentPane().add(jTextField1);
jTextField1.setBounds(210, 40, 110, 19);

getContentPane().add(jTextField2);
jTextField2.setBounds(210, 70, 110, 19);

getContentPane().add(jTextField3);
jTextField3.setBounds(210, 100, 110, 19);

jButton1.setText("Search");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(10, 130, 400, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(330, 40, 80, 80);

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null},
    }

```



```

boolean[] canEdit = new boolean [] {
    false, false, false
};

public boolean isCellEditable(int rowIndex, int columnIndex) {
    return canEdit [columnIndex];
}
});
jScrollPane1.setViewportView(jTable1);

getContentPane().add(jScrollPane1);
jScrollPane1.setBounds(10, 160, 400, 90);

pack();
}
// </editor-fold>

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
String ssn="";
String name="";
String surname="";
name=jTextField1.getText();
surname=jTextField2.getText();
ssn=jTextField3.getText();

if (ssn.equals("")){
    if (name.equals("")) {
        if (surname.equals("")){
            Ermes ermess=new Ermes("Search Failed!", "Please Fill Anyone of the
Fields!");
            ermess.setVisible(true);
        }else{
            //Searching with surname
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;
            Ssql = ("Select custnum,custnm,custsnm from customers where
custsnm="+surname+" ");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");

```

```

        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        //s=(sonuc.getString(1)+" "+sonuc.getString(2)+"
"+sonuc.getString(3));
        int row=0;
        int rowcount=jTable1.getRowCount();
        int columncount=jTable1.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable1.setValueAt("", rown,columnn);
            }
        }
        while (sonuc.next()){
            //s=(sonuc.getString(1)+" "+sonuc.getString(2)+"
"+sonuc.getString(3));
            int column=0;
            jTable1.setValueAt(sonuc.getString(1),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(2),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(3),row, column);
            row++;
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
}
}else{

//Searching with name
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;
Ssql = ("Select custnum,custnm,custsnm from customers where
custnm='"+name+"' ");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);

```

```

int row=0;
int rowcount=jTable1.getRowCount();
int columncount=jTable1.getColumnCount();
for (int rown = 0; rown < rowcount; rown++){
    for (int columnn = 0; columnn < columncount; columnn++){
        jTable1.setValueAt("", rown,columnn);
    }
}
while (sonuc.next()){
    int column=0;
    jTable1.setValueAt(sonuc.getString(1),row, column);
    column++;
    jTable1.setValueAt(sonuc.getString(2),row, column);
    column++;
    jTable1.setValueAt(sonuc.getString(3),row, column);
    row++;
}
stm.close();
con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
} else{

```

```

//Searching with ssn
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;
Ssql = ("Select custnum,custnm,custsnm from customers where
custnum='"+ssn+"' ");

```

```

try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

```

```

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}

```

```

try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    int rowcount=jTable1.getRowCount();
    int columncount=jTable1.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable1.setValueAt("", rown,columnn);
        }
    }
}

```



```

    }
    int row=0;
    while (sonuc.next()){
        int column=0;
        jTable1.setValueAt(sonuc.getString(1),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(2),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(3),row, column);
        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Reportcustomer().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration

}

```

Ermes.Java

```
/*
 * Ermes.java
 *
 * Created on January 4, 2006, 11:18 PM
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Ermes extends javax.swing.JFrame {

    /** Creates new form Ermes */
    public Ermes() {
        initComponents();
    }
    public Ermes(String x) {
        initComponents();
        this.setLocation(250, 250);
        this.setTitle(x);
        this.setSize(380,200);
        jLabel1.setText(x);
    }
    public Ermes(String x,String y) {
        initComponents();
        this.setTitle(y);
        this.setSize(380,200);
        jLabel1.setText(x);
        jLabel2.setText(y);
    }
    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jLabel2 = new javax.swing.JLabel();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Missing Field");
        getContentPane().add(jLabel1);
    }
}
```

```

jLabel1.setBounds(10, 10, 320, 30);

jButton1.setText("OK");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(120, 80, 110, 60);

jLabel2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel2.setText("Missing Field");
getContentPane().add(jLabel2);
jLabel2.setBounds(10, 40, 320, 30);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Ermes().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
// End of variables declaration

}

```


Openaccount.Java

```
/*
 * Openaccount.java
 *
 * Created on January 26, 2006, 1:53 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 *
 * @author Cheyenne
 */
public class Openaccount extends javax.swing.JFrame implements ItemListener {
    String user=Usrpass.getuser();
    String owner="Choose a Customer";
    /** Creates new form Openaccount */
    public Openaccount() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Customer");

        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                jComboBox1.addItem(s);
            }
        }
    }
}
```

```

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    jComboBox1.addItemListener(this);
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc="Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jLabel3 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel1.setText("Create New Account");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(100, 20, 200, 17);

    jLabel2.setText("Owner ");
    getContentPane().add(jLabel2);
    jLabel2.setBounds(20, 50, 60, 14);

    getContentPane().add(jComboBox1);
    jComboBox1.setBounds(110, 50, 180, 22);

    jLabel3.setText("Amount");
    getContentPane().add(jLabel3);
    jLabel3.setBounds(20, 90, 70, 14);

    jTextField1.setText("Enter Amount");
    getContentPane().add(jTextField1);
    jTextField1.setBounds(110, 90, 180, 19);

    jButton1.setText("Create Account");
    jButton1.addActionListener(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(20, 130, 270, 23);

    jButton2.setText("Cancel");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton2);
    jButton2.setBounds(300, 50, 80, 100);

    pack();
}
// </editor-fold>
public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        owner=jComboBox1.getSelectedItem().toString();
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String amount;
    Boolean amountcheck=true;
    Boolean ownercheck=true;
    amount=jTextField1.getText();
    if (amount.equals("Enter Amount")){
        amount="0";
    }
    if (isNumber(amount)==true){

```



```

        amountcheck=true;
    }else{
        amountcheck=false;
        Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive Number
For Amount !");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
    if (owner=="Choose a Customer"){
        Ermes ermess=new Ermes("Customer is Missing","Please Choose an Owner
First!");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
        ownercheck=false;
    }
    }else{
        owner=owner.substring(0,(owner.indexOf(" ")));
        ownercheck=true;
    }
    }

    if ((amountcheck==true)&&(ownercheck==true)){
        //Creating an Account
        Connection con;
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            PreparedStatement pstmt=con.prepareStatement(
                "INSERT INTO accounts(acustnum,aamount,usr,acredited)
VALUES(?,?,?,?)");
            String one=owner;
            String two=amount;

            pstmt.setString(1,one);
            pstmt.setString(2,two);
            pstmt.setString(3,user);
            pstmt.setString(4, "N");
            pstmt.executeUpdate();

            pstmt.close();
            con.close();
            jTextField1.setText("Enter Amount");

```

```

        Ermes ermess=new Ermes("This Account Created Successfully ! ",
"Account Created");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Account Hasn't Created","Account Create
Failed");
        ermess.setLocation(210,210);
        ermess.setVisible(true);
    }
}

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Openaccount().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JTextField jTextField1;
// End of variables declaration

}

```

Users.Java

```
/*
 * Users.java
 *
 * Created on February 13, 2006, 5:44 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;

/**
 *
 * @author Cheyenne
 */
public class Users extends javax.swing.JFrame implements ItemListener {
    private String name="";
    private String password="";
    private String type="";
    private String usrdesc="";
    Boolean usercheck=true;
    Boolean passwordcheck=true;
    /** Creates new form Users */
    public Users() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Type");
        jComboBox1.addItem("USER");
        jComboBox1.addItem("ADMIN");
        jComboBox1.addItemListener(this);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jComboBox1 = new javax.swing.JComboBox();
    }
}
```



```

jLabel5 = new javax.swing.JLabel();
jTextField3 = new javax.swing.JTextField();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Create User");
getContentPane().add(jLabel1);
jLabel1.setBounds(120, 20, 140, 17);

jLabel2.setText("User Name :");
getContentPane().add(jLabel2);
jLabel2.setBounds(30, 50, 120, 14);

jLabel3.setText("Password :");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 80, 110, 14);

jLabel4.setText("User Type :");
getContentPane().add(jLabel4);
jLabel4.setBounds(30, 110, 110, 14);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 170, 330, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(290, 50, 73, 110);

getContentPane().add(jTextField1);
jTextField1.setBounds(160, 50, 120, 19);

getContentPane().add(jTextField2);
jTextField2.setBounds(160, 80, 120, 19);

```

```

getContentPane().add(jComboBox1);
jComboBox1.setBounds(160, 110, 120, 22);

jLabel5.setText("User Description :");
getContentPane().add(jLabel5);
jLabel5.setBounds(30, 140, 120, 14);

getContentPane().add(jTextField3);
jTextField3.setBounds(160, 140, 120, 19);

pack();
}
// </editor-fold>
public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
        type=jComboBox1.getSelectedItem().toString();
        System.out.println("tip secildi :"+type);
    }
}

private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    System.out.println("tip :"+type);
    if (type.equals("")){
        type=jComboBox1.getSelectedItem().toString();
    }
    if (type.equals("Choose a Type")) {
        Ermes ermess=new Ermes("User Type Hasn't Chooosen! ", "Please Choose a
Type for User");
        ermess.setVisible(true);
    }else{
        name=jTextField1.getText().toUpperCase();
        if (name==""){

        }else{
            if (isNumber(name)==true){
                usercheck=false;
                Ermes ermess=new Ermes("Username Wrong","Please Enter
Username As Character!");
                ermess.setVisible(true);
            }else{

```

```

        usercheck=true;

    }
}
password=jTextField2.getText().toUpperCase();
if (password==""){
    Ermes ermess=new Ermes("Password Wrong","Please Fill Password
Field!");
    ermess.setVisible(true);
}else{
    passwordcheck=false;
}

usrdesc=jTextField3.getText().toUpperCase();
if((usercheck==true)&&(passwordcheck==true)){
    Connection con;
    Connection ccon;
    Statement stm;
    ResultSet sonuc;
    String Ssql="";
    int cname=0;
    Ssql = ("Select usnm from users");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String namecheck="";

        while (sonuc.next()){
            namecheck=sonuc.getString(1);
            if (name.equals(namecheck)){
                cname=cname+1;
            }
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    if (name.equals("")){
        Ermes ermess=new Ermes("User Creation Failed", "User Name is
Missing!");

```



```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Users().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration
}

```

Reports.Java

```
/*
 * Reports.java
 *
 * Created on February 13, 2006, 10:35 PM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Reports extends javax.swing.JFrame implements ActionListener{
    String selection;
    String[] fileItems = new String[] { "New", "Open", "Save", "Exit" };
    char[] fileShortcuts = { 'N','O','S','X' };
    /** Creates new form Reports */
    public Reports() {
        initComponents();

        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();
        jButton7 = new javax.swing.JButton();
        jButton11 = new javax.swing.JButton();

        getContentPane().setLayout(null);
```



```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Reports Menu");
getContentPane().add(jLabel1);
jLabel1.setBounds(120, 20, 170, 17);
```

```
jButton1.setText("Specific Customer Report");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton1);
jButton1.setBounds(30, 80, 190, 23);
```

```
jButton2.setText("All Customers");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton2);
jButton2.setBounds(30, 50, 190, 23);
```

```
jButton3.setText("Cancel");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton3);
jButton3.setBounds(230, 50, 80, 200);
```

```
jButton4.setText("All Accounts");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton4);
jButton4.setBounds(30, 140, 190, 23);
```

```
jButton5.setText("Specific Account Report");
jButton5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```

        jButton5ActionPerformed(evt);
    }
});

getContentPane().add(jButton5);
jButton5.setBounds(30, 170, 190, 23);

jButton6.setText("Customer's Balance Report");
jButton6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton6ActionPerformed(evt);
    }
});

getContentPane().add(jButton6);
jButton6.setBounds(30, 110, 190, 23);

jButton7.setText("All Fund Transfers");
jButton7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton7ActionPerformed(evt);
    }
});

getContentPane().add(jButton7);
jButton7.setBounds(30, 200, 190, 23);

jButton11.setText("Credit Reports");
jButton11.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton11ActionPerformed(evt);
    }
});

getContentPane().add(jButton11);
jButton11.setBounds(30, 230, 190, 23);

pack();
}
// </editor-fold>

private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reportcredit nreportcredit=new Reportcredit();
    nreportcredit.setTitle("Credit Report");
    nreportcredit.setSize(880,600);
    nreportcredit.setVisible(true);
}

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:
    Reportalltransfers nreportalltransfers=new Reportalltransfers();
    nreportalltransfers.setTitle("All Accounts");
    nreportalltransfers.setSize(820,620);
    nreportalltransfers.setVisible(true);
}

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Customerreport ncustomerreport=new Customerreport();
    ncustomerreport.setTitle("Customer Balance Report");
    ncustomerreport.setSize(480,330);
    ncustomerreport.setVisible(true);
}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reportaccount nreportaccount=new Reportaccount();
    nreportaccount.setTitle("Spesific Account Search");
    nreportaccount.setSize(480,330);
    nreportaccount.setVisible(true);
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Allaccounts nallaccounts=new Allaccounts();
    nallaccounts.setTitle("All Accounts");
    nallaccounts.setSize(480,330);
    nallaccounts.setVisible(true);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Allcustomers nallcustomers=new Allcustomers();
    nallcustomers.setTitle("All Customers");
    nallcustomers.setSize(470,330);
    nallcustomers.setVisible(true);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reportcustomer nreportcustomer=new Reportcustomer();
    nreportcustomer.setTitle("Spesific Customer Search");
    nreportcustomer.setSize(440,330);
}

```



```

nreportcustomer.setVisible(true);

}
public void actionPerformed(ActionEvent event) {
    System.out.println("Menu item [" + event.getActionCommand() +
        "] was pressed.");
}
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Reports().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton11;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JButton jButton6;
private javax.swing.JButton jButton7;
private javax.swing.JLabel jLabel1;
// End of variables declaration

```

```

}

```

Userreports.Java

```
/*
 * Userreports.java
 *
 * Created on February 20, 2000, 5:16 AM
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Userreports extends javax.swing.JFrame {

    /** Creates new form Userreports */
    public Userreports() {
        initComponents();
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        jLabel1.setText("User Reports");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(150, 10, 120, 20);

        jButton1.setText("Customer Report");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        getContentPane().add(jButton1);
        jButton1.setBounds(40, 40, 200, 23);
    }
}
```

```

jButton2.setText("Customer's Balance Report");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(40, 70, 200, 23);

jButton3.setText("Account Report");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(40, 100, 200, 23);

jButton4.setText("Cancel");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

getContentPane().add(jButton4);
jButton4.setBounds(260, 40, 65, 190);

pack();
}
// </editor-fold>

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reportaccount nreportaccount=new Reportaccount();
    nreportaccount.setTitle("Specific Account Search");
    nreportaccount.setSize(480,330);
    nreportaccount.setVisible(true);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```



```

Customerreport ncustomerreport=new Customerreport();
ncustomerreport.setTitle("Customer Balance Report");
ncustomerreport.setSize(480,330);
ncustomerreport.setVisible(true);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Reportcustomer nreportcustomer=new Reportcustomer();
    nreportcustomer.setTitle("Spesific Customer Search");
    nreportcustomer.setSize(440,330);
    nreportcustomer.setVisible(true);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Userreports().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
// End of variables declaration

}

```

Reportcredit.Java

```
/*
 * Reportcredit.java
 *
 * Created on February 20, 2000, 5:22 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Reportcredit extends javax.swing.JFrame implements ItemListener {
    String customer="Choose a Customer";
    String account="Choose an Account";
    /** Creates new form Reportcredit */
    public Reportcredit() {
        initComponents();
        this.setLocation(200, 200);
        buttonGroup1.add(jRadioButton1);
        buttonGroup1.add(jRadioButton2);
        buttonGroup1.add(jRadioButton3);
        jRadioButton1.addItemListener(this);
        jRadioButton2.addItemListener(this);
        jRadioButton3.addItemListener(this);
        jLabel2.setVisible(false);
        jLabel3.setVisible(false);
        jLabel4.setVisible(false);
        jLabel5.setVisible(false);
        jComboBox1.setVisible(false);
        jComboBox2.setVisible(false);
        jScrollPane1.setVisible(false);
        jScrollPane2.setVisible(false);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        buttonGroup1 = new javax.swing.ButtonGroup();
        jLabel1 = new javax.swing.JLabel();
        jRadioButton1 = new javax.swing.JRadioButton();
```

```

jRadioButton2 = new javax.swing.JRadioButton();
jRadioButton3 = new javax.swing.JRadioButton();
jComboBox1 = new javax.swing.JComboBox();
jLabel2 = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();
jComboBox2 = new javax.swing.JComboBox();
jScrollPane1 = new javax.swing.JScrollPane();
jTable1 = new javax.swing.JTable();
jScrollPane2 = new javax.swing.JScrollPane();
jTable2 = new javax.swing.JTable();
jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jButton1 = new javax.swing.JButton();

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setText("Credit Report");
getContentPane().add(jLabel1);
jLabel1.setBounds(140, 10, 120, 20);

jRadioButton1.setText("Only One Account Credit Situation");
getContentPane().add(jRadioButton1);
jRadioButton1.setBounds(50, 40, 230, 23);

jRadioButton2.setText("Open Credits");
getContentPane().add(jRadioButton2);
jRadioButton2.setBounds(50, 70, 150, 23);

jRadioButton3.setText("Closed Credits");
getContentPane().add(jRadioButton3);
jRadioButton3.setBounds(50, 100, 180, 23);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(410, 50, 140, 22);

jLabel2.setText("Choose Customer ");
getContentPane().add(jLabel2);
jLabel2.setBounds(280, 50, 130, 14);

jLabel3.setText("Choose Account");
getContentPane().add(jLabel3);
jLabel3.setBounds(280, 90, 130, 14);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(410, 90, 140, 22);

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

```



```

jButton1.setText("Cancel");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(560, 50, 90, 110);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}
public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jButton1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        int rowcount=jTable1.getRowCount();
        int columncount=jTable1.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable1.setValueAt("", rown,columnn);
            }
        }
        rowcount=jTable2.getRowCount();
        columncount=jTable2.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable2.setValueAt("", rown,columnn);
            }
        }
        jLabel2.setVisible(true);
        jLabel3.setVisible(true);
        jLabel4.setVisible(true);
        jLabel5.setVisible(true);
        jComboBox1.setVisible(true);
        jComboBox2.setVisible(true);
        jScrollPane1.setVisible(true);
        jScrollPane2.setVisible(true);
        //Filling the Combobox for owner
        jComboBox1.addItem("Choose a Customer");
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

```

```

Ssql = ("Select custnum,custnm,custsnm from customers");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
        jComboBox1.addItem(s);
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
}
if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    //Customer Selection
    customer=jComboBox1.getSelectedItem().toString();
    jComboBox2.removeItemListener(this);
    jComboBox2.removeAllItems();
    jComboBox2.addItem("Choose an Account");
    jComboBox2.addItemListener(this);
    if (customer=="Choose a Customer"){
        Owner First!";
        Ermes ermess=new Ermes("Customer is Missing","Please Choose an
        ermess.setVisible(true);

    }else{
        customer=customer.substring(0,(customer.indexOf(" ")));
        System.out.println("customer"+ customer);
        //Account Selection
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        String check=""+"YES"+"";
        System.out.println("customer"+customer);

```

```

        Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+"and acredited="+check+" ");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            int count=0;
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2));
                jComboBox2.addItem(s);
                count=count+1;
            }
            if (count==0){
                Ermes ermess=new Ermes("Customer Has Not Had Any Credited
Account!","Please Check The Customer!");
                ermess.setVisible(true);
            }
            stm.close();
            con.close();
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
            Ermes ermess=new Ermes("Customer Has Not Had Any
Account!","Please Check The Account!");
            ermess.setVisible(true);
        }
    }
}

if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    account=jComboBox2.getSelectedItem().toString();

    if (account=="Choose an Account"){
        Ermes ermess=new Ermes(" Account is Missing","Please Choose an
Account First!");
        ermess.setVisible(true);
    }else{
        account=account.substring(0,(account.indexOf(" ")));
        //Filling Table With Related Account Credit Payments
        Connection con;
        Statement stm;
    }
}

```



```

ResultSet sonuc;
String Ssql;
Ssql = ("Select custnum,accnum,paymentamount,paymentdate from
creditpayments where accnum='"+account+"' order by custnum,accnum");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    int row=0;
    int rowcount=jTable1.getRowCount();
    int columncount=jTable1.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable1.setValueAt("", rown,columnn);
        }
    }
    while (sonuc.next()){
        int column=0;
        jTable1.setValueAt(sonuc.getString(1),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(2),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(3),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        row++;
    }
    if (row==0){
        Ermes ermess=new Ermes(" Account Has Not Had Any
Payment","Account Has Not Had Credit Payment!");
        ermess.setVisible(true);
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
//Filling Table With Related Account Credit Payments
Connection ccon;
Statement cstm;
ResultSet csonuc;
String cSsql;

```

```

        cSsql = ("Select
canumber,caamount,ctype,totinsnum,insamount,insleft,amountleft from
creditedaccounts where canumber='"+account+"' order by canumber");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            ccon = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            cstm = ccon.createStatement();
            csonuc = cstm.executeQuery(cSsql);
            int row=0;
            int rowcount=jTable2.getRowCount();
            int columncount=jTable2.getColumnCount();
            for (int rown = 0; rown < rowcount; rown++){
                for (int columnn = 0; columnn < columncount; columnn++){
                    jTable2.setValueAt("", rown,columnn);
                }
            }
            while (csonuc.next()){
                int column=0;
                jTable2.setValueAt(csonuc.getString(1),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(2),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(3),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(4),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(5),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(6),row, column);
                column++;
                jTable2.setValueAt(csonuc.getString(7),row, column);
                row++;
            }
            cstm.close();
            ccon.close();
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
        }
    }
}

if( event.getSource() == jRadioButton2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    int rowcount=jTable2.getRowCount();

```

```

int columncount=jTable2.getColumnCount();
for (int rown = 0; rown < rowcount; rown++){
    for (int columnn = 0; columnn < columncount; columnn++){
        jTable2.setValueAt("", rown,columnn);
    }
}
jLabel2.setVisible(false);
jLabel3.setVisible(false);
jLabel4.setVisible(true);
jLabel5.setVisible(false);
jComboBox1.setVisible(false);
jComboBox2.setVisible(false);
jScrollPane1.setVisible(false);
jScrollPane2.setVisible(true);
//Filling Table With Credited Accounts
Connection con;
Statement stm;
ResultSet sonuc;
String Ssql;
Ssql = ("Select
canumber,caamount,ctype,totinsnum,insamount,insleft,amountleft from
creditedaccounts where creditclosed='"+N+"' order by canumber");
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}
try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    int row=0;
    rowcount=jTable2.getRowCount();
    columncount=jTable2.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable2.setValueAt("", rown,columnn);
        }
    }
    while (sonuc.next()){
        int column=0;
        jTable2.setValueAt(sonuc.getString(1),row, column);
        column++;
        jTable2.setValueAt(sonuc.getString(2),row, column);
        column++;
        jTable2.setValueAt(sonuc.getString(3),row, column);
        column++;
    }
}

```



```

        jTable2.setValueAt(sonuc.getString(4),row, column);
        columnn++;
        jTable2.setValueAt(sonuc.getString(5),row, column);
        columnn++;
        jTable2.setValueAt(sonuc.getString(6),row, column);
        columnn++;
        jTable2.setValueAt(sonuc.getString(7),row, column);
        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
if( event.getSource() == jButton3 && event.getStateChange() ==
ItemEvent.SELECTED ) {
    int rowcount=jTable2.getRowCount();
    int columncount=jTable2.getColumnCount();
    for (int rown = 0; rown < rowcount; rown++){
        for (int columnn = 0; columnn < columncount; columnn++){
            jTable2.setValueAt("", rown,columnn);
        }
    }
    jLabel2.setVisible(false);
    jLabel3.setVisible(false);
    jLabel4.setVisible(true);
    jLabel5.setVisible(false);
    jComboBox1.setVisible(false);
    jComboBox2.setVisible(false);
    jScrollPane1.setVisible(false);
    jScrollPane2.setVisible(true);
    //Filling Table With Credited Accounts
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;
    Ssql = ("Select
canumber,caamount,ctype,totinsnum,insamount,insleft,amountleft from
creditedaccounts where creditclosed='"+Y+"' order by canumber");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");

```

```

private stm = con.createStatement();
private sonuc = stm.executeQuery(Ssql);
private int row=0;
private rowcount=jTable2.getRowCount();
private columncount=jTable2.getColumnCount();
private for (int rown = 0; rown < rowcount; rown++){
private     for (int columnn = 0; columnn < columncount; columnn++){
private         jTable2.setValueAt("", rown,columnn);
private     }
private }
private while (sonuc.next()){
private     int column=0;
private     jTable2.setValueAt(sonuc.getString(1),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(2),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(3),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(4),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(5),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(6),row, column);
private     column++;
private     jTable2.setValueAt(sonuc.getString(7),row, column);
private     row++;
private }
private stm.close();
private con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
}
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Reportcredit().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.ButtonGroup buttonGroup1;
private javax.swing.JButton jButton1;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;

```

```

private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JRadioButton jButton1;
private javax.swing.JRadioButton jButton2;
private javax.swing.JRadioButton jButton3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTable jTable1;
private javax.swing.JTable jTable2;
// End of variables declaration
}

```


Creditopening.Java

```
/*
 * Creditopening.java
 *
 * Created on January 28, 2006, 10:06 PM
 */
package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
import java.math.*;

/**
 *
 * @author Cheyenne
 */
public class Creditopening extends javax.swing.JFrame implements ItemListener{
    String user=Usrpass.getuser();
    String customer="Choose a Customer";
    String account="Choose an Account";
    String type="Choose a Credit Type";
    String revenue="0";
    String amount="0";
    String totinsnum="0";
    String insamount="0";
    String ratio="0";
    Boolean amountcheck=true;
    /** Creates new form Creditopening */
    public Creditopening() {
        initComponents();
        this.setLocation(200, 200);
        this.setSize(400, 400);
        jComboBox1.addItem("Choose a Customer");
        jComboBox2.addItem("Choose an Account");
        jComboBox3.addItem("Choose a Credit Type");

        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
        }
    }
}
```

```

        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String s="";
        while (sonuc.next()){
            s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
            jComboBox1.addItem(s);
        }

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }

    //Fillin the Type Of Credit

    Ssql = ("Select TYPEID,type,ratio from creditypes");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String s="";
        while (sonuc.next()){
            s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+"%"+sonuc.getString(3));
            jComboBox3.addItem(s);
        }

        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    jComboBox1.addItemListener(this);
    jComboBox2.addItemListener(this);
    jComboBox3.addItemListener(this);
}

```

```

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Customer Selection
        customer=jComboBox1.getSelectedItem().toString();
        jComboBox2.removeItemListener(this);
        jComboBox2.removeAllItems();
        jComboBox2.addItem("Choose an Account");
        jComboBox2.addItemListener(this);
        if (customer=="Choose a Customer"){
            Owner Ermes ermess=new Ermes("Customer is Missing","Please Choose an
Owner First!");
            ermess.setVisible(true);

        }else{
            customer=customer.substring(0,(customer.indexOf(" ")));
            System.out.println("customer"+ customer);
            //Account Selection
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;

            Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+" ");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stm = con.createStatement();
                sonuc = stm.executeQuery(Ssql);
                String s="";
                while (sonuc.next()){
                    s=(sonuc.getString(1)+" "+sonuc.getString(2));
                    jComboBox2.addItem(s);
                }

                stm.close();
                con.close();
            } catch(SQLException ex) {
                System.err.print("SQLException: "+ex.getMessage());
                Ermes ermess=new Ermes("Customer Has Not Had Any
Account!","Please Check The Customer!");
                ermess.setVisible(true);
            }
        }
    }
}

```



```

    }

}

//account Selection
if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    account=jComboBox2.getSelectedItemAt().toString();
    if (account=="Choose an Account"){
        Ermes ermess=new Ermes("Account Has not Chooosen","Please Choose
an Account!");
        ermess.setVisible(true);

    }else{
        account=account.substring(0,(account.indexOf(" ")));
        System.out.println("account"+account);
    }
}

//Type Selection
if( event.getSource() == jComboBox3&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    type=jComboBox3.getSelectedItemAt().toString();
    if (type=="Choose a Credit Type"){
        Ermes ermess=new Ermes("Credit Type Has not Chooosen","Please
Choose an Credit Type!");
        ermess.setVisible(true);

    }else{
        type=type.substring(0,(type.indexOf(" ")));
        System.out.println("type"+type);

        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select maxinsnumber, ratio from creditypes where typeid="+type+"
");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
    }
}

```

```

try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    String s="";
    while (sonuc.next()){
        s=(sonuc.getString(1));
        ratio=sonuc.getString(2);
        jLabel7.setText(s);
    }
    totinsnum=s;

    int monthlpayment=0;
    amount=jTextField1.getText();
    if (amount==""){
        Number For Amount !");
        Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive
        ermess.setVisible(true);
    }else{
        if (isNumber(amount)==true){
            amountcheck=true;
        }else{
            amountcheck=false;
            Positive Number For Amount !");
            Ermes ermess=new Ermes("Amount Wrong","Please Enter
            ermess.setVisible(true);
        }
        if (amountcheck==true){
            revenue=String.valueOf((Integer.parseInt(amount)/100)*Integer.parseInt(ratio));
            amount=String.valueOf(Integer.parseInt(amount)+Integer.parseInt(revenue));
            monthlpayment=Math.round(Integer.parseInt(amount)/Integer.parseInt(s));

            insamount=String.valueOf(monthlpayment);
            jLabel9.setText(String.valueOf(monthlpayment));
            jLabel11.setText(amount);
            stm.close();
            con.close();
        }
    }
} catch (SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
    Ermes ermess=new Ermes("Customer Has Not Had Any
    Account!","Please Check The Customer!");
    ermess.setVisible(true);
}

```

```

    }
}

}
/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jLabel5 = new javax.swing.JLabel();
    jComboBox2 = new javax.swing.JComboBox();
    jTextField1 = new javax.swing.JTextField();
    jComboBox3 = new javax.swing.JComboBox();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jLabel6 = new javax.swing.JLabel();
    jLabel7 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
    jLabel10 = new javax.swing.JLabel();
    jLabel11 = new javax.swing.JLabel();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel1.setText("Credit Opening");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(140, 20, 160, 20);

    jLabel2.setText("Account Number");
    getContentPane().add(jLabel2);
    jLabel2.setBounds(10, 90, 130, 14);

    jLabel3.setText("Credit Amount");
    getContentPane().add(jLabel3);
    jLabel3.setBounds(10, 120, 120, 14);

    jLabel4.setText("Credit Type");
    getContentPane().add(jLabel4);
    jLabel4.setBounds(10, 150, 100, 14);

```



```

getContentPane().add(jComboBox1);
jComboBox1.setBounds(150, 60, 150, 22);

jLabel5.setText("Customer");
getContentPane().add(jLabel5);
jLabel5.setBounds(10, 60, 110, 14);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(150, 90, 150, 22);

getContentPane().add(jTextField1);
jTextField1.setBounds(150, 120, 150, 19);

getContentPane().add(jComboBox3);
jComboBox3.setBounds(150, 150, 150, 22);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(10, 250, 370, 23);

jButton2.setText("Close");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(310, 60, 70, 160);

jLabel6.setText("Maximum Instalment Number");
getContentPane().add(jLabel6);
jLabel6.setBounds(10, 180, 180, 14);

jLabel7.setText("Installment Number");
getContentPane().add(jLabel7);
jLabel7.setBounds(200, 180, 110, 14);

jLabel8.setText("Monthly Payment");
getContentPane().add(jLabel8);
jLabel8.setBounds(10, 200, 140, 14);

jLabel9.setText("Payment Amount");

```

```
getContentPane().add(jLabel9);
jLabel9.setBounds(200, 200, 110, 14);
```

```
jLabel10.setText("Total Payment");
getContentPane().add(jLabel10);
jLabel10.setBounds(10, 220, 160, 14);
```

```
jLabel11.setText("Payment");
getContentPane().add(jLabel11);
jLabel11.setBounds(200, 220, 110, 14);
```

```
pack();
}
// </editor-fold>
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
```

```
    this.setVisible(false);
}
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
```

```
    if (amountcheck==true){
        System.out.println("buton calisti");
        Connection con;
        Connection conn;
        String Ssql="";
        Statement stm;
        Statement stmm;
        ResultSet sonuc;
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }
```

```
        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
```

```
        Ssql="select ACREDITED from accounts where accountnum="+account+" ";
        System.out.println(Ssql);
```

```

try {
    con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    stm = con.createStatement();
    sonuc = stm.executeQuery(Ssql);
    System.out.println("dwewe");
    String acceptance="";
    int r=0;
    while (sonuc.next()){
        acceptance=sonuc.getString(1);
        r=r+1;
    }
    System.out.println("acceptance : "+acceptance);
    System.out.println(account);
    System.out.println(r);
    if (acceptance.equals("YES")){
        System.out.println("olu");
        Ermes ermess=new Ermes("This Customer Has Been Credited With
This Account Number","Please Change The Account Number ");
        ermess.setVisible(true);
        stm.close();
        con.close();
    }else{
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            PreparedStatement pstmt=con.prepareStatement(
                "INSERT INTO
creditedaccounts(canumber,caamount,ctype,creditclosed,totinsnum,insleft,insamount
,amountleft,usr) VALUES(?,?,?,?,?,?,?,?,?)");
            String one=account;
            String two=amount;
            String three=type;
            String four="N";
            String five=totinsnum;
            String six=totinsnum;
            String seven=insamount;
            String eight=amount;

            pstmt.setString(1,one);
            pstmt.setString(2,two);
            pstmt.setString(3,three);
            pstmt.setString(4,four);
            pstmt.setString(5,five);
            pstmt.setString(6,six);
            pstmt.setString(7,seven);
            pstmt.setString(8,eight);
            pstmt.setString(9,user);
            pstmt.executeUpdate();

            pstmt.close();
            con.close();
        }
    }
}

```



```

jTextField1.setText("");
Ermes ermess=new Ermes("Credit Assigned To Account Successfully
! ", "Credit Assigned");
ermess.setVisible(true);

// Marking Account As Credited
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());

}

try{
    conn = DriverManager.getConnection("jdbc:odbc:FC", "ff", "f");
    stmm = conn.createStatement();
    String pos=""+"YES"+"";

    Ssql="update accounts set acredited="+pos+" where
accountnum="+account+" ";
    stmm.addBatch(Ssql);
    stmm.executeBatch();

    stmm.close();
    conn.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}

} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
    Ermes ermess=new Ermes("Credit Assigning To Account Failed !
", "Check Related Fields");
    ermess.setVisible(true);
}

```

```

/*****
*****/

```

```

//Adding revenue to bank revenues
Connection connec;
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}

catch(java.lang.ClassNotFoundException e) {
    System.err.print("ClassNotFoundException: ");
    System.err.print(e.getMessage());
}

```

```

try {
    connec = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
    PreparedStatement pstmts=connec.prepareStatement(
        "INSERT INTO
bankrevenues(revcustomer,revaccount,revamout,revdesc) VALUES(?,?,?,?)");
    pstmts.setString(1,customer);
    pstmts.setString(2,account);
    pstmts.setString(3,revenue);
    pstmts.setString(4,"Credit revenue");
    pstmts.executeUpdate();
    pstmts.close();
    connec.close();
    jTextField1.setText("");
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
    Ermes ermess=new Ermes("Fund Hasn't Creditedd","Crediting
Failed");
    ermess.setVisible(true);
}
} catch(SQLException ex) {
}
}
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Creditopening().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JComboBox jComboBox3;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;

```

```
private javax.swing.JLabel jLabel7;  
private javax.swing.JLabel jLabel8;  
private javax.swing.JLabel jLabel9;  
private javax.swing.JTextField jTextField1;  
// End of variables declaration
```

```
}
```


Accountmenu.Java

```
/*
 * Accountmenu.java
 *
 * Created on February 20, 2000, 11:43 PM
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Accountmenu extends javax.swing.JFrame {

    /** Creates new form Accountmenu */
    public Accountmenu() {
        initComponents();
        this.setLocation(200, 200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jButton5 = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();

        jButton5.setText("jButton5");

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 12));
        jLabel1.setText("Account Menu");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(130, 10, 100, 20);

        jButton1.setText("Create Account");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        })
    }
}
```

```

});

getContentPane().add(jButton1);
jButton1.setBounds(50, 50, 220, 23);

jButton2.setText("Add Money To Account");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(50, 80, 220, 23);

jButton3.setText("Draw Money From Account");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(50, 110, 220, 23);

jButton4.setText("Cancel");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

getContentPane().add(jButton4);
jButton4.setBounds(280, 50, 80, 80);

pack();
}
// </editor-fold>

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Drawmoney ndrawmoney= new Drawmoney();
    ndrawmoney.setTitle("Draw Money From Account");
    ndrawmoney.setSize(480, 250);
    ndrawmoney.setVisible(true);
}

```

```

    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        Addmoney naddmoney= new Addmoney();
        naddmoney.setTitle("Add Money To an Account");
        naddmoney.setSize(480, 250);
        naddmoney.setVisible(true);
    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        Openaccount nopenaccount= new Openaccount();
        nopenaccount.setTitle("Create New Account");
        nopenaccount.setSize(450, 210);
        nopenaccount.setVisible(true);
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Accountmenu().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JButton jButton5;
    private javax.swing.JLabel jLabel1;
    // End of variables declaration

}

```


Creditmenu.Java

```
/*
 * Creditmenu.java
 *
 * Created on February 20, 2000, 11:54 PM
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Creditmenu extends javax.swing.JFrame {

    /** Creates new form Creditmenu */
    public Creditmenu() {
        initComponents();
        this.setLocation(200,200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 12));
        jLabel1.setText("Credit Menu");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(130, 10, 110, 20);

        jButton1.setText("Open Credit");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        getContentPane().add(jButton1);
        jButton1.setBounds(50, 50, 180, 23);
    }
}
```

```

jButton2.setText("Credit Payment");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(50, 80, 180, 23);

jButton3.setText("Cancel");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

getContentPane().add(jButton3);
jButton3.setBounds(240, 50, 90, 50);

pack();
}
// </editor-fold>

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Creditpayments cpay=new Creditpayments();
    cpay.setTitle("Credit Payments");
    cpay.setSize(470, 350);
    cpay.setVisible(true);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Creditopening copen=new Creditopening();
    copen.setTitle("Credit Opening");
    copen.setSize(400, 400);
    copen.setVisible(true);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

```

```

java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new Creditmenu().setVisible(true);
    }
});
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JLabel jLabel1;
// End of variables declaration

```

```

}

```


Managerreport.Java

```
/*
 * Managerreport.java
 *
 * Created on February 21, 2000, 2:24 AM
 */

package c;

/**
 *
 * @author Cheyenne
 */
public class Managerreport extends javax.swing.JFrame {

    /** Creates new form Managerreport */
    public Managerreport() {
        initComponents();
        this.setLocation(200,200);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
    private void initComponents() {
        jLabel1 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();

        getContentPane().setLayout(null);

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 12));
        jLabel1.setText("Managerial Reports");
        getContentPane().add(jLabel1);
        jLabel1.setBounds(120, 10, 200, 20);

        jButton1.setText("All Bank Revenues");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        getContentPane().add(jButton1);
    }
}
```

```
jButton1.setBounds(30, 40, 240, 23);
```

```
jButton2.setText("Spesific Revenue Report");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton2);
jButton2.setBounds(30, 70, 240, 23);
```

```
jButton3.setText("Date Based Revenue Report");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton3);
jButton3.setBounds(30, 100, 240, 23);
```

```
jButton4.setText("Cancel");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});
```

```
getContentPane().add(jButton4);
jButton4.setBounds(280, 40, 90, 80);
```

```
pack();
```

```
}
```

```
// </editor-fold>
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Revenuedate nrevenuedate=new Revenuedate();
    nrevenuedate.setTitle("Date based Revenue Report");
    nrevenuedate.setSize(520,350);
    nrevenuedate.setVisible(true);
}
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    Revenuereport nrevenuereport=new Revenuereport();
    nrevenuereport.setTitle("Spesific Revenue Report");
    nrevenuereport.setSize(520,350);
    nrevenuereport.setVisible(true);
}
```

```

    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        Allrevenues nallrevenues=new Allrevenues();
        nallrevenues.setTitle("All Revenues");
        nallrevenues.setSize(820,600);
        nallrevenues.setVisible(true);
    }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        this.setVisible(false);
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Managerreport().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JLabel jLabel1;
    // End of variables declaration
}

```


Banks.Java

```
/*
 * Banks.java
 *
 * Created on February 9, 2006, 4:11 PM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Banks extends javax.swing.JFrame implements ItemListener{
    String country="";
    int ccode=0;
    int cname=0;
    String user=Usrpass.getuser();
    Boolean bcodecheck=true;
    /** Creates new form Banks */
    public Banks() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Country");
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select countryid,countryname from country");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {

            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2));
                jComboBox1.addItem(s);
            }
        }
    }
}
```

```

    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
}

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        country=jComboBox1.getSelectedItem().toString();
        country=country.substring(0,(country.indexOf(" ")));
    }
}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

```

```

// <editor-fold defaultstate="collapsed" desc=" Generated Code ">

```

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jLabel3 = new javax.swing.JLabel();
    jTextField2 = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel1.setText("Create Bank");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(130, 20, 110, 17);

    jLabel2.setText("Bank Code");
    getContentPane().add(jLabel2);
    jLabel2.setBounds(30, 60, 110, 14);

    getContentPane().add(jTextField1);
    jTextField1.setBounds(160, 60, 150, 19);
}

```

```

jLabel3.setText("Bank Name");
getContentPane().add(jLabel3);
jLabel3.setBounds(30, 90, 110, 14);

getContentPane().add(jTextField2);
jTextField2.setBounds(160, 90, 150, 19);

jLabel4.setText("Country");
getContentPane().add(jLabel4);
jLabel4.setBounds(30, 120, 110, 14);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(160, 120, 150, 22);

jButton1.setText("Save");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(30, 150, 370, 23);

jButton2.setText("Cancel");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(320, 60, 80, 80);

pack();
}
// </editor-fold>
private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    String code=jTextField1.getText().toUpperCase();

```



```

if (code==""){
    Ermes erness=new Ermes("Bank Code Wrong","Please Enter Positive
Number For Bank Code !");
    erness.setVisible(true);
}else{
    if (isNumber(code)==true){
        bcodecheck=true;
    }else{
        bcodecheck=false;
        Ermes erness=new Ermes("Bank Code Wrong","Please Enter Positive
Number For Bank Code !");
        erness.setVisible(true);
    }
}
if (bcodecheck==true){
    String bank=jTextField2.getText().toUpperCase();
    Connection con;
    Connection ccon;
    Statement stm;
    ResultSet sonuc;
    String Ssql="";
    Ssql = ("Select bankcode,bankname from banks");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String codecheck="";
        String namecheck="";

        while (sonuc.next()){
            codecheck=sonuc.getString(1);
            namecheck=sonuc.getString(2);
            if (code.equals(codecheck)){
                ccode=ccode+1;
            }
            if (bank.equals(namecheck)){
                cname=cname+1;
            }
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {

```

```

        System.err.print("SQLException: "+ex.getMessage());
    }
    if (ccode==0){
        if (cname==0){
            if (code!=""){
                if (bank!=""){
                    if (country!="Choose a Country"){

                        try {
                            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
                        }

                        catch(java.lang.ClassNotFoundException e) {
                            System.err.print("ClassNotFoundException: ");
                            System.err.print(e.getMessage());
                        }
                        try {
                            ccon = DriverManager.getConnection("jdbc:odbc:FC","ff","f");
                            PreparedStatement pstmt=ccon.prepareStatement(
                                "INSERT INTO banks(bankcode,bankname,country,usr)
VALUES(?,?,?,?)");
                            pstmt.setString(1,code);
                            pstmt.setString(2,bank);
                            pstmt.setString(3, country);
                            pstmt.setString(4,user);
                            pstmt.executeUpdate();

                            pstmt.close();
                            ccon.close();
                            jTextField1.setText("");

                            Ermes ermess=new Ermes("Bank Created Successfully ! ",
"Bank Created");
                            ermess.setVisible(true);
                        } catch(SQLException ex) {
                            System.err.print("SQLException: "+ex.getMessage());
                            Ermes ermess=new Ermes("Bank Creation Failed", "Bank
Creation Failed");
                            ermess.setVisible(true);
                        }
                    }else{
                        Ermes ermess=new Ermes("Bank Creation Failed", "Country is
Missing");
                        ermess.setVisible(true);
                    }
                }else{
                    Ermes ermess=new Ermes("Bank Creation Failed", "Bank Name is
Missing");
                    ermess.setVisible(true);
                }
            }
        }
    }

```

```

        }else{
            Ermes ermess=new Ermes("Bank Creation Failed", "Bank Code is
Missing");
            ermess.setVisible(true);
        }
    }else{
        Ermes ermess=new Ermes("Bank Name Failed", "Bank Name Has Used
For Another Bank");
        ermess.setVisible(true);
    }
}
}
}
}

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

```

```

/**

```

```

 * @param args the command line arguments

```

```

 */

```

```

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Banks().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify

```

```

private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
// End of variables declaration

```

```

}

```


Creditpayments.Java

```
/*
 * Creditpayments.java
 *
 * Created on January 29, 2006, 11:51 PM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Creditpayments extends javax.swing.JFrame implements ItemListener{
    String customer="Choose a Customer";
    String account="Choose an Account";
    String amount="";
    String insleft="";
    String user=Usrpass.getuser();
    Boolean amountcheck=true;

    /** Creates new form Creditpayments */
    public Creditpayments() {
        initComponents();
        this.setLocation(200, 200);
        jComboBox1.addItem("Choose a Customer");
        jComboBox2.addItem("Choose an Account");
        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
        }
    }
}
```

```

String s="";
while (sonuc.next()){
    s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
    jComboBox1.addItem(s);
}

stm.close();
con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
jComboBox1.addItemListener(this);
jComboBox2.addItemListener(this);
}

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jComboBox1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        //Customer Selection
        customer=jComboBox1.getSelectedItem().toString();
        jComboBox2.removeItemListener(this);
        jComboBox2.removeAllItems();
        jComboBox2.addItem("Choose an Account");
        jComboBox2.addItemListener(this);
        if (customer=="Choose a Customer"){
            Owner ermess=new Owner("Customer is Missing","Please Choose an
Owner First!");
            ermess.setVisible(true);

        }else{
            customer=customer.substring(0,(customer.indexOf(" ")));
            System.out.println("customer"+ customer);
            //Account Selection
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;
            String check=""+ "YES" +"";
            System.out.println("customer"+customer);
            Ssql = ("Select accountnum,aamount from accounts where
acustnum="+customer+"and accredited="+check+" ");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");

```

```

        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String s="";
        int count=0;
        while (sonuc.next()){
            s=(sonuc.getString(1)+" "+sonuc.getString(2));
            jComboBox2.addItem(s);
            count=count+1;
        }
        if (count==0){
            Ermes ermess=new Ermes("Customer Has Not Had Any Credited
Account!", "Please Check The Customer!");
            ermess.setVisible(true);
        }
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!", "Please Check The Account!");
        ermess.setVisible(true);
    }
}
}

```

```

//account Selection
if( event.getSource() == jComboBox2&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    account=jComboBox2.getSelectedItem().toString();
    if (account=="Choose an Account"){
        Ermes ermess=new Ermes("Account Has not Chooosen", "Please Choose
an Account!");
        ermess.setVisible(true);
    }else{
        account=account.substring(0,(account.indexOf(" ")));
        System.out.println("account"+account);
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        Ssql = ("Select insleft,amountleft,insamount from creditedaccounts where
canumber="+account+" ");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
    }
}

```



```

    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff","f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        String insleft="";
        String amountleft="";
        String insamount="";
        while (sonuc.next()){
            insleft=sonuc.getString(1);
            amountleft=sonuc.getString(2);
            insamount=sonuc.getString(3);
        }
        jLabel10.setText(insamount);
        jLabel7.setText(insleft);
        jLabel8.setText(amountleft);
        stm.close();
        con.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Customer Has Not Had Any
Account!","Please Check The Customer!");
        ermess.setVisible(true);
    }
}
}
}
}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

```

```

// <editor-fold defaultstate="collapsed" desc="Generated Code ">

```

```

private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    jComboBox2 = new javax.swing.JComboBox();
    jTextField1 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jLabel7 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
    jLabel10 = new javax.swing.JLabel();
}

```

```

getContentPane().setLayout(null);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Credit Payments");
getContentPane().add(jLabel1);
jLabel1.setBounds(140, 10, 150, 17);

jLabel2.setText("Credit Owner");
getContentPane().add(jLabel2);
jLabel2.setBounds(10, 40, 90, 14);

jLabel3.setText("Credited Account");
getContentPane().add(jLabel3);
jLabel3.setBounds(10, 70, 90, 14);

jLabel4.setText("Payment Amount");
getContentPane().add(jLabel4);
jLabel4.setBounds(10, 100, 100, 14);

getContentPane().add(jComboBox1);
jComboBox1.setBounds(160, 40, 180, 22);

getContentPane().add(jComboBox2);
jComboBox2.setBounds(160, 70, 180, 22);

getContentPane().add(jTextField1);
jTextField1.setBounds(160, 100, 180, 19);

jButton1.setText("Close");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

getContentPane().add(jButton1);
jButton1.setBounds(350, 40, 70, 80);

jButton2.setText("Save Payment");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

getContentPane().add(jButton2);
jButton2.setBounds(10, 130, 410, 23);

```

```

jLabel5.setText("Installment Left");
getContentPane().add(jLabel5);
jLabel5.setBounds(30, 160, 120, 14);

jLabel6.setText("Amount Left");
getContentPane().add(jLabel6);
jLabel6.setBounds(30, 200, 110, 14);

jLabel7.setText("Installment");
getContentPane().add(jLabel7);
jLabel7.setBounds(160, 160, 90, 14);

jLabel8.setText("Amount");
getContentPane().add(jLabel8);
jLabel8.setBounds(160, 200, 140, 14);

jLabel9.setText("Installment Amount");
getContentPane().add(jLabel9);
jLabel9.setBounds(30, 180, 120, 14);

jLabel10.setText("Installment Amount");
getContentPane().add(jLabel10);
jLabel10.setBounds(160, 180, 150, 14);

pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

private static final boolean isNumber(final String s) {
    final char[] numbers = s.toCharArray();
    for (int x = 0; x < numbers.length; x++) {
        final char c = numbers[x];
        if ((c >= '0') && (c <= '9')) continue;
        return false; // invalid
    }
    return true; // valid
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    amount=jTextField1.getText();
    if (amount==""){

    }else{
        if (isNumber(amount)==true){
            amountcheck=true;
        }else{

```



```

        amountcheck=false;
        Ermes ermess=new Ermes("Amount Wrong","Please Enter Positive
Number For Amount !");
        ermess.setVisible(true);
    }
}
if (amountcheck==true){
    Connection con;
    Connection conn;
    Statement stmm;
    String Ssql;
    ResultSet sonuc;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        PreparedStatement pstmt=con.prepareStatement(
            "INSERT INTO creditpayments(custnum,accnum,paymentamount,usr)
VALUES(?,?,?,?)");
        System.out.println("customer on "+customer);
        System.out.println("account on"+account);
        System.out.println("amount on"+amount);

        pstmt.setString(1,customer);
        pstmt.setString(2,account);
        pstmt.setString(3,amount);
        pstmt.setString(4,user);
        pstmt.executeUpdate();
        System.out.println("ilki oldu ");
        pstmt.close();
        con.close();
        jTextField1.setText("");

        Ermes ermess=new Ermes("Credit Installment Paid Successfully !",
"Installment Paid");
        ermess.setVisible(true);

        //Updating Account Situation
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

        }

        catch(java.lang.ClassNotFoundException e) {

```

```

        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try{
        conn = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stmm = conn.createStatement();
        String pos=""+"YES"+"";

        Ssql="update accounts set acredited="+pos+" where
accountnum="+account+" ";
        stmm.addBatch(Ssql);
        stmm.executeBatch();
        System.out.println("ikinci oldu");
        stmm.close();
        conn.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    //Subtracting Installment Number From Credited Account

    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }

    try{
        conn = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stmm = conn.createStatement();
        String pos=""+"YES"+"";

        Ssql="update creditedaccounts set insleft=insleft-
1,amountleft=amountleft-"+amount+" where canumber="+account+" and
creditclosed='N' ";
        stmm.addBatch(Ssql);
        stmm.executeBatch();

        stmm.close();
        conn.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    // Closing Credit If It's Installment End

    try {

```

```

        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try{
        conn = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stmm = conn.createStatement();
        String s="";

        Ssql="select amountleft from creditedaccounts where
canumber="+account+" ";
        stmm = con.createStatement();
        sonuc = stmm.executeQuery(Ssql);
        while (sonuc.next()){
            insleft=sonuc.getString(1);
        }
        if (Integer.parseInt(insleft)==0){
            // Marking Account As Credit Payment Ended
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try{
                conn = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stmm = conn.createStatement();
                String pos="""+"NO"+""";

                Ssql="update accounts set acredited="+pos+" where
accountnum="+account+" ";
                stmm.addBatch(Ssql);
                stmm.executeBatch();

                stmm.close();
                conn.close();
                Ermes ermesss=new Ermes("Account Installment Payment's Has
Finished ! ", "Installment Payment's Finished");
                ermess.setVisible(true);

            } catch(SQLException ex) {

```



```

        System.err.print("SQLException: "+ex.getMessage());
    }
    }else{
        Ermes ermesss=new Ermes("Account Has "+Integer.parseInt(insleft)+
" Installment More ! ", Integer.parseInt(insleft)+" Installment Payment More");
        ermess.setVisible(true);
    }
    stmm.close();
    conn.close();
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
    }
    } catch(SQLException ex) {
        System.err.print("SQLException: "+ex.getMessage());
        Ermes ermess=new Ermes("Error","Missing Field or Fields");
        ermess.setVisible(true);
    }
    }
}

```

/**

* @param args the command line arguments
*/

```

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Creditpayments().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JTextField jTextField1;
// End of variables declaration

```

```

}

```

Reportalltransfers.Java

```
/*
 * Reportalltransfers.java
 *
 * Created on February 14, 2006, 7:16 AM
 */

package c;
import java.sql.*;
/**
 *
 * @author Cheyenne
 */
public class Reportalltransfers extends javax.swing.JFrame {

    /** Creates new form Reportalltransfers */
    public Reportalltransfers() {
        initComponents();
        this.setLocation(50, 50);
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        //jTable1.setValueAt("Name", 0, 1);
        //jTable1.setValueAt("Surname", 0, 2);

        Ssql = ("Select
ftdate,FUNDTRNUM,FTCUSTNUM,FTACCOUNTNUM,FRCUSTNUM,fraccounnum,
FRCUSTNM,famount,frbank from fundtransfers order by fundtrnum");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            int row=0;
            while (sonuc.next()){
                //s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                int column=0;
                jTable1.setValueAt(sonuc.getString(1),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(2),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(3),row, column);
            }
        }
    }
}
```

```

        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(5),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(6),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(7),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(8),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(9),row, column);
        row++;
    }

    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
    jLabel1.setText("All Fund Transfers ");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(270, 20, 240, 17);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},

```



```

    }
    ){
        boolean[] canEdit = new boolean [] {
            false, false, false, false, false, false, false, false, false
        };

        public boolean isCellEditable(int rowIndex, int columnIndex) {
            return canEdit [columnIndex];
        }
    });
    jScrollPane1.setViewportView(jTable1);

    getContentPane().add(jScrollPane1);
    jScrollPane1.setBounds(22, 42, 670, 510);

    jButton1.setText("Cancel");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    getContentPane().add(jButton1);
    jButton1.setBounds(710, 40, 90, 510);

    pack();
}
// </editor-fold>
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Reportalltransfers().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration
}

```

Revenuereport.java

```
/*
 * Revenuereport.java
 *
 * Created on February 15, 2006, 4:32 AM
 */

package c;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;
/**
 *
 * @author Cheyenne
 */
public class Revenuereport extends javax.swing.JFrame implements ItemListener{
    int flag=0;
    /** Creates new form Revenuereport */
    public Revenuereport() {
        initComponents();
        this.setLocation(200, 200);
        buttonGroup1.add(jRadioButton1);
        buttonGroup1.add(jRadioButton2);
        buttonGroup1.add(jRadioButton3);
        jLabel2.setVisible(false);
        jComboBox1.setVisible(false);
        jRadioButton1.addItemListener(this);
        jRadioButton2.addItemListener(this);
        jRadioButton3.addItemListener(this);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code ">
    private void initComponents() {
        buttonGroup1 = new javax.swing.ButtonGroup();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jComboBox1 = new javax.swing.JComboBox();
        jRadioButton1 = new javax.swing.JRadioButton();
        jRadioButton2 = new javax.swing.JRadioButton();
        jRadioButton3 = new javax.swing.JRadioButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
        jButton1 = new javax.swing.JButton();
    }
}
```



```
getContentPane().setLayout(null);
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
```

```
jLabel1.setText("Revenue Search");
```

```
getContentPane().add(jLabel1);
```

```
jLabel1.setBounds(210, 10, 180, 17);
```

```
jLabel2.setText("Customer");
```

```
getContentPane().add(jLabel2);
```

```
jLabel2.setBounds(40, 130, 150, 14);
```

```
getContentPane().add(jComboBox1);
```

```
jComboBox1.setBounds(170, 130, 200, 22);
```

```
jRadioButton1.setText("Search By Customer");
```

```
getContentPane().add(jRadioButton1);
```

```
jRadioButton1.setBounds(40, 40, 200, 23);
```

```
jRadioButton2.setText("Search By Account");
```

```
getContentPane().add(jRadioButton2);
```

```
jRadioButton2.setBounds(40, 70, 170, 23);
```

```
jRadioButton3.setText("Search By Revenue Type");
```

```
getContentPane().add(jRadioButton3);
```

```
jRadioButton3.setBounds(40, 100, 180, 23);
```

```
jTable1.setModel(new javax.swing.table.DefaultTableModel(
```

```
new Object [][] {
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```

```
{null, null, null, null},
```


[illegible]


```

    pack();
}
// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    this.setVisible(false);
}

public void itemStateChanged( ItemEvent event ) {
    if( event.getSource() == jButton1 && event.getStateChange() ==
ItemEvent.SELECTED ) {
        JComboBox1.removeItemListener(this);
        JComboBox1.removeAllItems();
        jLabel2.setText("Choose Customer");
        JComboBox1.addItem("Choose a Customer");
        jLabel2.setVisible(true);
        JComboBox1.setVisible(true);
        //Filling the Combobox for owner
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;

        Ssql = ("Select custnum,custnm,custsnm from customers");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            String s="";
            while (sonuc.next()){
                s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
                JComboBox1.addItem(s);
            }

            stm.close();
            con.close();
        } catch(SQLException ex) {
            System.err.print("SQLException: "+ex.getMessage());
        }
        JComboBox1.addItemListener(this);
        flag=1;
    }
}

```



```

        if( event.getSource() == jButton2 && event.getStateChange() ==
ItemEvent.SELECTED ) {
            jComboBox1.removeItemListener(this);
            jComboBox1.removeAllItems();
            jLabel2.setText("Choose Account");
            jLabel2.setVisible(true);
            jComboBox1.addItem("Choose an Account");
            jComboBox1.setVisible(true);
            //Filling the Combobox for Account
            Connection con;
            Statement stm;
            ResultSet sonuc;
            String Ssql;

            Ssql = ("Select accountnum,acustnum from accounts");
            try {
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            }

            catch(java.lang.ClassNotFoundException e) {
                System.err.print("ClassNotFoundException: ");
                System.err.print(e.getMessage());
            }
            try {
                con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
                stm = con.createStatement();
                sonuc = stm.executeQuery(Ssql);
                String s="";
                while (sonuc.next()){
                    s=(sonuc.getString(1)+" Owner:"+sonuc.getString(2));
                    jComboBox1.addItem(s);
                }

                stm.close();
                con.close();
            } catch(SQLException ex) {
                System.err.print("SQLException: "+ex.getMessage());
            }
            flag=2;
            jComboBox1.addItemListener(this);
        }
        if( event.getSource() == jButton3 && event.getStateChange() ==
ItemEvent.SELECTED ) {
            jComboBox1.removeItemListener(this);
            jComboBox1.removeAllItems();
            jLabel2.setText("Choose Revenue Type");
            jLabel2.setVisible(true);
            jComboBox1.addItem("Choose Revenue Type");
            jComboBox1.addItem("Credit revenue");
            jComboBox1.addItem("Fund Transfer revenue");

```

```

jComboBox1.setVisible(true);
flag=3;
jComboBox1.addItemListener(this);
}
if( event.getSource() == jComboBox1&& event.getStateChange() ==
ItemEvent.SELECTED ) {
    if (flag==0){
        Ermes ermess=new Ermes("Choose A Search Condition","Please Choose
A Search Condition!");
        ermess.setVisible(true);
    }
    if (flag==1){
        String customer;
        customer=jComboBox1.getSelectedItem().toString();
        customer=customer.substring(0,(customer.indexOf(" ")));
        //Searching with Customer
        Connection con;
        Statement stm;
        ResultSet sonuc;
        String Ssql;
        Ssql = ("Select revenuedate,revcustomer,revamout,revdesc from
bankrevenues where revcustomer='"+customer+"' ");
        try {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        }

        catch(java.lang.ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.print(e.getMessage());
        }
        try {
            con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
            stm = con.createStatement();
            sonuc = stm.executeQuery(Ssql);
            //s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
            int row=0;
            int rowcount=jTable1.getRowCount();
            int columncount=jTable1.getColumnCount();
            for (int rown = 0; rown < rowcount; rown++){
                for (int columnn = 0; columnn < columncount; columnn++){
                    jTable1.setValueAt("", rown,columnn);
                }
            }
            while (sonuc.next()){
                //s=(sonuc.getString(1)+" "+sonuc.getString(2)+"
"+sonuc.getString(3));
                int column=0;
                jTable1.setValueAt(sonuc.getString(1),row, column);
                column++;
                jTable1.setValueAt(sonuc.getString(2),row, column);

```

```

        column++;
        jTable1.setValueAt(sonuc.getString(3),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
if (flag==2){
    String account;
    account=jComboBox1.getSelectedItem().toString();
    account=account.substring(0,(account.indexOf(" ")));
    //Searching with Account
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;
    Ssql = ("Select revenuedate,revaccount,revamout,revdesc from
bankrevenues where revcustomer='"+account+"' ");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        //s=(sonuc.getString(1)+" "+sonuc.getString(2)+" "+sonuc.getString(3));
        int row=0;
        int rowcount=jTable1.getRowCount();
        int columncount=jTable1.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable1.setValueAt("", rown,columnn);
            }
        }
        while (sonuc.next()){
            int column=0;
            jTable1.setValueAt(sonuc.getString(1),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(2),row, column);
            column++;

```



```

        jTable1.setValueAt(sonuc.getString(3),row, column);
        column++;
        jTable1.setValueAt(sonuc.getString(4),row, column);
        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
if (flag==3){
    String type;
    type=jComboBox1.getSelectedItem().toString();
    //Searching with Revenue Type
    Connection con;
    Statement stm;
    ResultSet sonuc;
    String Ssql;
    Ssql = ("Select revenuedate,revcustomer,revamout,revdesc from
bankrevenues where revdesc='"+type+"' ");
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    }

    catch(java.lang.ClassNotFoundException e) {
        System.err.print("ClassNotFoundException: ");
        System.err.print(e.getMessage());
    }
    try {
        con = DriverManager.getConnection("jdbc:odbc:FC","ff", "f");
        stm = con.createStatement();
        sonuc = stm.executeQuery(Ssql);
        int row=0;
        int rowcount=jTable1.getRowCount();
        int columncount=jTable1.getColumnCount();
        for (int rown = 0; rown < rowcount; rown++){
            for (int columnn = 0; columnn < columncount; columnn++){
                jTable1.setValueAt("", rown,columnn);
            }
        }
        while (sonuc.next()){
            int column=0;
            jTable1.setValueAt(sonuc.getString(1),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(2),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(3),row, column);
            column++;
            jTable1.setValueAt(sonuc.getString(4),row, column);

```

```

        row++;
    }
    stm.close();
    con.close();
} catch(SQLException ex) {
    System.err.print("SQLException: "+ex.getMessage());
}
}
}
}
}
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Revenuereport().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.ButtonGroup buttonGroup1;
private javax.swing.JButton jButton1;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JRadioButton jRadioButton1;
private javax.swing.JRadioButton jRadioButton2;
private javax.swing.JRadioButton jRadioButton3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration

```