



**NEAR EAST UNIVERSITY
FACULTY OF ECONOMICS AND ADMINISTRATIVE
SCIENCES**

**DEPARTMENT OF COMPUTER INFORMATION
SYSTEMS**

**2005/2006 SPRING TERM
CIS 400
(Graduation Project)**

ELECTRONICS SHOP SHOWROOM SYSTEM

SUBMITTED TO: DEPARTMENT OF C.I.S.

**SUBMITTED FROM: ÇAĞLA DELİCEİRMAK
20000115**

LEFKOŞA 2006



NEU

CONTENTS



I. ACKNOWLEDGEMENT.....	2
II. ABSTRACT.....	3
III. INTRODUCTION.....	4
IV. DESING OF THE SYSTEM.....	5
➤ <i>Explanation</i>	5
➤ <i>Data Flow Diagram</i>	10
➤ <i>System Block Diagram</i>	18
➤ <i>Database Design</i>	19
V. USER MANUEL.....	21
➤ User Guidelines.....	21
➤ User Flow Charts.....	27
VI. REFERANCES.....	50
APPENDIX I.....	51
➤ <i>Screen Outputs</i>	51
APPENDIX II.....	83
➤ Source Code.....	83

I. ACKNOWLEDGEMENT

First I would like to thank very much Mrs. Nadire Çavuş who never left his support and always encouraged me during my study.

I would also like to thanks all instructors of department of Computer Information System for their support.

I would like to thank very much to my family, and my friends who have much contributions to my all studies which are invisible.

II. ABSTRACT

Electronics shop showroom system is a program to help the electronic shops to control their stocks as play stations, dvd players, stereos, tvs etc. This program is taking attentions with its user friendly interface also it is very easy to use with high level of functionality.

The program provides the stock controlling features and evaluation criteria is based on a small to medium manufacturing company. A scheduled user manual prepared for helping the users to select an appropriate action.

System provides manage and take hold of records to business transactions.

System has a database for storing data of transactions in company and has an interface to easy use of users.

III. INTRODUCTION

The companies were doing their jobs manually, such as accounting, receiving and order, stock controlling... Etc. But recently IT (Information Technology) started to help companies or firms. Then it has been very popular because it is faster, cheaper than manually and so easy work with IT.

At this point Electronics Shop Showroom System will provide easiness and quickness of controlling stock.

Electronics shop showroom system is a program to help the electronic shops to control their stocks as play stations, dvd players, stereos, tvs etc.

This program is taking attentions with its user friendly interface also it is very easy to use with high level of functionality.

IV. DESIGN OF THE SYSTEM

EXPLANATION

The minimum requirements of the system are as follows :

- Intel Pentium III Processor
- 550 MHz
- 128 MB of Ram
- 30 GB Free Hard Disk Space
- 14'' Monitor
- Microsoft Windows 2000
- Toshiba CD-ROM Drive WM-6802B

The recommended requirements of the system are as follows:

- Intel Pentium IV Processor
- 800 MHz
- 256 MB of Ram
- 360GB Free Hard Disk Space
- 17'' Monitor
- Microsoft Windows XP Professional
- Toshiba CD-ROM Drive WM-6802B

I used Delphi for writing this program and Access database that has connection with Delphi

1) PROJECT IDENTIFICATION AND SELECTION

THE AIM OF THE PROJECT:

Electronics shop showroom system is a program to help the electronic shops to control their stocks as play stations, dvd players, stereos, tvs etc. This program is taking attentions with its user friendly interface also it is very easy to use with high level of functionality.

The existing system's handicaps, the easyness and flexibility that the new system wishes to bring the users and the corrections to be made on the existing system are explained in detail, under the headlines "The Existing System's Capabilities" and "New System's providences".

PROJECT BOUNDARIES :

1. The system will be effored in one pc without network.
2. In this system manager can define different users with some limits.
3. Every user will have his/her password to enter the system.
4. Each product have been checked before delivery. According to the checking operation, Electronics Shop Showroom System does not contain returned goods process.
5. Customer process does not include deleting process because, any current record in the same name of the customes won't be erased.
6. Supplier process does not include deleting process because, any current record in the same name of the suppliers won't be erased.

RESOURCE LIMITS :

System need approximately 1000\$-1200\$ cost common PC. Some of the employees may be sent to computer courses. So all the expenses for courses and hardware will be afforded by the business's own budget.

2) PROJECT INITIATION AND PLANNING

Technical Feasibility:

The hardwares that are necessary for the system are as follows:

- One computer
- One printer

Beside this hardwares, some hardwares and softwares for the new system that will applied and should be afforded as well.

Operational Feasibility:

A test has been made among the employees to find out the computer knowledge levels. According to the results, can choose the employees who have some knowledge about the basics of computer and capable to use the system. However they should be educated about the system.

Economical Feasibility:

All the expences will be afforded by the business's own budget as mentioned before.

3) ANALYSIS

THE REQUIRMENT DETERMINATION :

The requirments made by the users are as follows:

- To increase the speed of operations
- To prevent the loss of data
- To prevent the data duplications
- To provide easy and quick data access
- To prevent the caos caused by the increasing number of customers and stocks
- To restrict data access with the means of system security with password entry

EXISTING SYSTEM CAPABILITIES:

- The shop works in manual system. All of the documents are held in hand.
- All the customer, supplier, product and employee cards are held in hand.
- All archive stores are not in order so finding them is difficult.

NEW SYSTEM PROVIDENCES:

Complete security with user's profile:

Manager can add or delete users who work in his/her business. Manager can give authorize or restrict access to users with security process. Electronics shop automation system can recognize users that manager define and prevent others authomatically.

Customer controll :

Users can see registered customers debts. Also users can make it possible to determine percentage discount of sales.

Product control :

Users can see most demanded products of stocks. Also users can see stocks which are under minimum stock level.

Code based selling and purcahsing:

For selling and purchasing process program use code base sysytem. Users enter the code and sell registered products to customers or add new products in stocks more secure with this system. Code property cause keep more than one product with same name without mix them.

Code based cash or credit receipt:

One of the most extensive properties of the system is code basen cash or credit receipt process. Users can follow products boughth by whom, product information and payment type with code base system. Also it can help to control sellings and customer informations quicker and better.

Chance to see details cash accounts:

Users can see details of cash operations either daily or between two dates. Program can list selling processes of manager or employees. Program can print the list on printer.

Extensive reports:

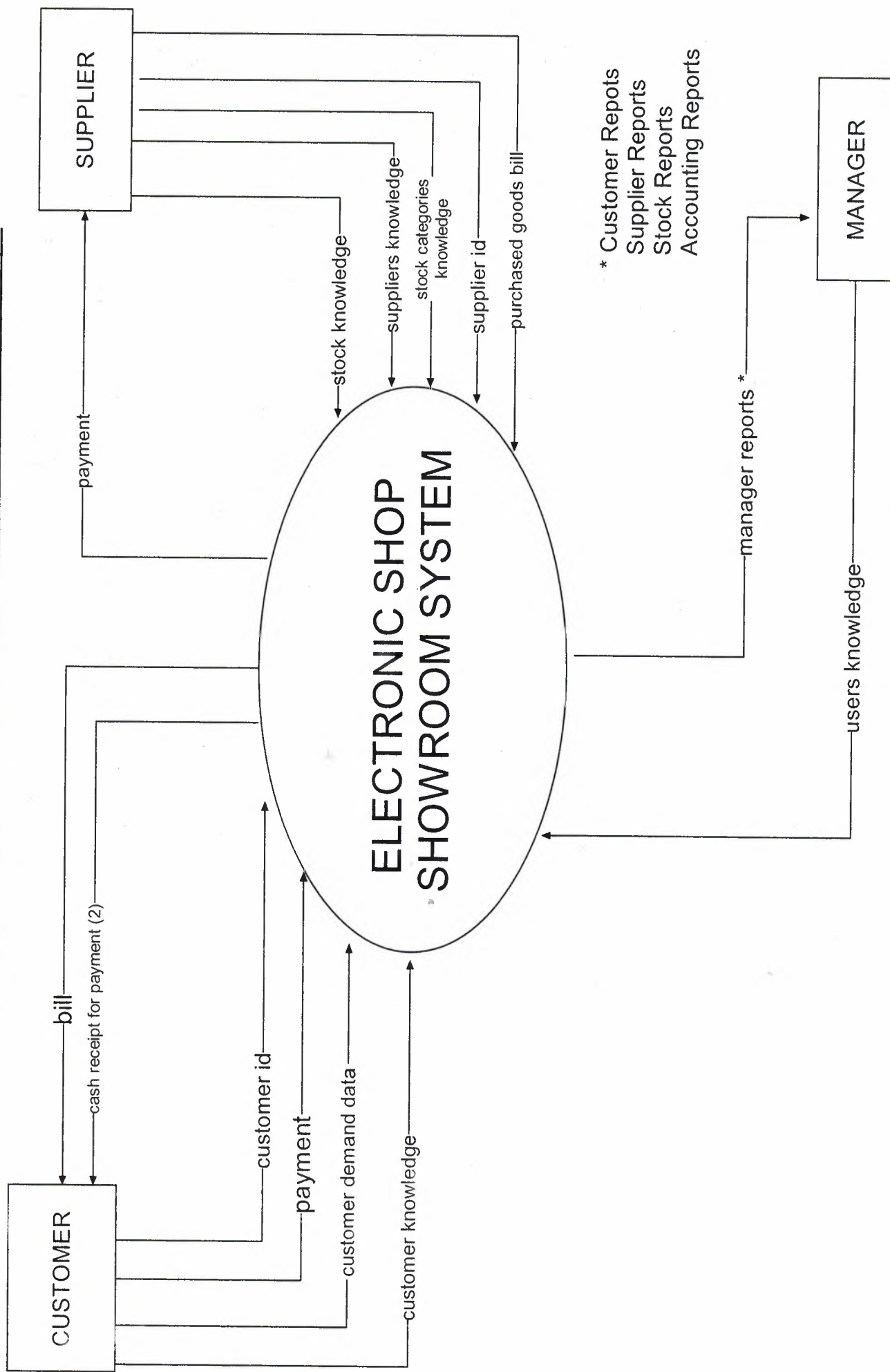
Program can prepare extensive reports concerned with customers,, products, accounting, suppliers of choice.

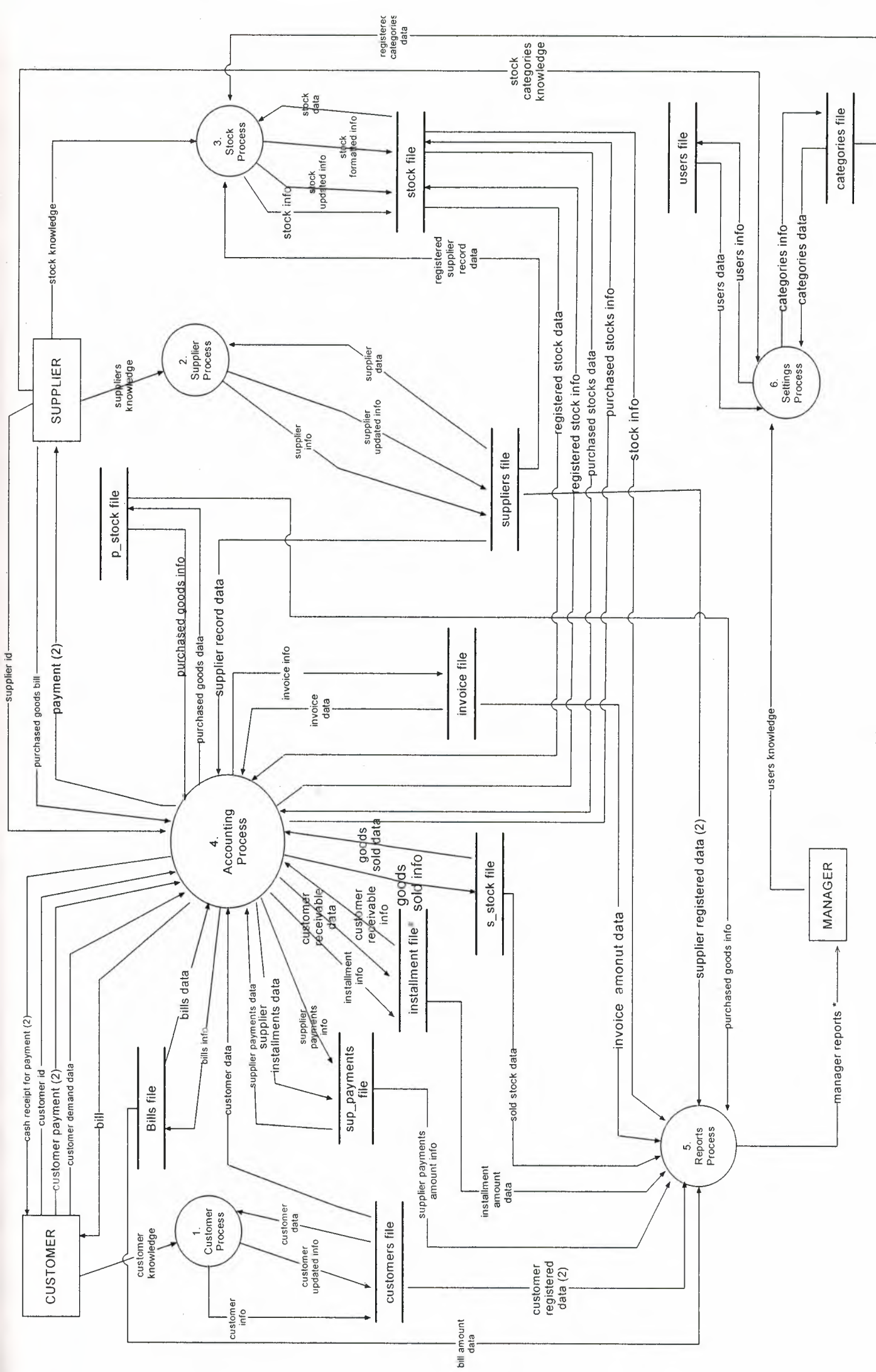
Appearance and themes options for users: Electronics shop automation system addressing users pleasure. Users can change desktop background of program.

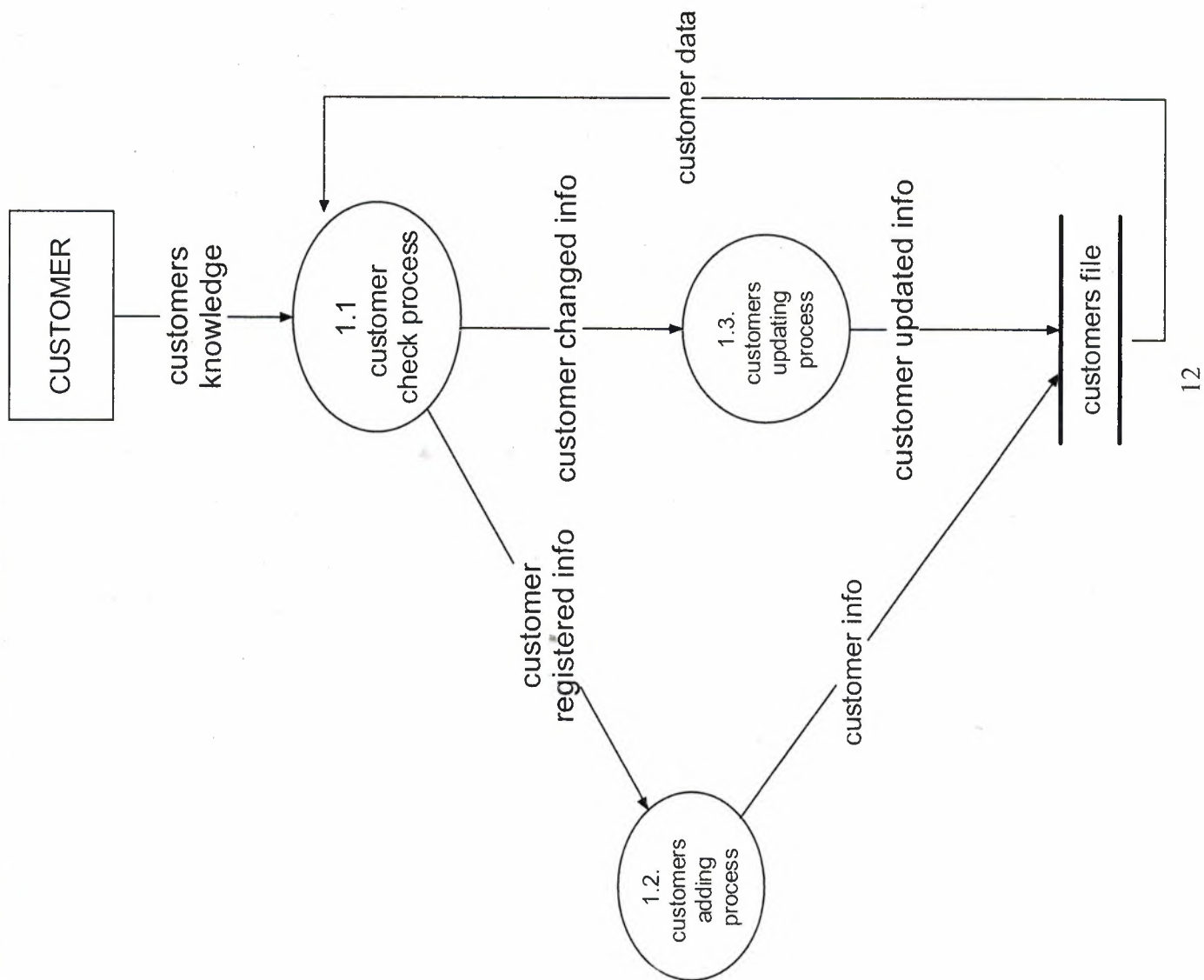
DATABASES:

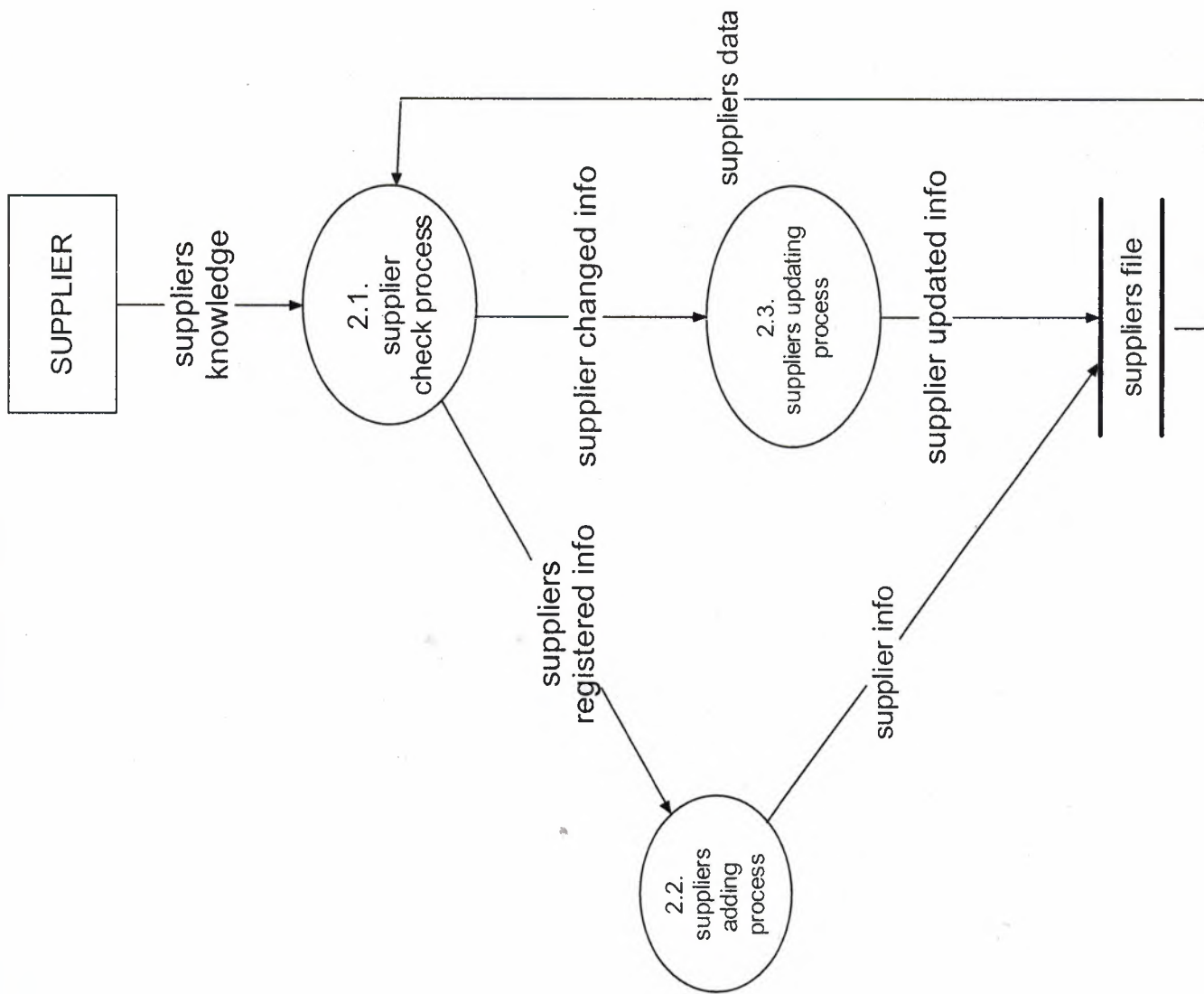
Customers
Stocks
Suppliers
Users
Categories
S_Stocks
P_stocks
Invoice
Bills
Installment
Sup_payments

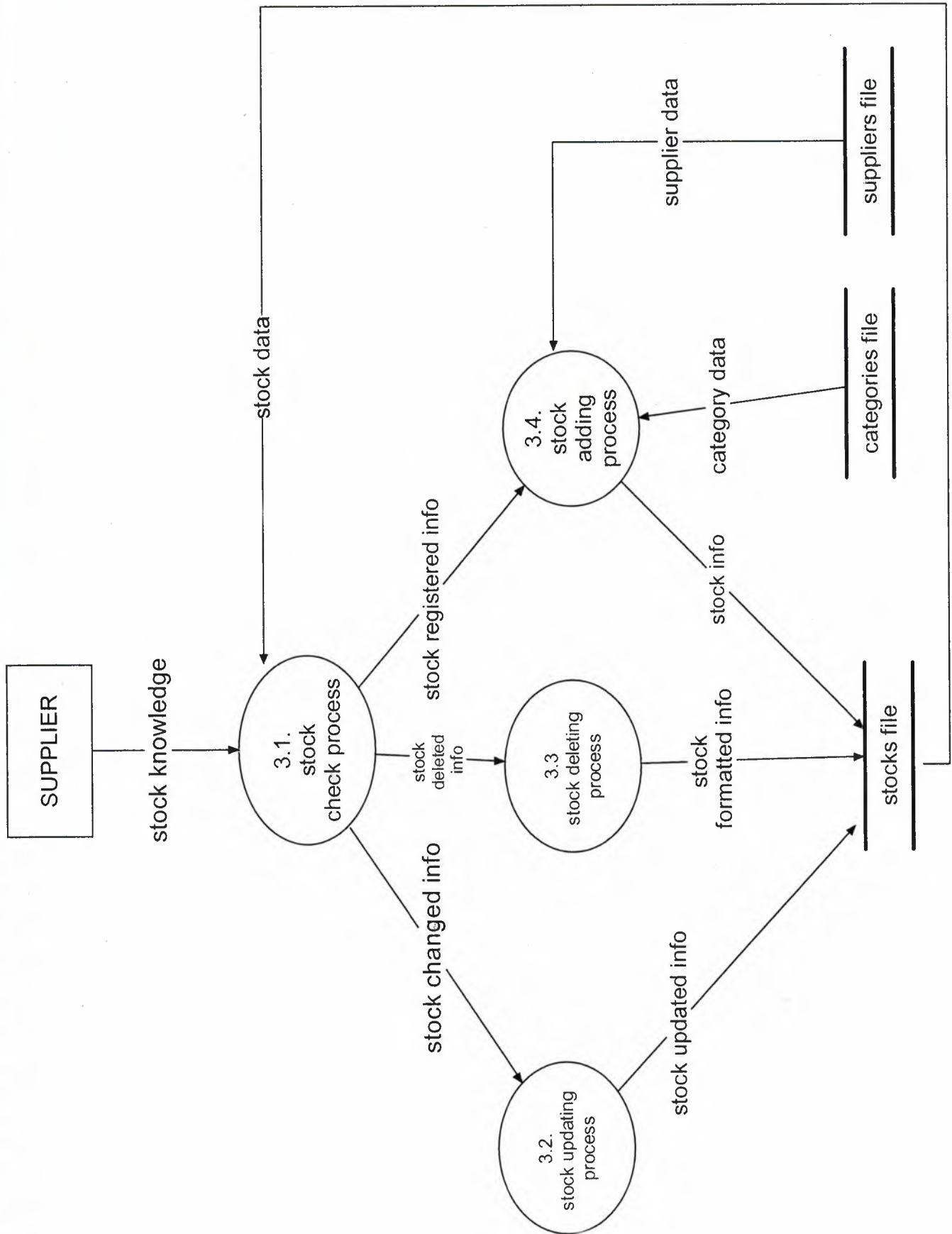
ELECTRONIC SHOP SHOWROOM SYSTEM CONTEXT DATA FLOW DIAGRAM



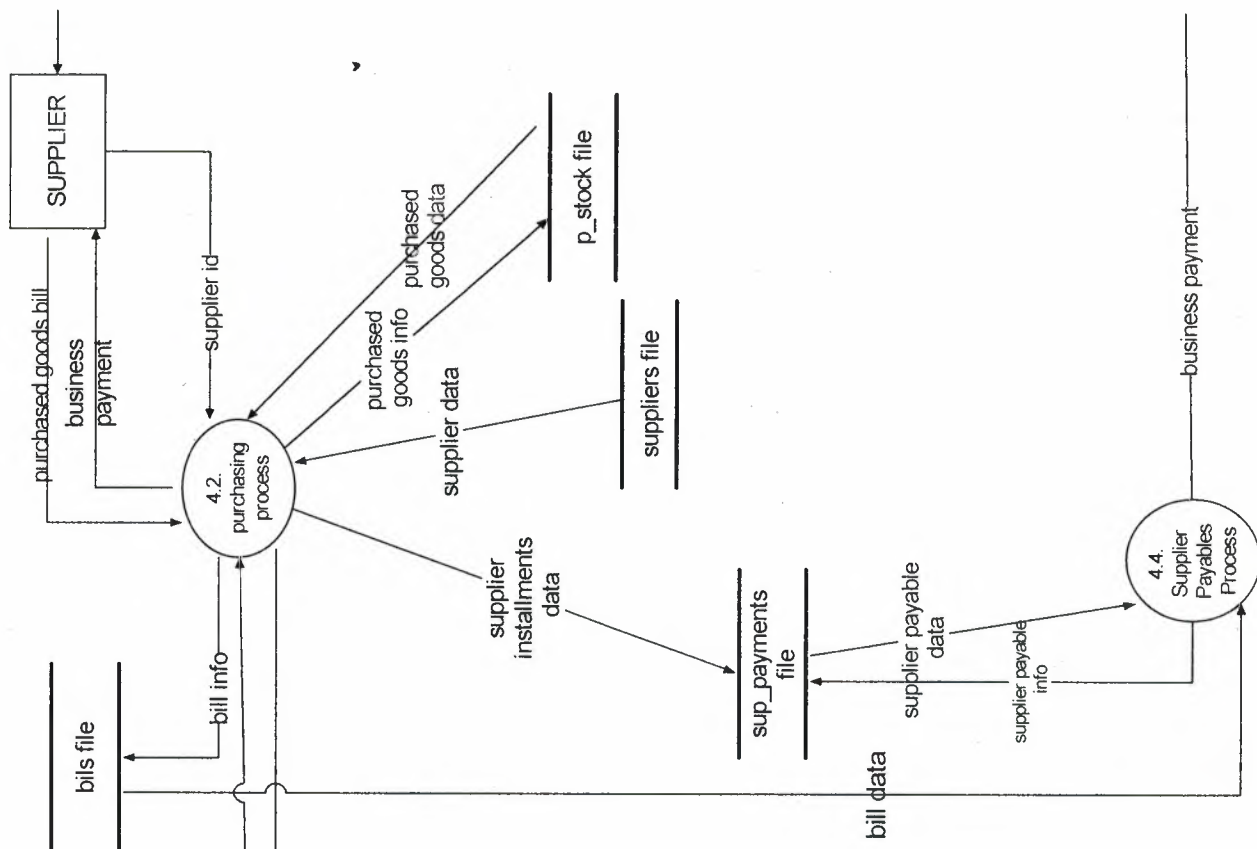
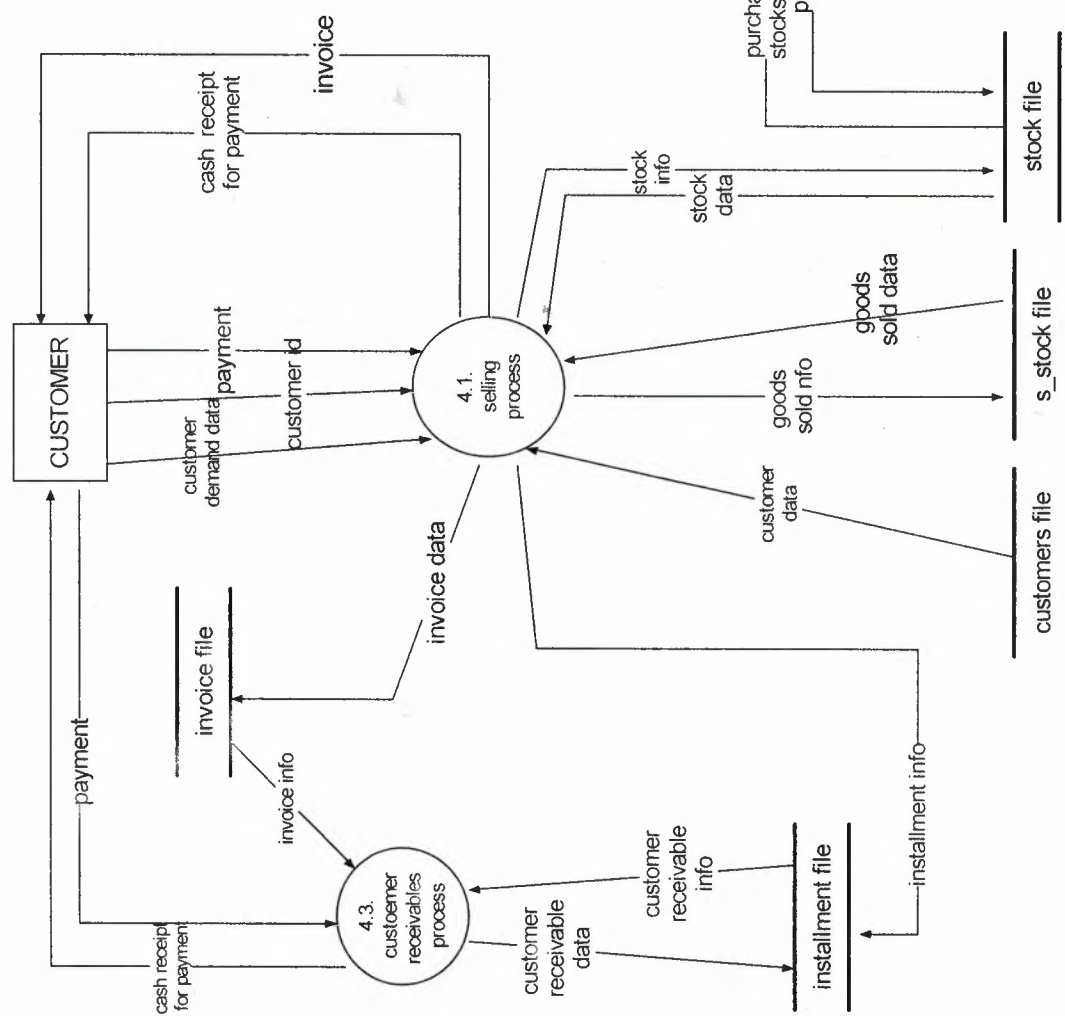


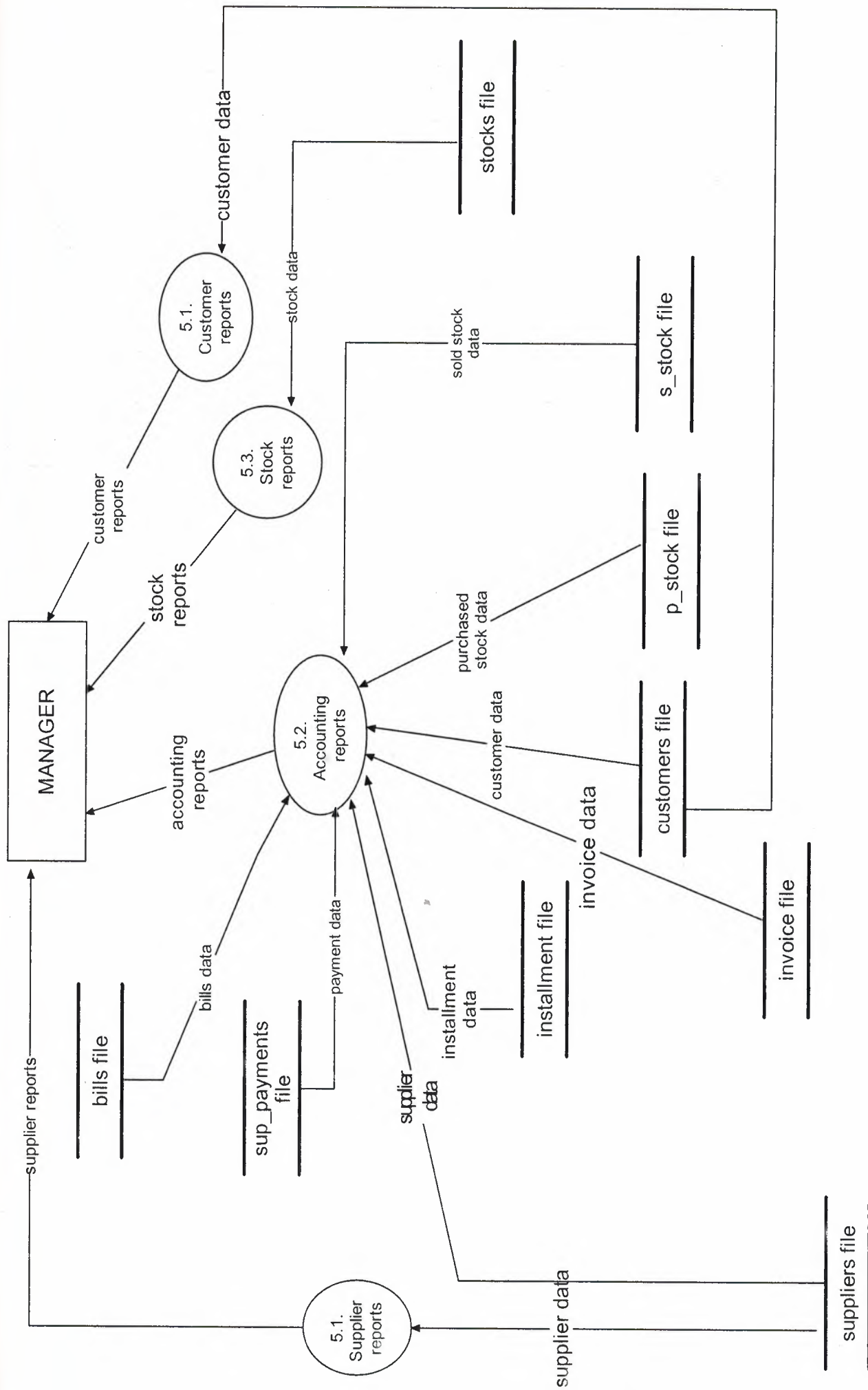




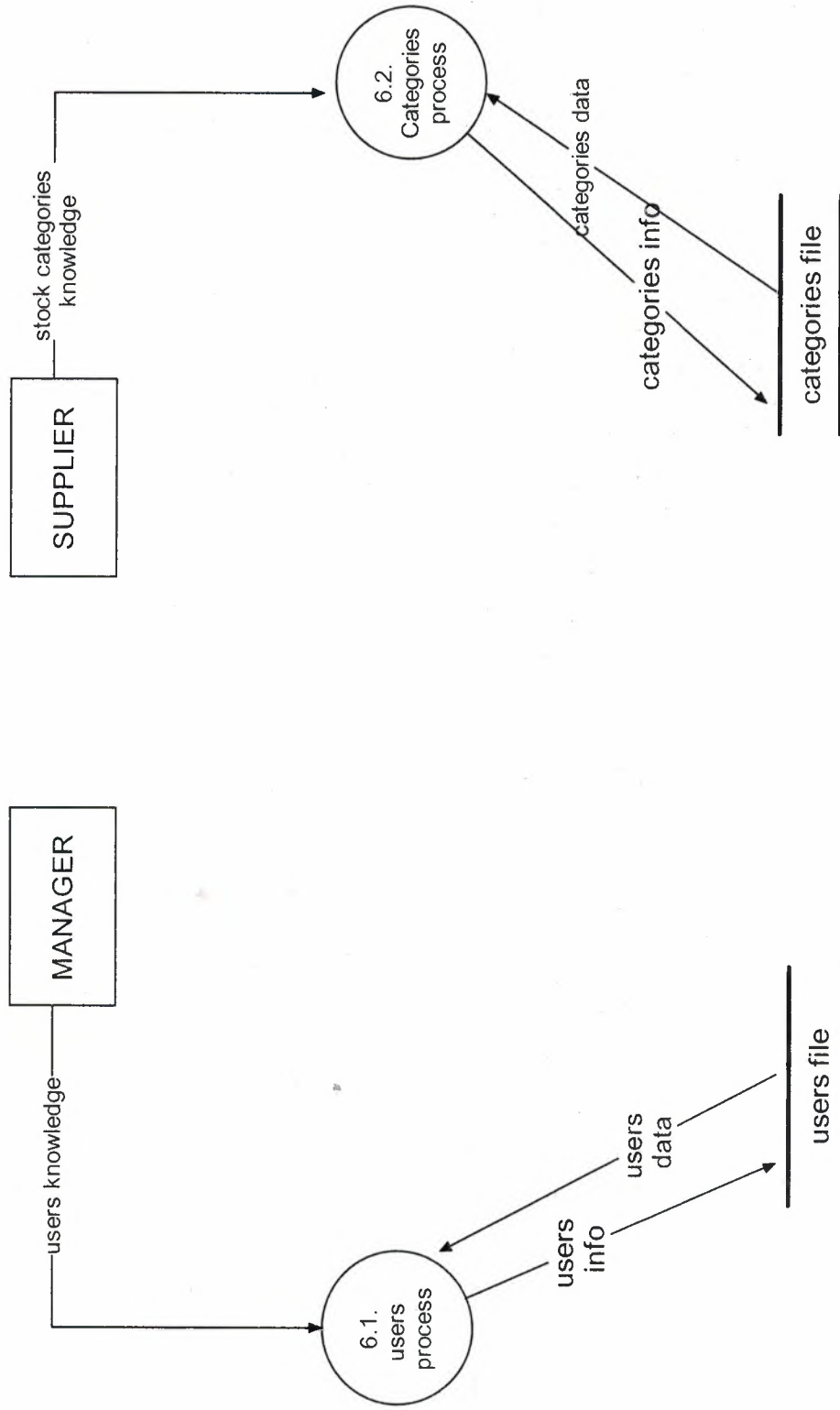


ACCOUNTING PROCESS DETAIL DFD FOR ELECTRONIC SHOP SHOWROOM SYSTEM

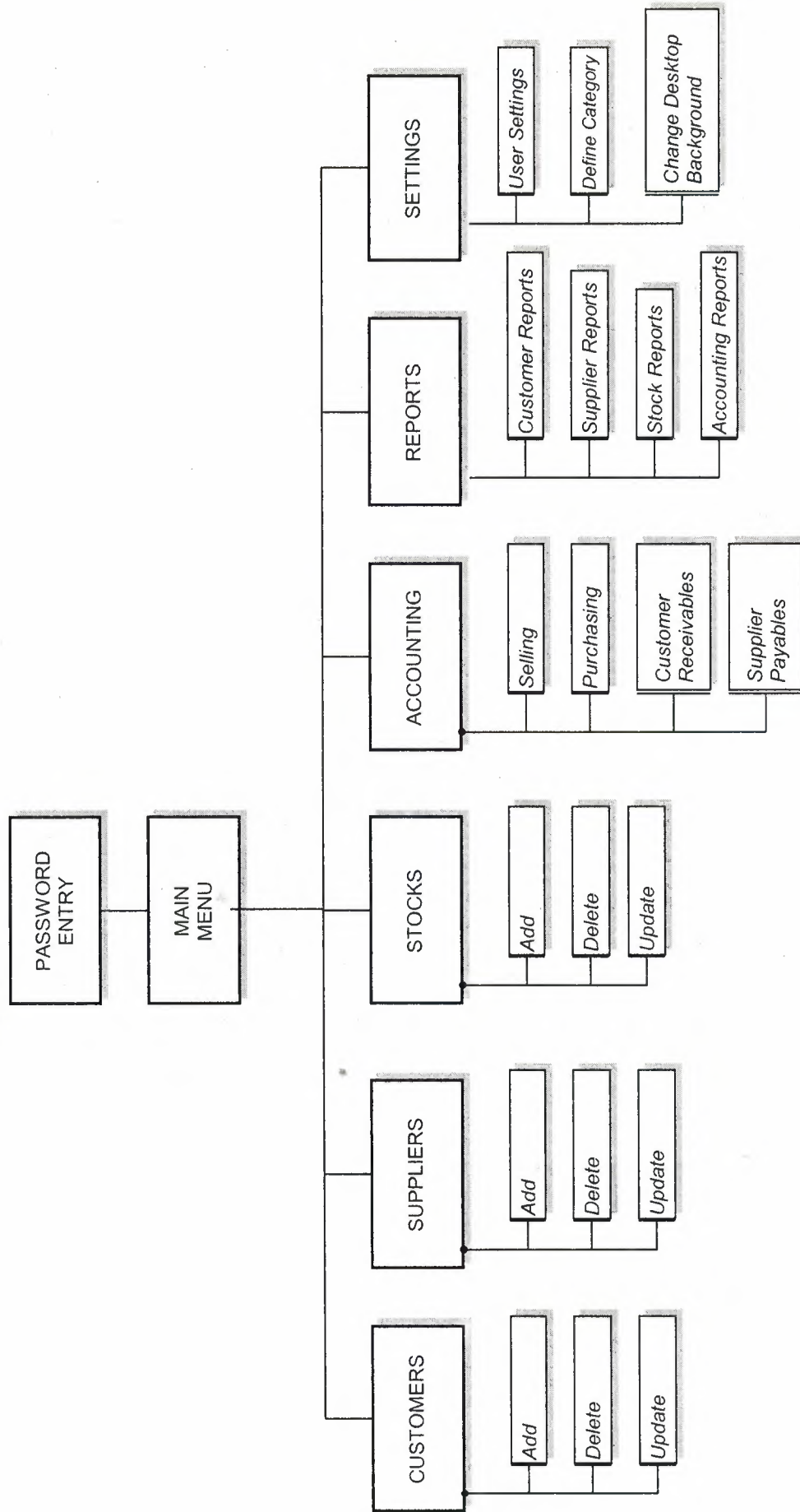




SETTINGS PROCESS DETAIL DFD FOR ELECTRONIC SHOP SHOWROOM SYSTEM



ELECTRONIC SHOP SHOWROOM SYSTEM'S SYSTEM BLOCK DIAGRAM



categories
CategoryId
CategoryName
Picture

Users
UserID
Password
User_Name
User_Surname
Title
Address
City
Phone
Mobile
Customers
Suppliers
Stocks
Accounting
Reports
Settings

V. USER MANUAL

USER GUIDELINES:

1. MAIN MENU

Use main menu items to make any operation or use shortcut buttons to reach specified processes.

- Shortcut to selling is take you to selling process quickly.
- Shortcut to purchasing is take you to purchasing process quickly.
- Shortcut to customer receivables is take you to customer receivables process quickly.
- Shortcut to under minimum stock level report is take you to stocks under minimum stock level report process quickly.

2. CUSTOMERS

You can search, add, delete or update customer records in this process.

- To find any customer, select an item from the list, enter necessary data and click Find button.
- To add new customer record click new button, enter necessary data and click Add button.
- To delete any customer record select customer's record from the list (or use guide buttons) and click delete button. If selected customer have debts, program will not allow you to delete.
- To update any customer record select customer's record from the list (or use guide buttons) and click update button.
- Cancel button enables you to cancel last changings.
- Main menu button returns you to main menu

3. SUPPLIERS

You can search, add, delete or update supplier records in this process.

-To find any supplier, select an item from the list, enter necessary data and click Find button.

- To add new supplier record click new button, enter necessary data and click Add button.

-To delete any supplier record select supplier 's record from the list (or use guide buttons) and click delete button. If you have debts to selected supplier, program will not allow you to delete.

-To update any supplier record select supplier 's record from the list (or use guide buttons) and click update button.

-Cancel button enables you to cancel last changings.

-Main menu button returns you to main menu.

4. STOCKS

You can search, add, delete or update stock records in this process.

-To find any stock, select an item from the list, enter necessary data and click Find button.

-To add new stock record click new button, enter necessary data and click Add button.

-To delete any stock record select stock 's record from the list (or use guide buttons) and click delete button. If selected stock's quantity is not zero or if it is found in returned goods , program will not allow you to delete.

-To update any stock record select stock 's record from the list (or use guide buttons) and click update button.

-Cancel button enables you to cancel last changings.

-Main menu button returns you to main menu.

-Add new category button take you to categories process quickly.

5. ACCOUNTING:

5.1 SELLING

Steps to selling:

1. To sell new product enter customer id, product id , or use help button to find any customer or product. To find any customer or product select an item from the list, enter necessary data, click find and add buttons.
2. Then enter the quantity that you going to sell, press enter key. Program will automaticly calculate sub total, total and total quanttiy.
3. Click add to sales button to add selected product to sales.
4. Click calculate total button if multiple stocks have been added to sellings.
5. Choose payment type and click done button. If you choose credit selling enter necessary data and then click done button.

Main menu button returns you to main menu.

Program prepare bill automatically.

5.2 PURCHASING

Steps to purchasing:

1. To purchase new product enter supplier id, product id , or use help button to find any supplier or product. To find any supplier or product select an item from the list, enter necessary data, click find and add buttons.
2. Then enter the quantity that you going to buy, press enter key. Program will automaticly calculate total quanttiy.
3. Click add to purchases button to add selected product to purchasing list.
4. Click calculate total button if multiple stocks have been added to purchasing list.
5. Choose payment type and click done button. If you choose credit selling enter necessary data and then click done button.

Add new product take you to stock process quickly.

Main menu button returns you to main menu.

5.3 CUSTOMER RECEIVABLES

Steps:

1. To get new payment from the customers select invoice information from the list and click make new payment button. To find any record, select an item from the list, enter necessary data and click Find button.
2. Enter payment amount and press enter key. Program will automatically calculate remaining payment amount. If this is last installment program get remaining payment amount as payment.
3. Click add button to get payment.

Cancel button enables you to cancel last changes.

Main menu button returns you to main menu.

5.4 SUPPLIER PAYMENTS

Steps:

1. To make new payment to the supplier select bill information from the list and click make new payment button. To find any record, select an item from the list, enter necessary data and click Find button.
2. Enter payment amount and press enter key. Program will automatically calculate remaining payment amount. If this is last installment program get remaining payment amount as payment.
3. Click add button to make payment.

Cancel button enables you to cancel last changes.

Main menu button returns you to main menu.

6. REPORTS:

6.1 CUSTOMER REPORTS

Make report choice and click print preview button.

6.2 SUPPLIER REPORTS

Make report choice and click print preview button.

6.3 STOCK REPORTS

Make report choice and click print preview button.

6.4 ACCOUNTING REPORTS

Make report choice and click print preview button.

When all sales, all purchases or cash analyse report has been selected, select any date from date interval section and click filter button.

7. SETTINGS

7.1 USERS

You can search, add, delete or update user records in this process.

-To find any user, select an item from the list, enter necessary data and click Find button.

-To add new user record click new button, enter necessary data and click Add button.

-To delete any user record select customer's record from the list (or use guide buttons) and click delete button.

-To update any user record select customer's record from the list (or use guide buttons) and click update button.

-Cancel button enables you to cancel last changes.

-Main menu button returns you to main menu

7.2 CATEGORIES

You can search, add, delete or update categories in this process.

- To find any category, select an item from the list, enter necessary data and click Find buton.

- To add new category click new buton, enter necessary data and click Add buton.

- To delete any category select customer's record from the list (or use guide buttons) and click delete buton.

- To update any category select customer's record from the list (or use guide buttons) and click update buton.

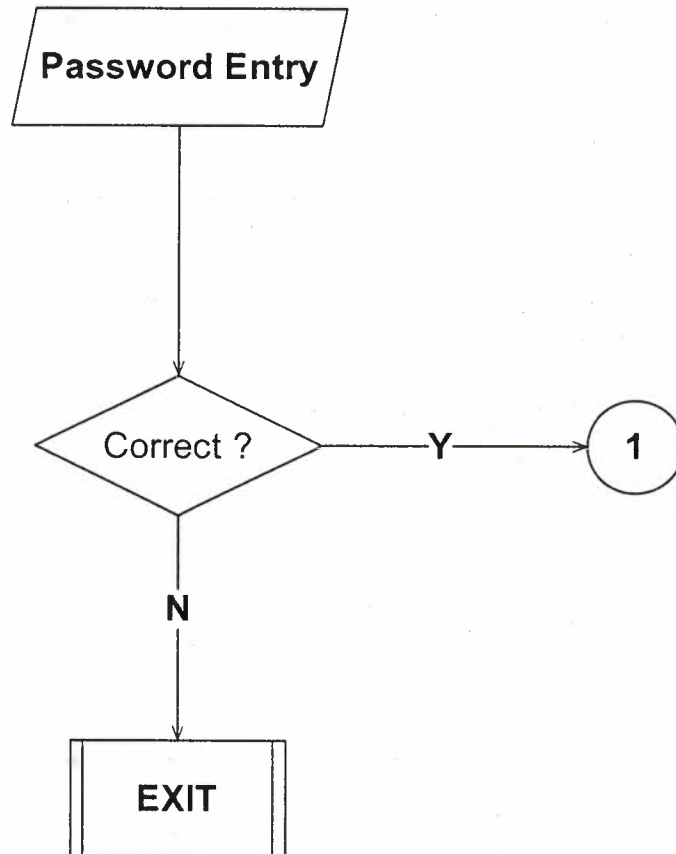
- Cancel buton enables you to cancel last changings.

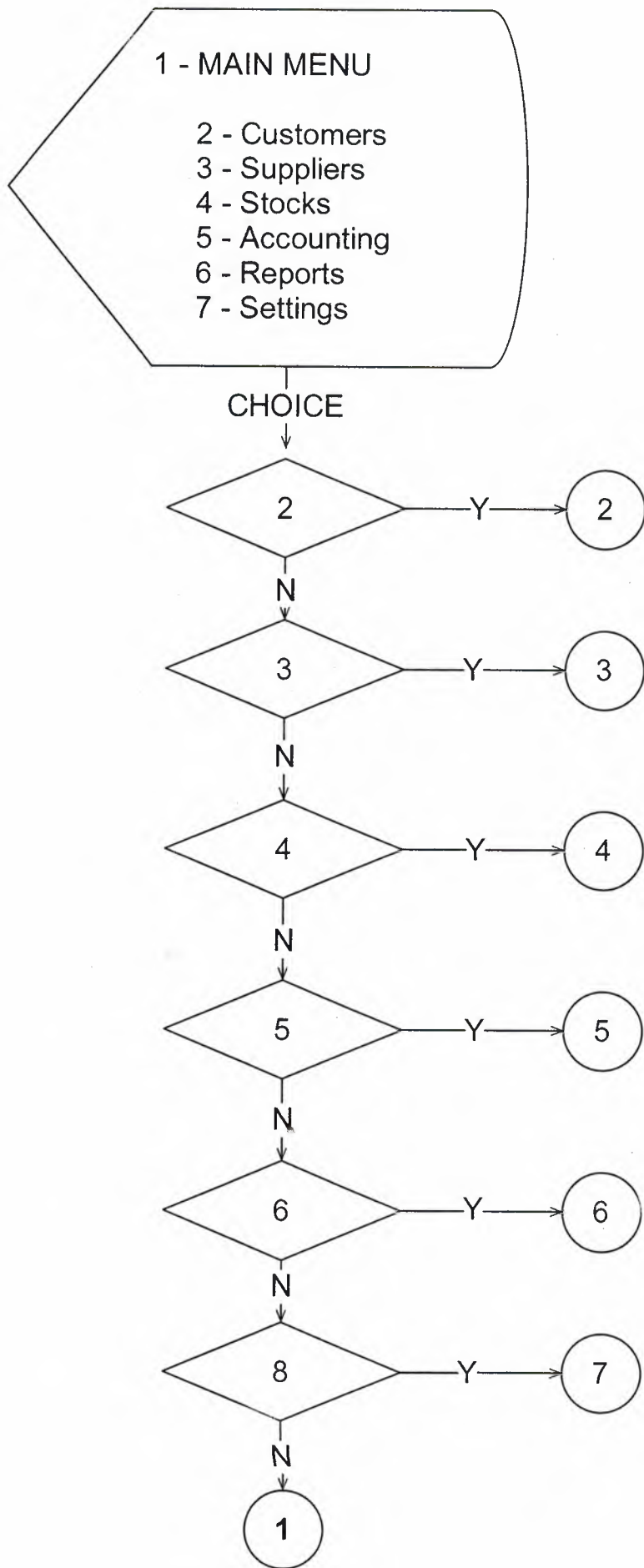
- Main menu button returns you to main menu.

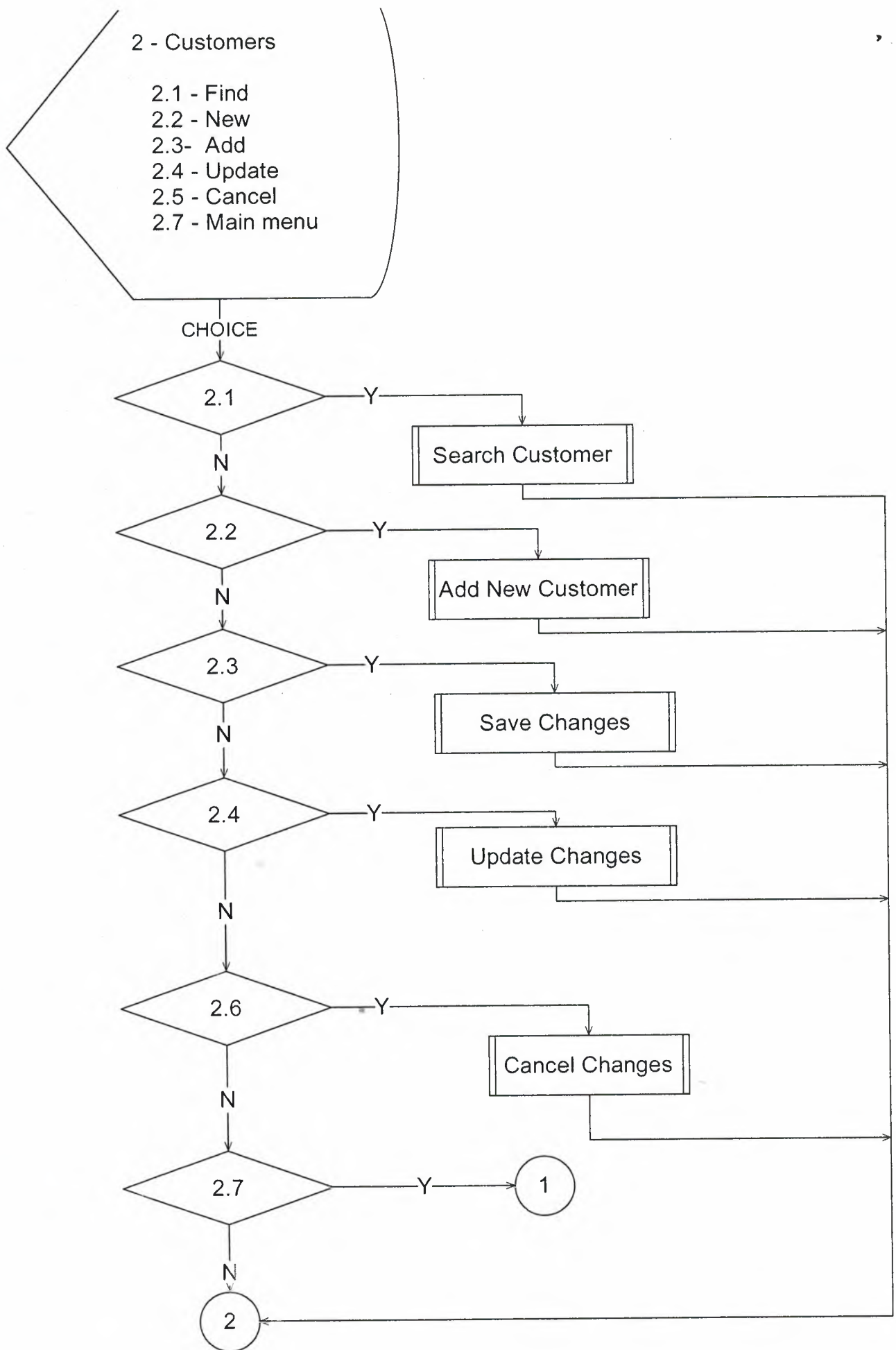
7.3 CHANGE BACKGROUND

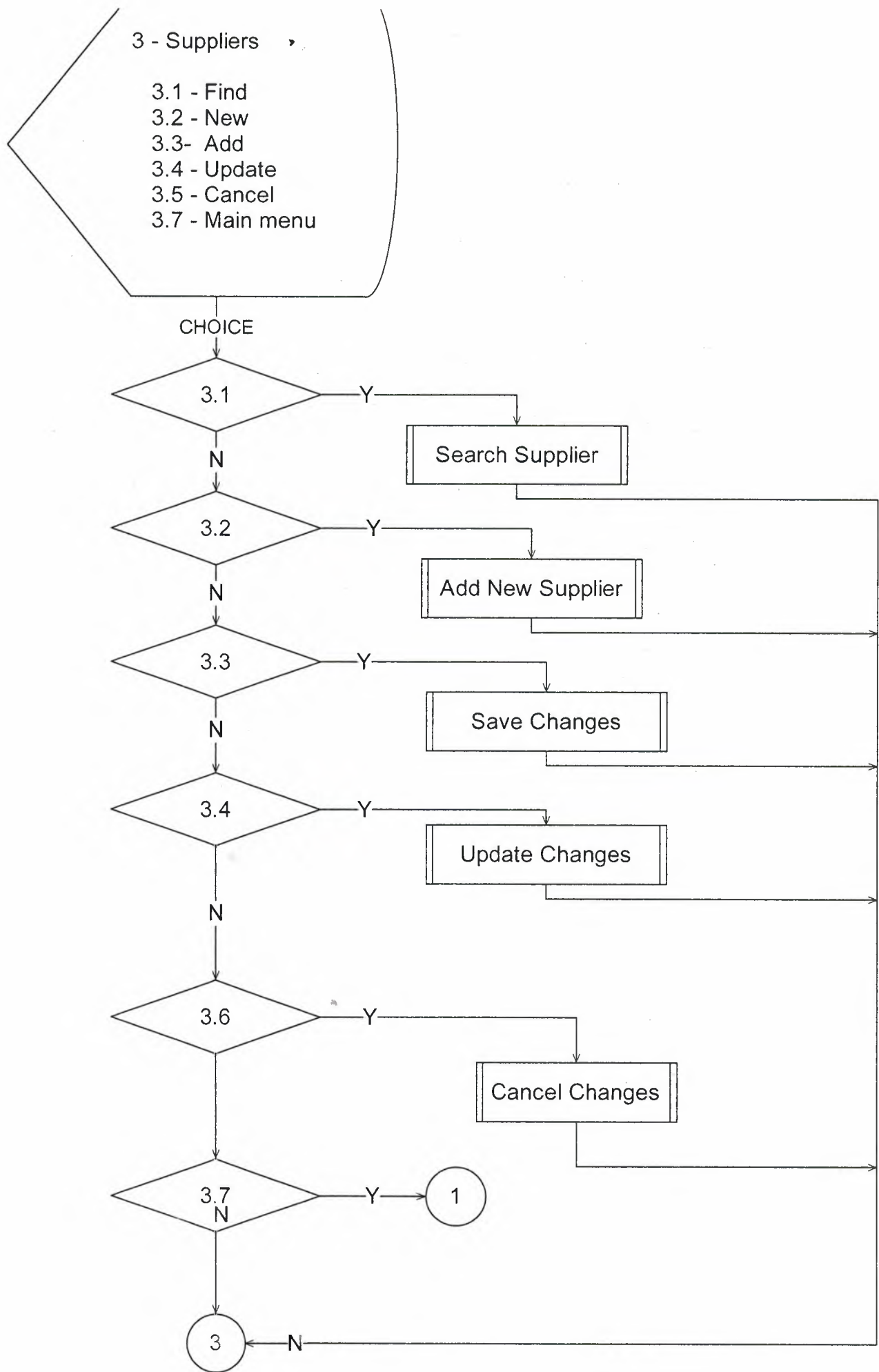
To change program's background select a Picture file from the window and click ok.

USER FLOWCHARTS:



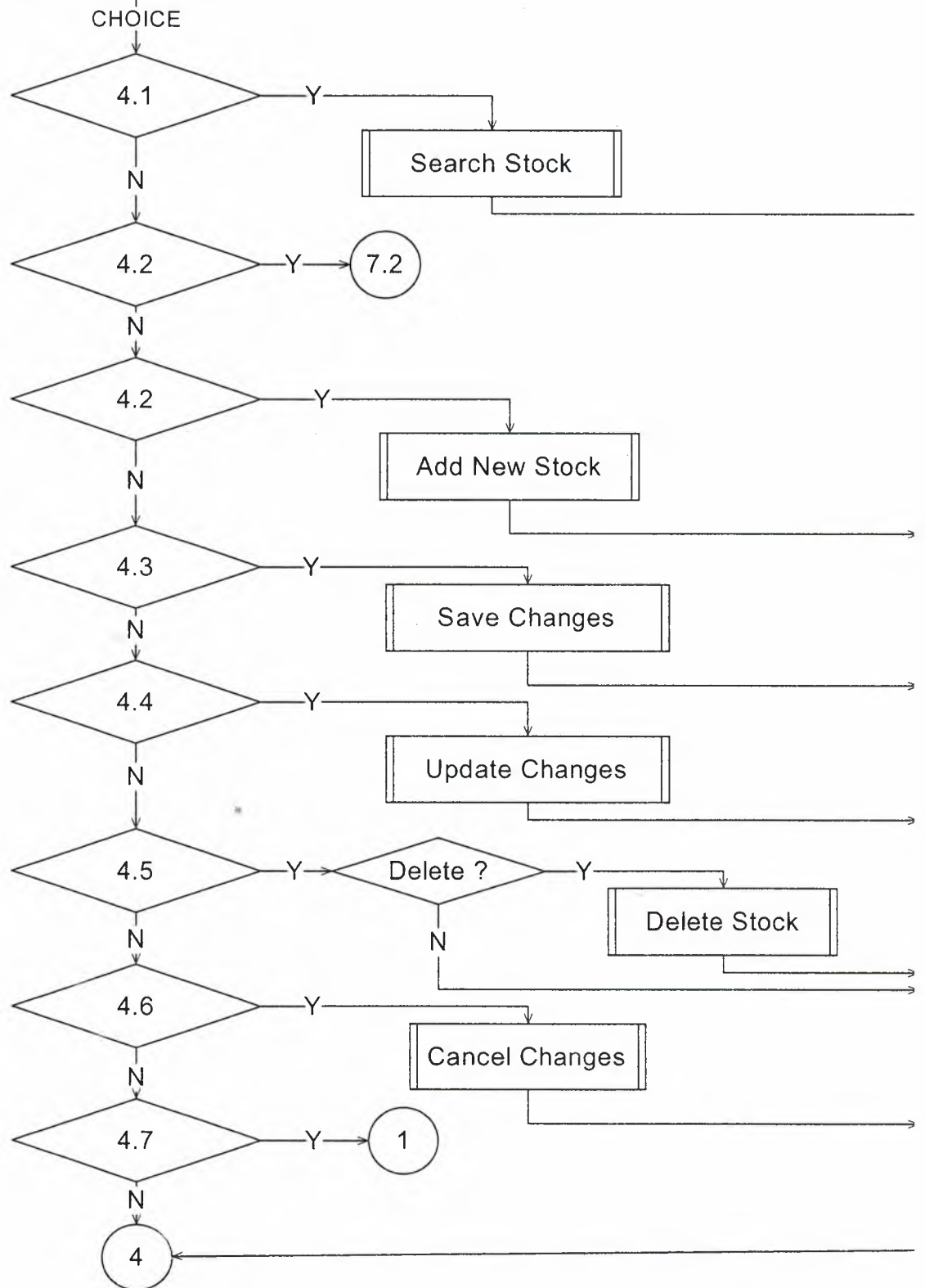


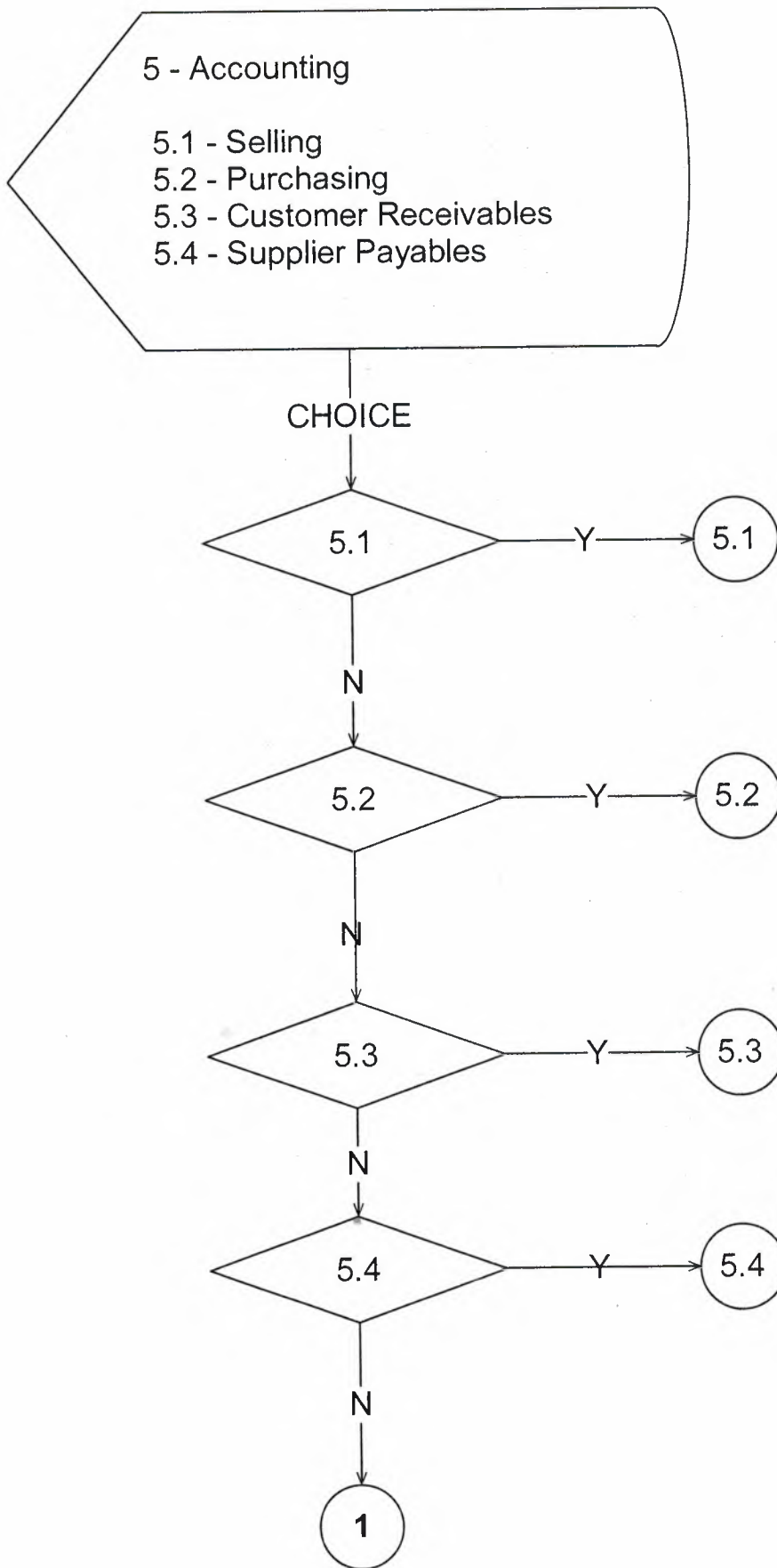




4 - Stocks

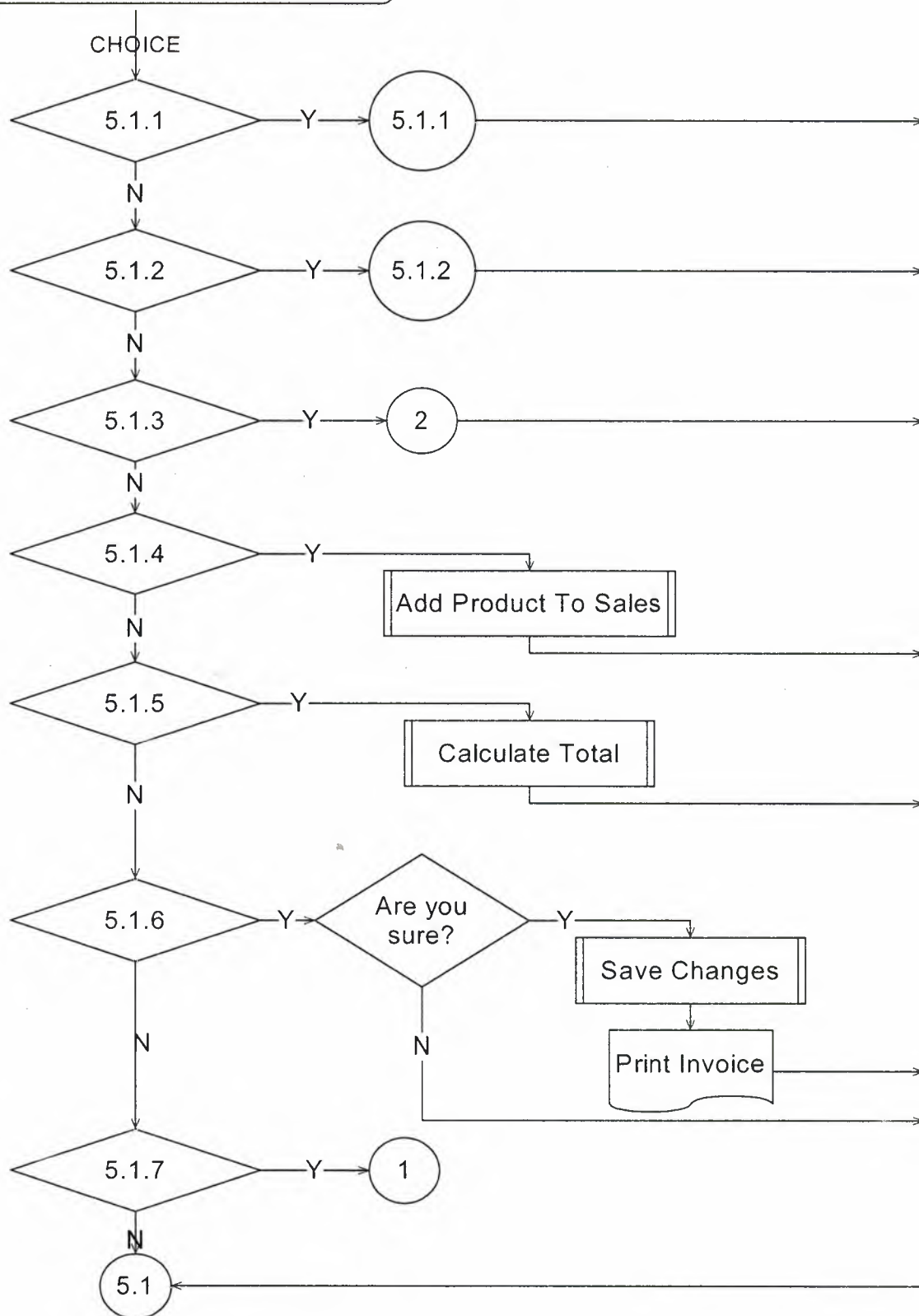
- 4.1 - Find
- 4.2 - Add New Category
- 4.3 - New
- 4.4- Add
- 4.5 - Update
- 4.6 - Delete
- 4.7 - Cancel
- 4.8 - Main menu

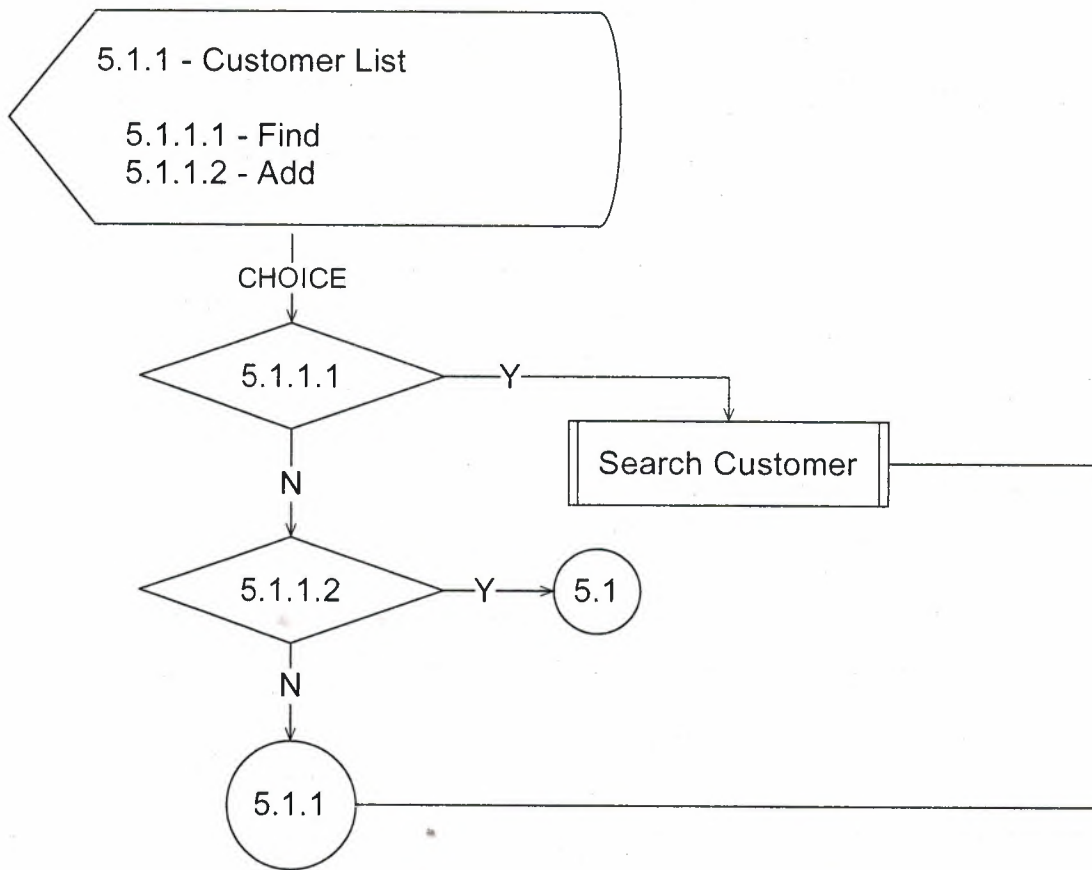


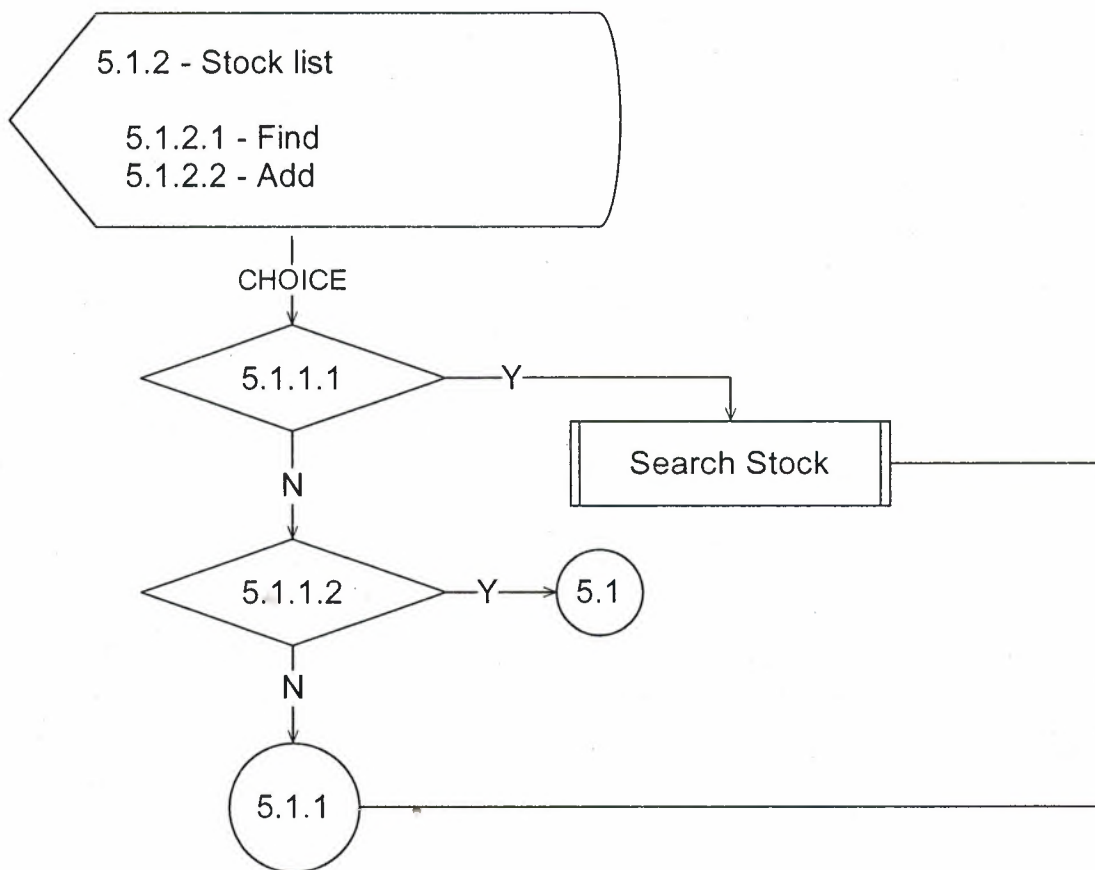


5.1 - Selling

- 5.1.1 - Help
- 5.1.2 - Help
- 5.1.3- Add New Customer
- 5.1.4 - Add To Invoice
- 5.1.5 - Calculate Total
- 5.1.6 - Done
- 5.1.7 - Main Menu

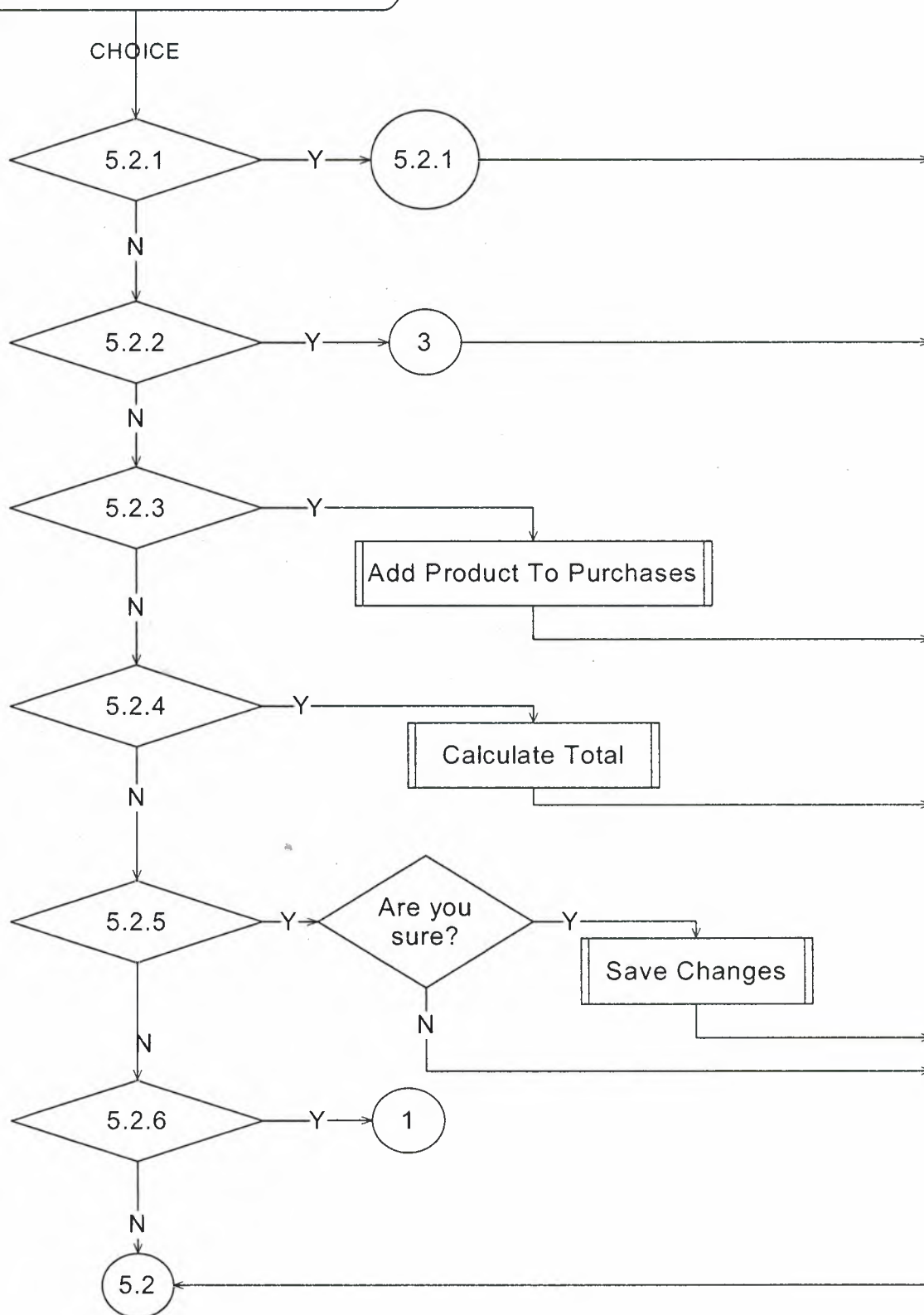


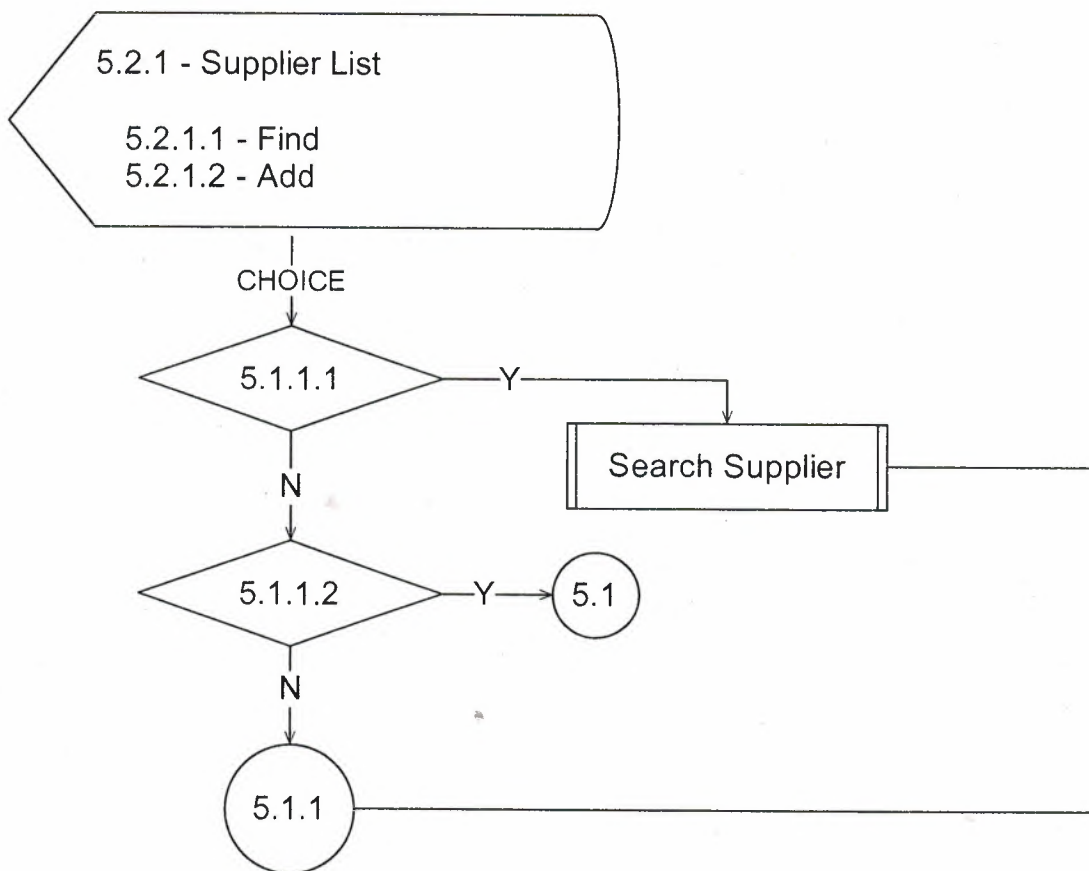


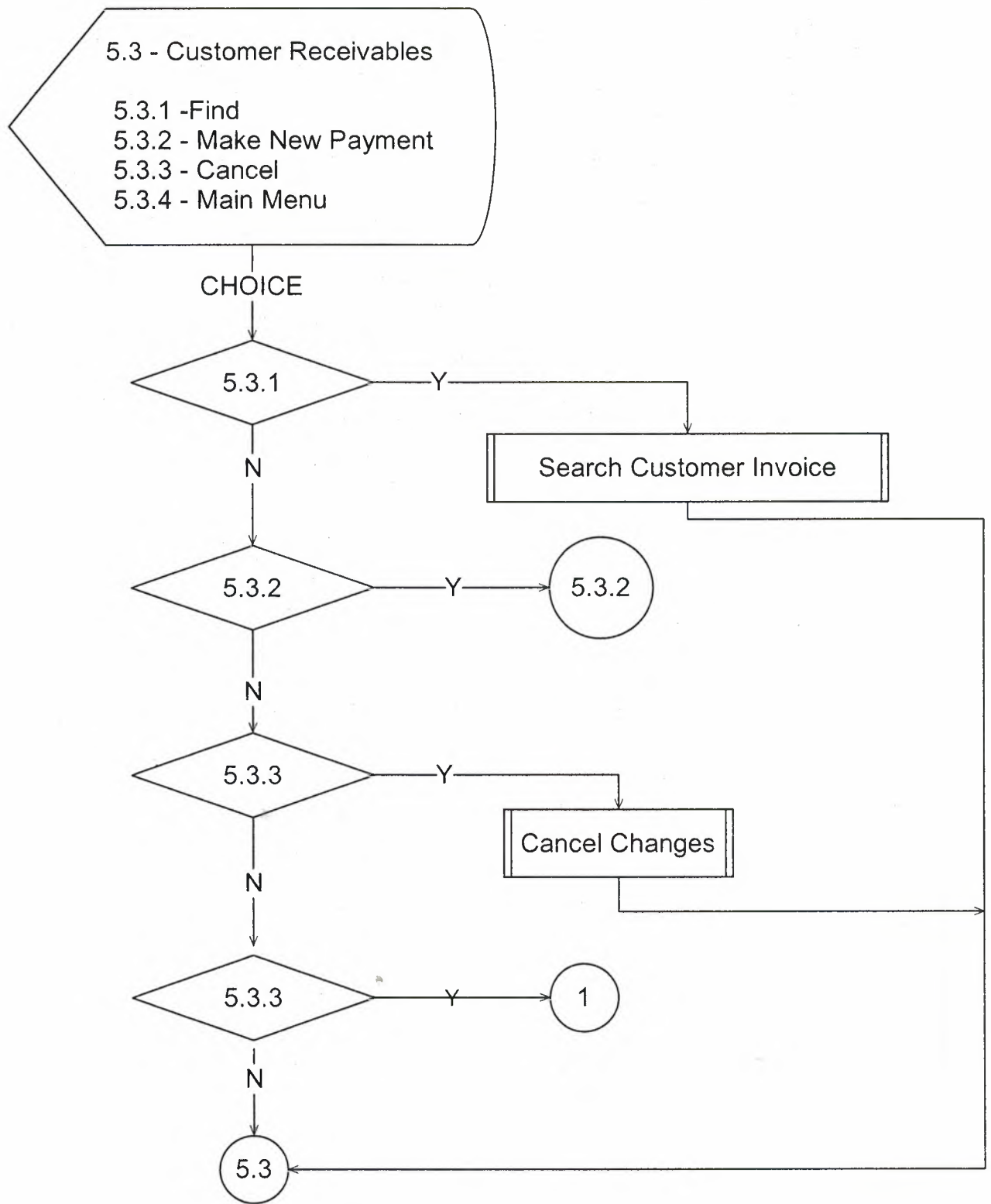


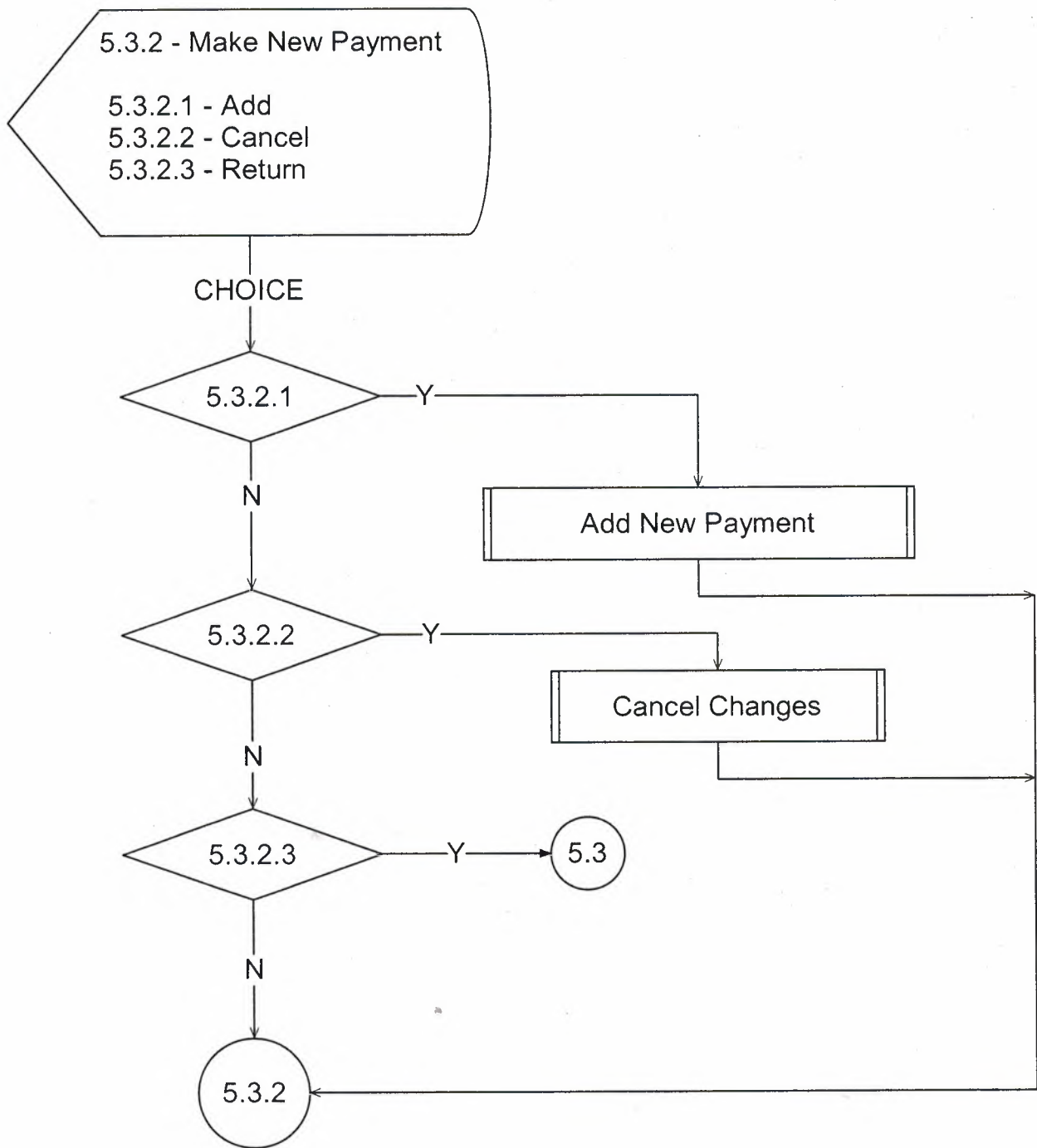
5.2 - Purchasing

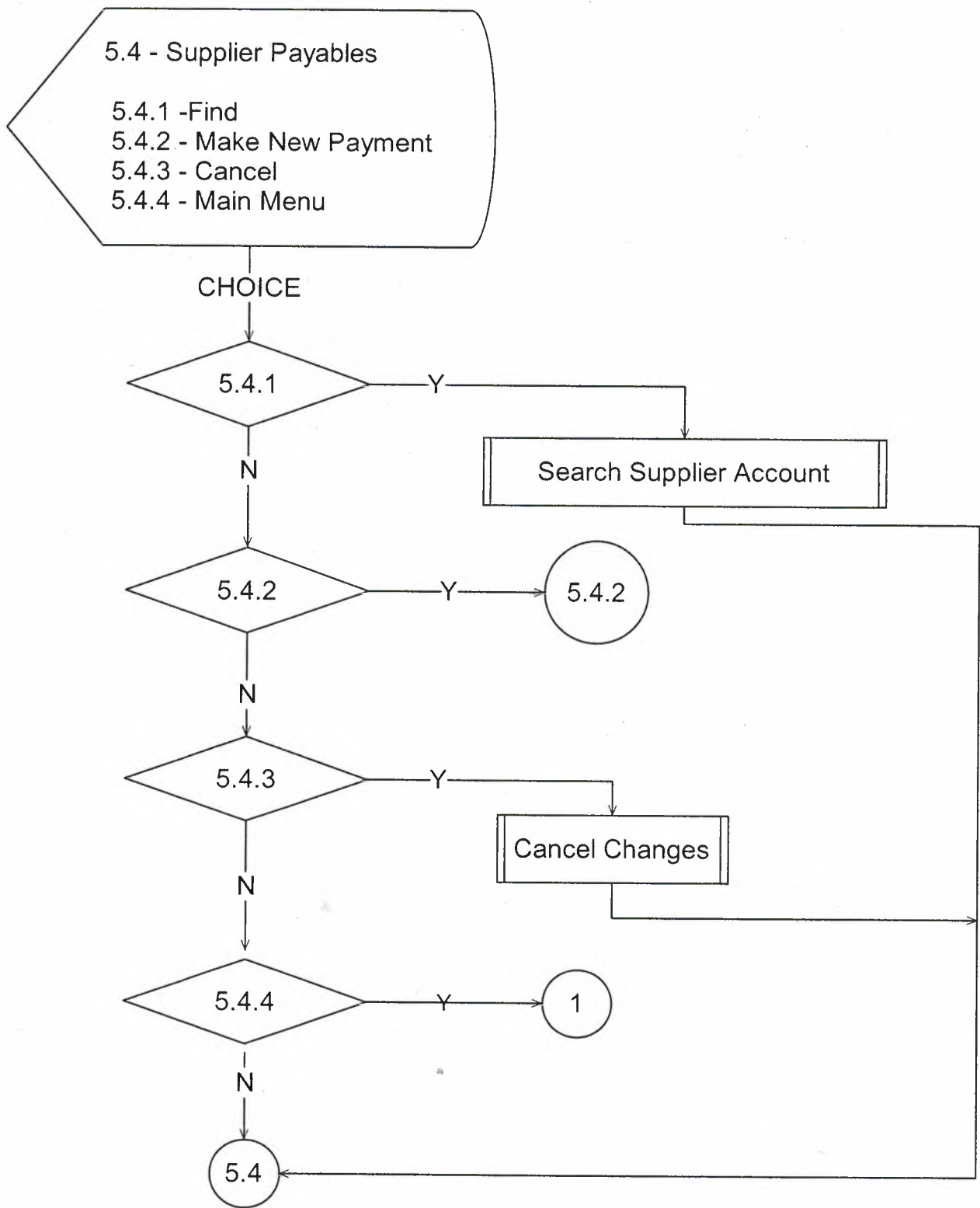
- 5.2.1 - Help
- 5.2.2- Add New Product
- 5.2.3 - Add To Purchases
- 5.2.4 - Calculate Total
- 5.2.5 - Done
- 5.2.6 - Main Menu

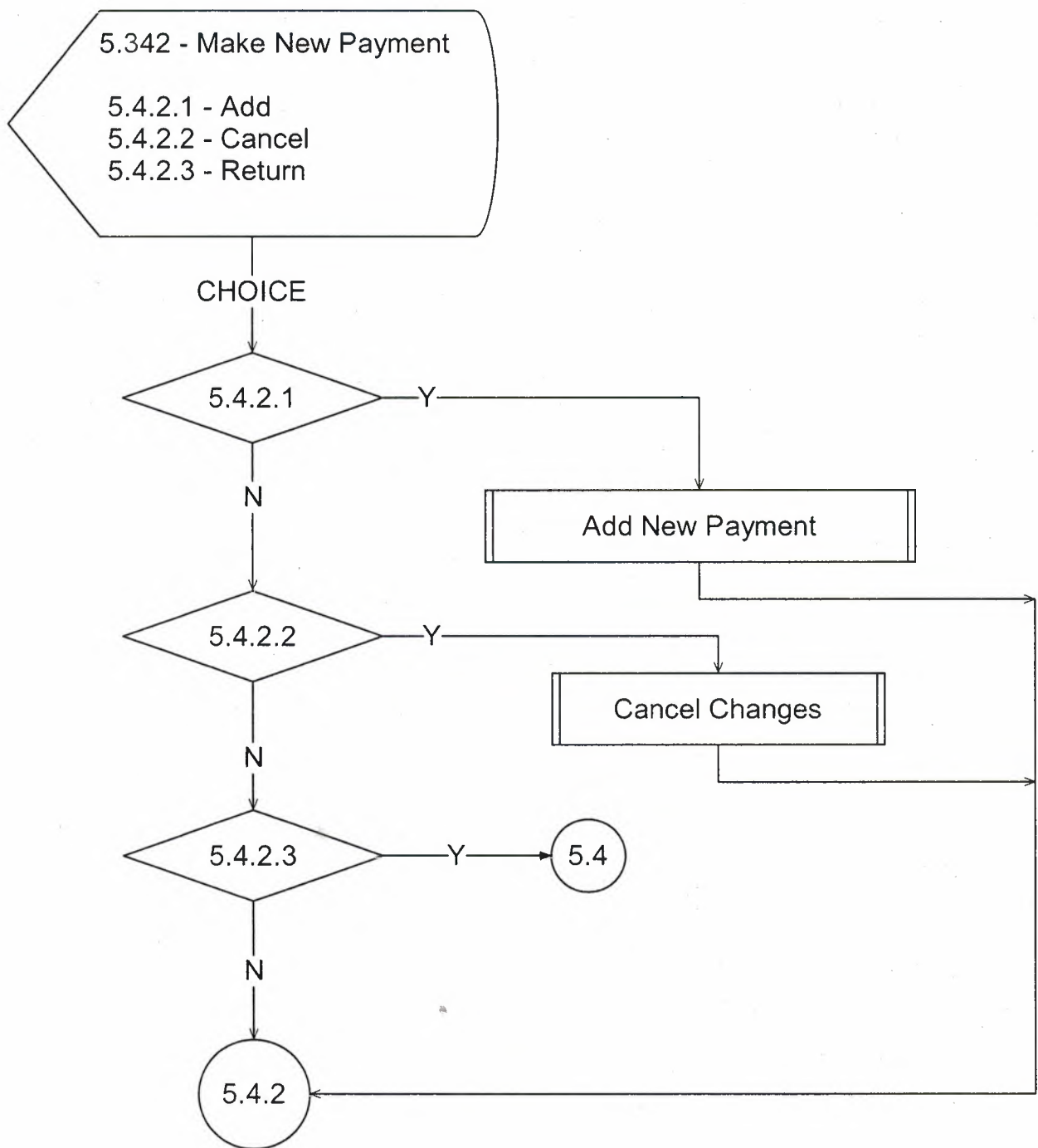


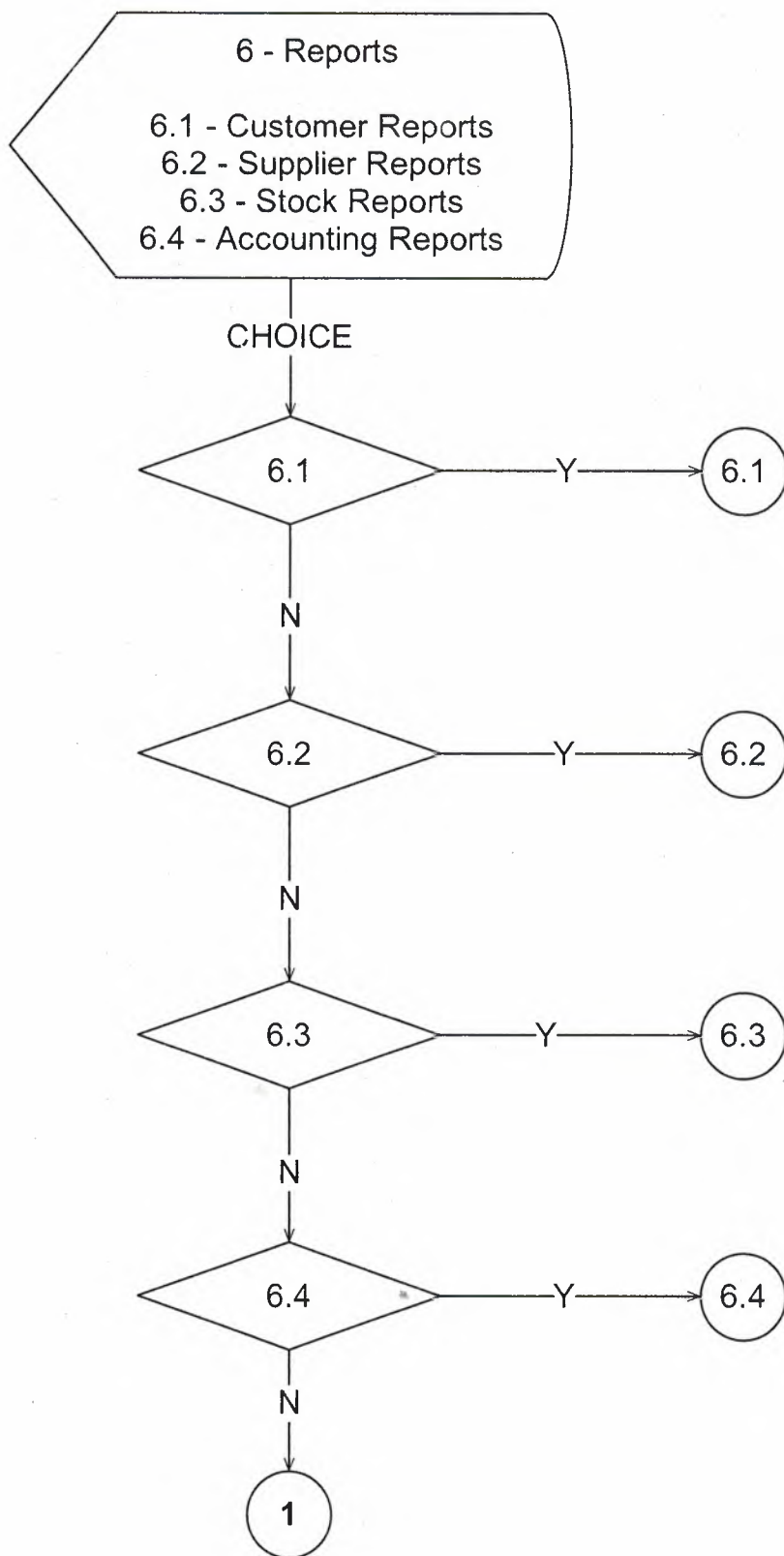


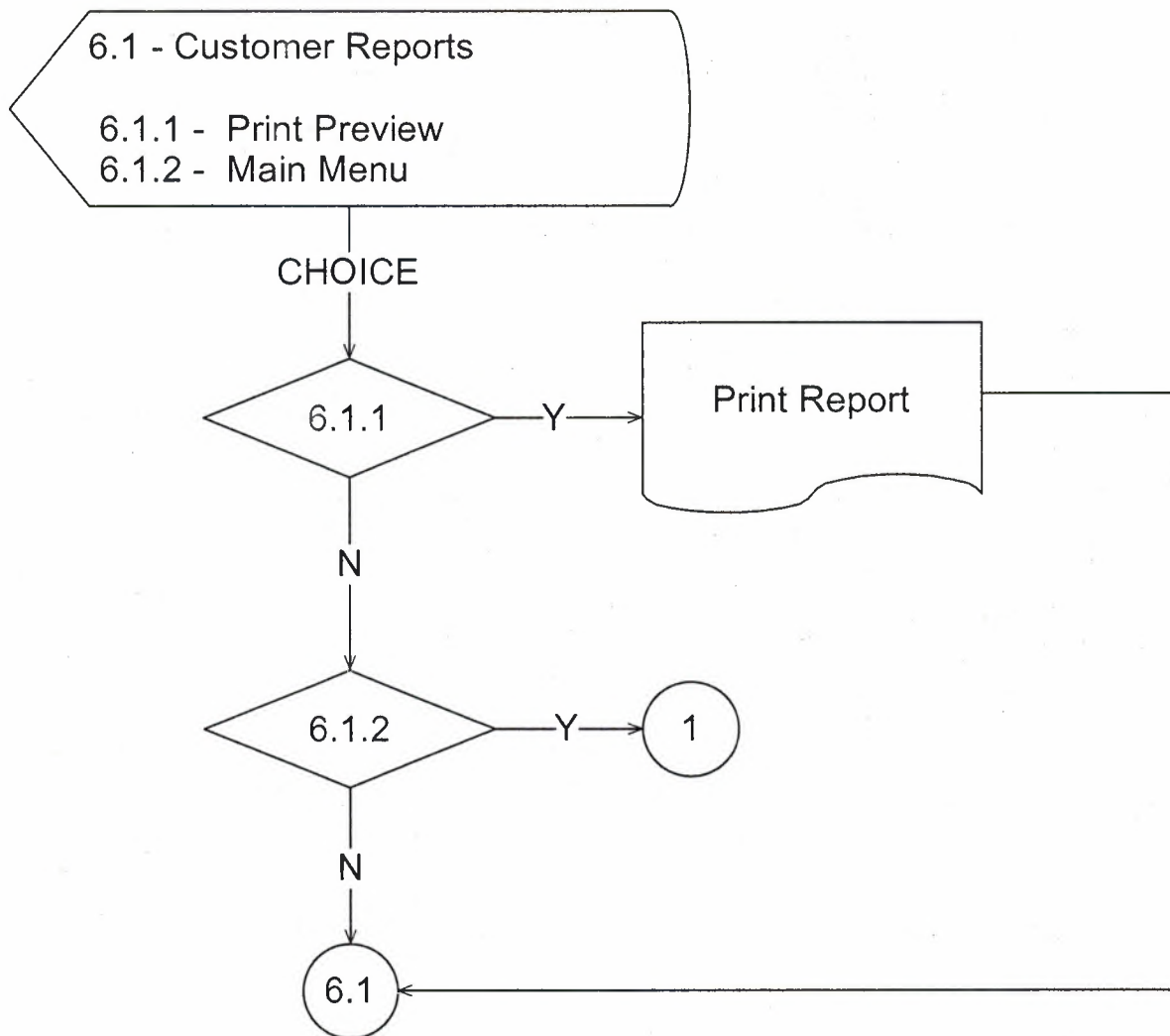


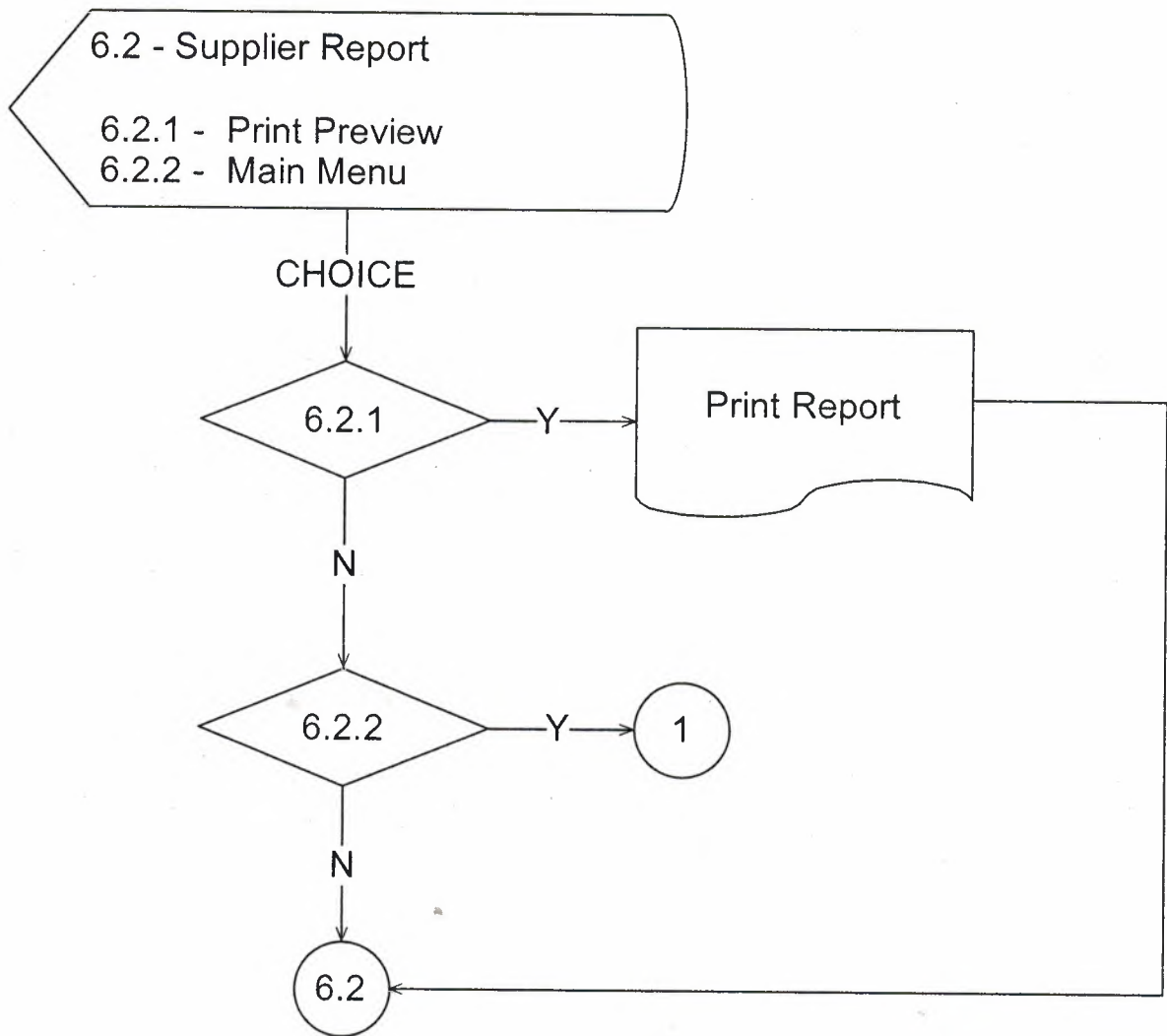


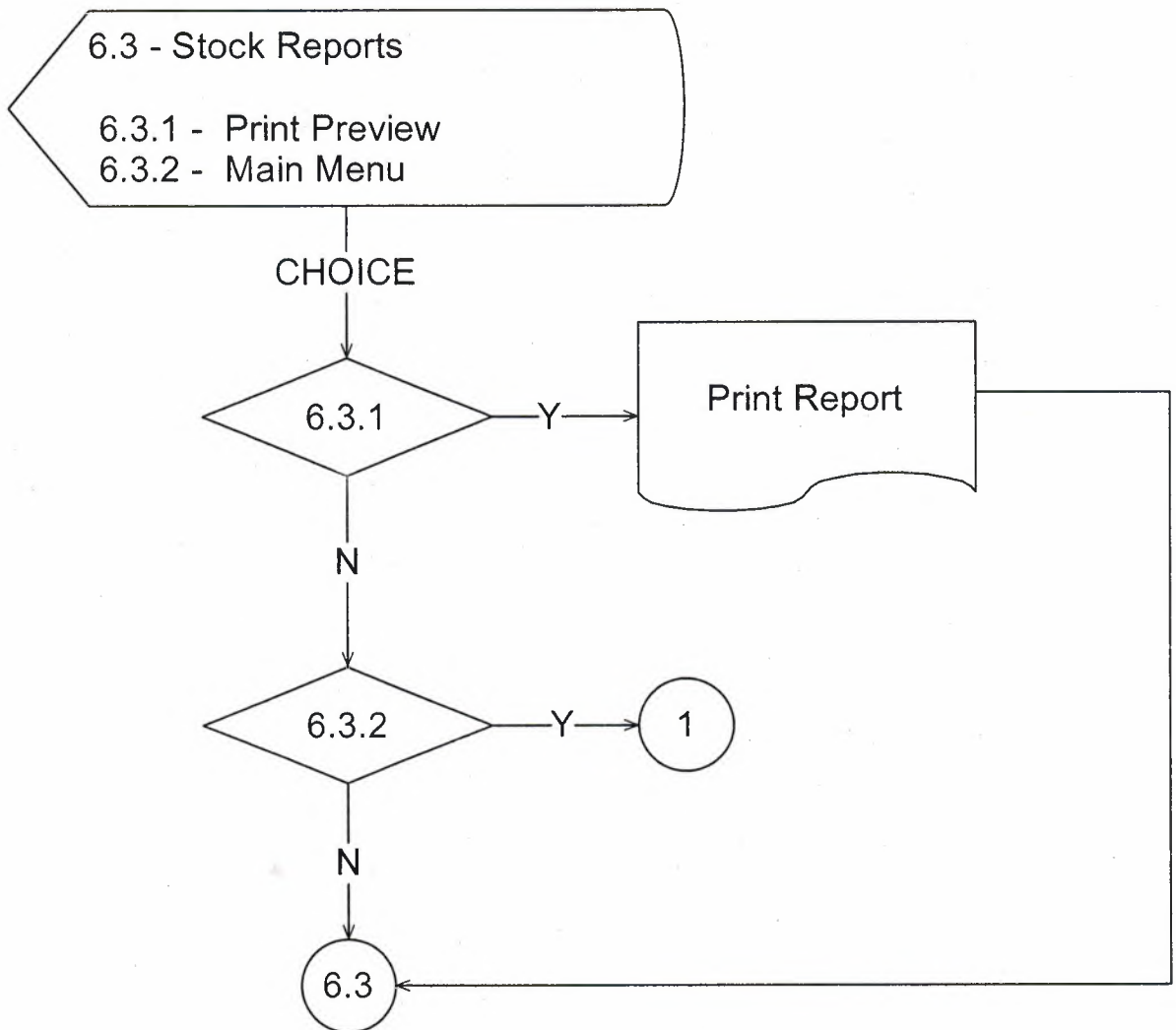


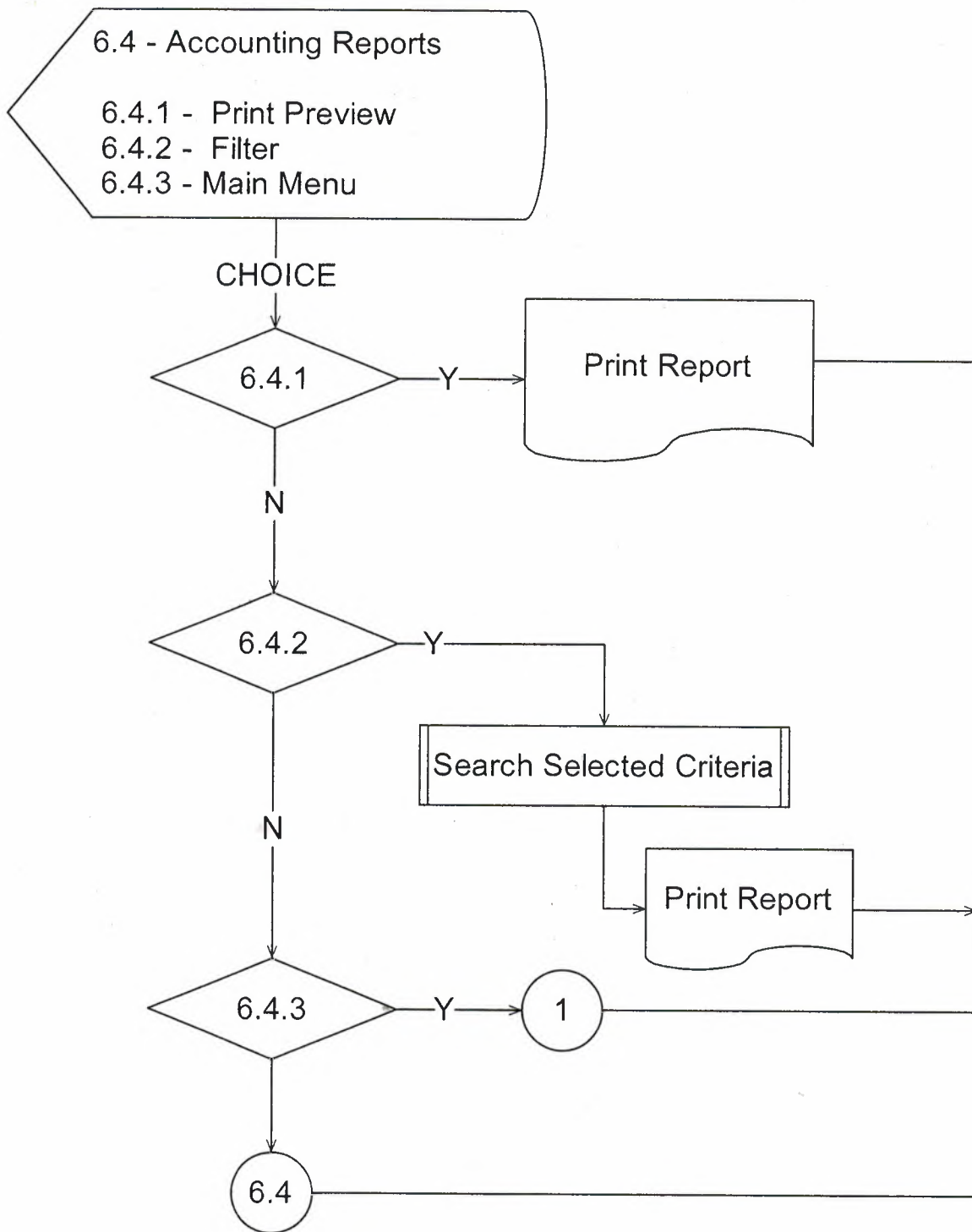


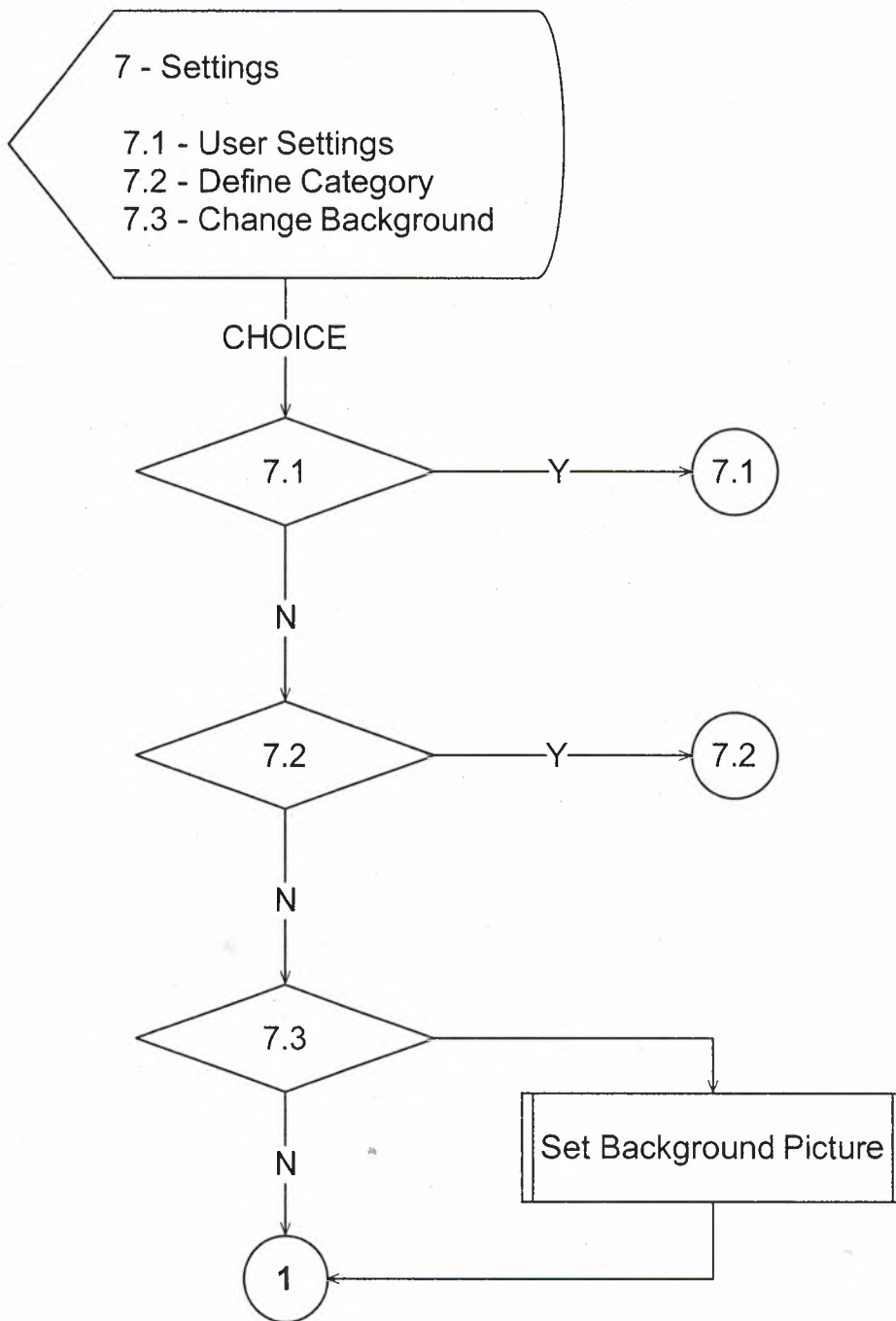


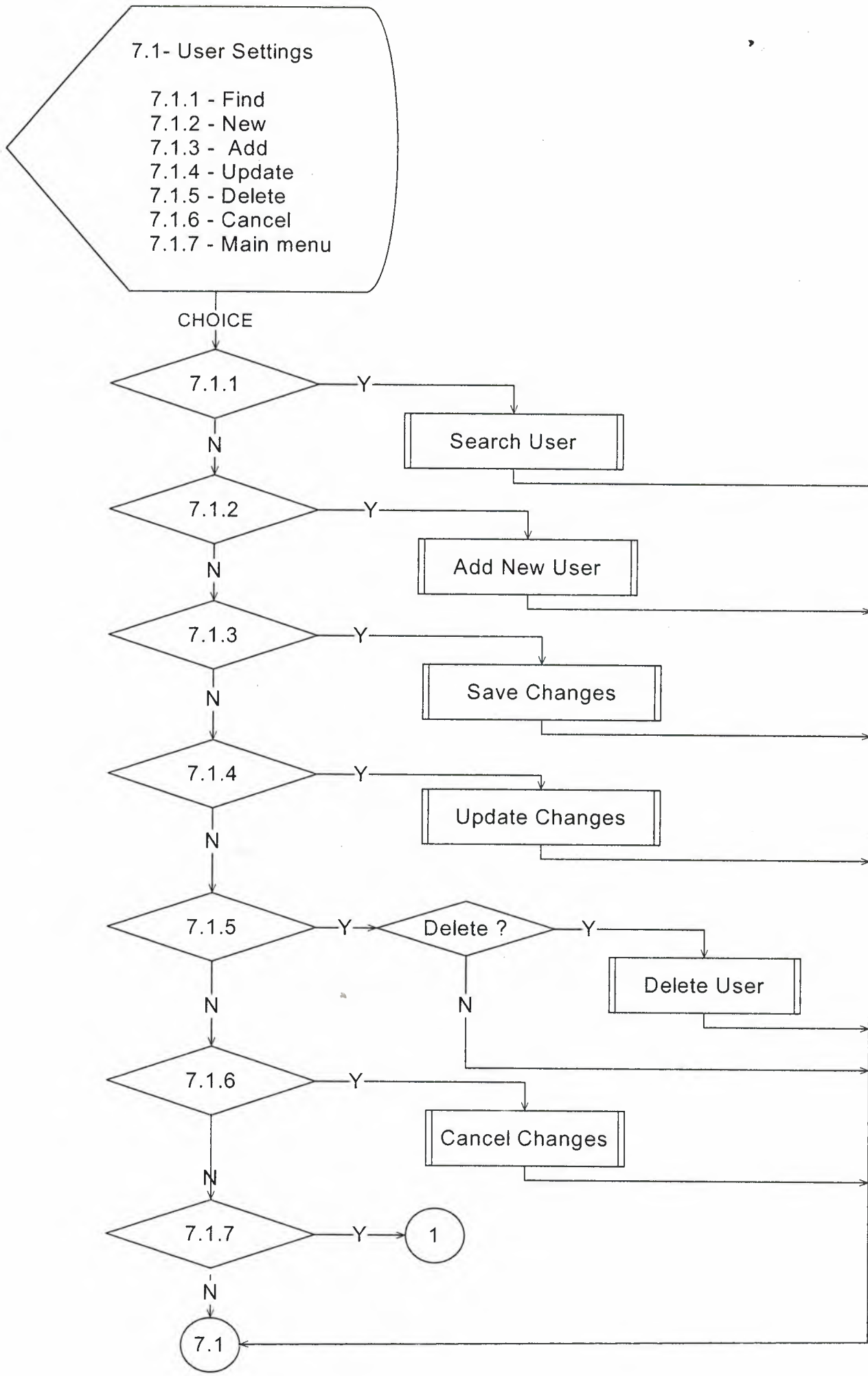


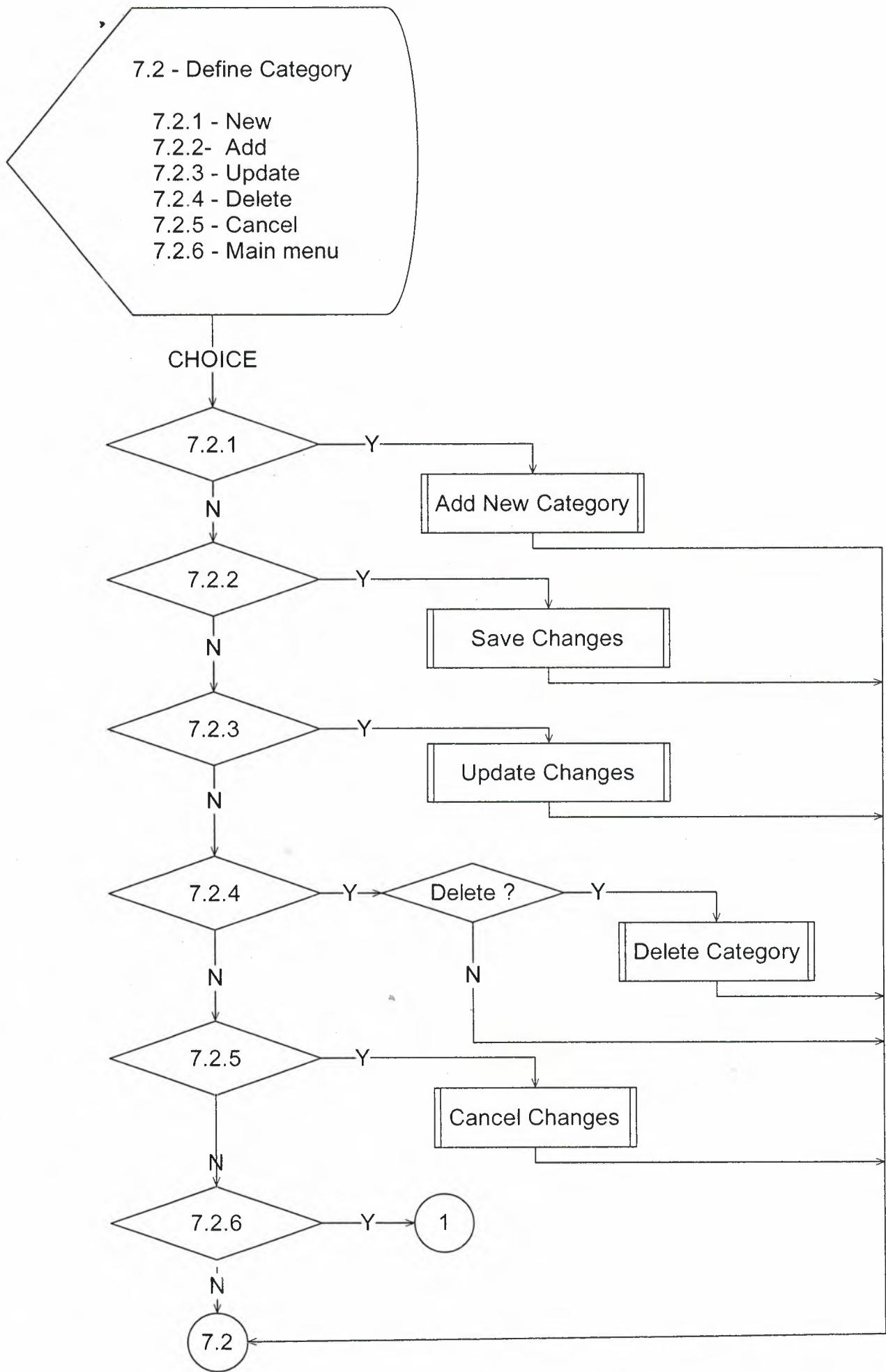












VI. REFERENCES

Books:

Borland Delphi 7 – İhsan Karagülle (Türkmen Kitabevi)

Borland Delphi 7 – Ezel Balkan (Seçkin Yayıncılık)

Borland Delphi ile Veritabanı – İhsan Karagülle & Zeydin Pala (Türkmen Kitabevi)

Web Sites:

www.delphiturk.com

www.programlama.com

www.dwlphidunyasi.com

APPENDIX I.

SCREEN OUTPUTS:

ABOUT FORM:




PASSWORD FORM:





Customers...

CUSTOMERS

Looking For..... Enter Data.....  Find

CustomerNo	Name	Surname	Phone	Mobile	Address
1	cagla	deliceirmak	8223520	05338647950	karaoglanogu
2	ahmet	kara	8152639	05338693214	yukarı girne
3	ayse	beyaz	2296354	05338647941	k. kaymaklı
*					

Navigation: <<< << >> >>>






Customer Number..... Address.....

Name.....

Surname.....

Phone..... City.....

Mobile..... Country.....


 New
 Add
 Update
 Cancel
 Main Menu

Total number of customers..... 3

PRESS F1 TO HELP

Suppliers...

SUPPLIERS

Lookinf for..... Enter Data.....  **Find**

SupplierID	Name	Surname	CompanyName	Phone	Mobile
1	ali	cerene	arcelik	2265789	0533864791
2	selma	kalkan	sony	8156324	0533864521
3	hakan	dalan	beko	8223698	053396321
*	4				

Navigation: <<< << >> >>>

Supplier Number..... 4

Name.....

Surname.....

Company Name.....






Phone.....

Mobile.....

Address.....

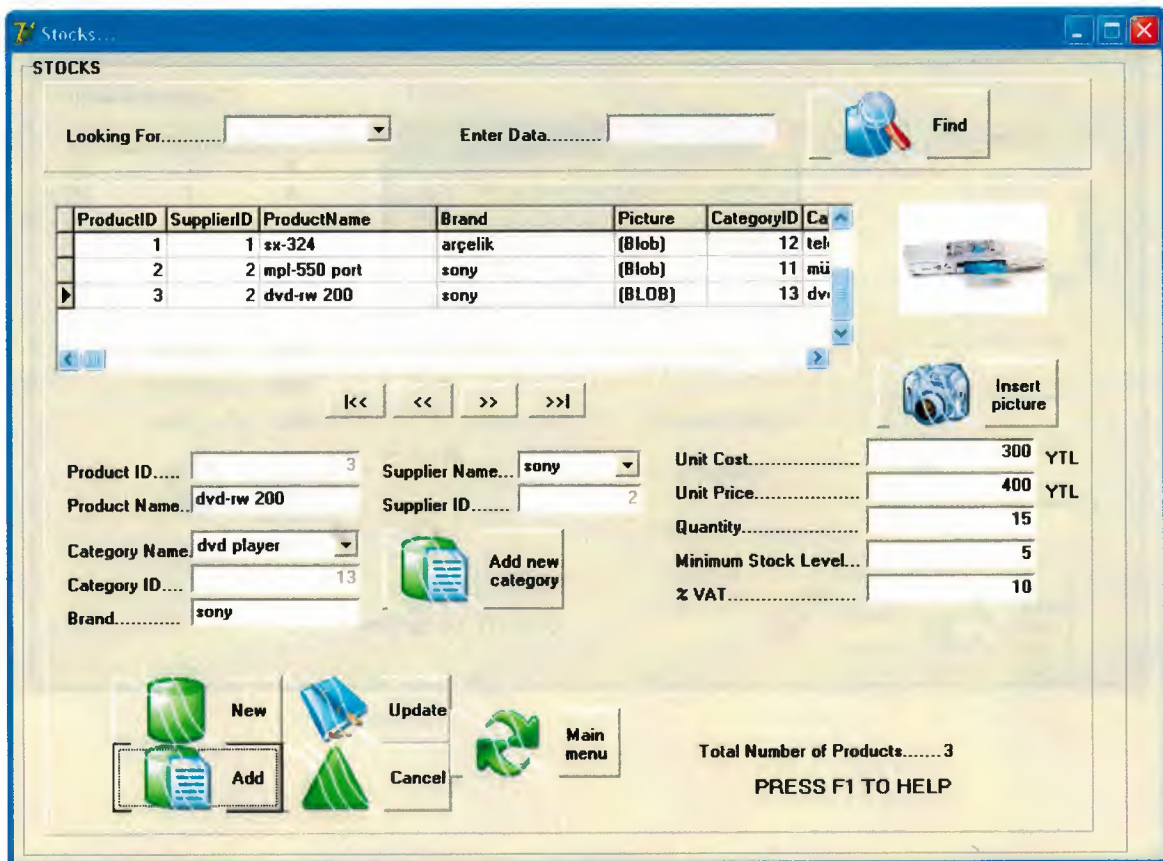
City.....

Country.....

 **New**
 **Update**
 **Add**
 **Cancel**
 **Main menu**

Total number of Suppliers..... 3


PRESS F1 TO HELP



■

Selling...

SELLING

Customer ID..... 1 **help**  28.06.2006

Product ID..... 4 **help**

Quantity..... 2 Stock quantity..... 5 **Cancel Product**

Invoice Number.... 1



CustomerNo	ProductID	InvoiceN	CategoryName	UnitCost
1	5			
1	4			

Confirm Are you sure to save selling information?
Yes No

Sub total..... 5000
Total VAT..... 500
Total..... 5500 **Calculate total**

Payment Type:
☐ Cash
☒ Credit

Number of Installment..... 4
Required Payment..... 1375
First Payment..... 1500
Balance..... 4000

PRESS F1 TO HELP  **Done**  **Main menu**



Print Preview

Close

INVOICE

Invoice Number: 1

Customer Number: 1

Date: 28.06.2006

<u>Product ID</u>	<u>Product Name</u>	<u>Quantity</u>	<u>Total</u>
5	el-sx	3	495
4	home theatre	2	5500
Total:			5500

Page 1 of 1


start

Desktop

Project1

Purchasing...

PURCHASING

Product No..... 5 **help**  **Add new product** 01.06.2006


Supplier Number..... 2

Quantity..... 5 Stock Quantity.....20

Bill Number: 1

ProductID	SupplierID	BillNo	ProductName	CategoryName	UnitCost
2	2	1	20s-rwl	dvd player	
5	2	1	el-sx	müzik seti	



Confirm Are you sure?
Yes No

Total: 950  **Calculate total**

Payment type
☐ Cash
☒ Credit

Number Of Invoice..... 4
 Required Payment..... 237.5
 First Payment..... 250
 Balance..... 700

PRESS F1 TO HELP

 **Buy**  **Main menu**

Customer Receivables...

CUSTOMER RECEIVABLES

Looking For.....

Enter data.....

Find

Customer Account:

InvoiceNo	CustomerNo	Total	Date
1	1	5500	28.06.2006
2	3	1221	28.06.2006
3	2	297	28.06.2006
4	1	2750	28.06.2006

Total Amount: 5500

Total Paid: 2800

Balance: 2700

Label7

Make new payment

Cancel

Main menu

Customer Account Detail:

InvoiceNo	CustomerNo	Date	TotalPrice
1	1	28.06.2006	5500
1	1	21.06.2006 22:30:1	5500

PRESS F1 TO HELP

Customer Receive **Customer Payment...**

CUSTOMER RECEIPT

Looking For.....

Customer Account

InvoiceNo	CustomerNo
1	1
2	1
3	1
4	1




CUSTOMER PAYMENT

Customer Id..... 1
Customer Name..... cagla
Customer Surname..... deliceirmak

Invoice No..... 1
Installment No.....

Date.....
Total.....

Required Payment..... 1350
Payment..... 1500
Remaining payment..... 1200

 Add  Cancel  Return

Confirm

Are You Sure?

Find

5500
2800
2700

Make new payment

Cancel

Main menu

F1 TO HELP

Print Preview

Customer RECEIPT FOR PAYMENT

Customer No: 1

Customer Name: cagla

Customer Surname: deliceirmak

Invoice No: 1

Payment: 1500


Date: 21.06.2006 22:30:13

0% Page 1 of 1

start

Supplier Payables...

SUPPLIER PAYABLES

Looking For..... **choose** Enter data.....  **Find**




Supplier Account

BillNo	SupplierID	Date	Total
1	2	01.06.2006	950
3	2	01.06.2006	500

Total Amount..... 500
Total Paid..... 150
Balance..... 350

Supplier Account Detail

BillNo	SupplierID	Date	Total	Nu
3	2	01.06.2006	500	

 **Make new payment**
 **Cancel**
 **Main menu**

PRESS F1 TO HELP

Supplier Payables
Supplier Payments...

SUPPLIER PAYABLE

Looking For.....

Supplier Account

BillNo	
1	
3	

Supplier Account

BillNo	
3	
3	
3	
3	

SUPPLIER PAYMENT

Customer ID..... 2

Company Name..... sony

Invoice No..... 3

Installment No..... 4

Date..... 21.06.2006

Total..... 500

Required Payment.... 150

Payment..... 150

Remaining payment.. 0

Add
Cancel
Return

Find

unt..... 500

..... 550

..... -50

Make new payment

Cancel

Main menu

S F1 TO HELP

Electronic Shop Automation System Main Menu

Customer Reports...

CUSTOMER LIST

CustomerNo	Name	Surname	Phone	Mobile	Address
1	cagla	deliceirmak	8223520	05338647950	karaoglanogu
2	ahmet	kara	8152639	05338693214	yukarı girne
3	ayse	beyaz	2296354	05338647941	k. kaymaklı

Print preview

Main menu

Total found.....

PRESS F1 TO HELP

payments

PRESS F1 TO HELP

Print Preview

Close

ALL CUSTOMERS REPORT

<u>Name</u>	<u>Surname</u>	<u>Customer no</u>	<u>Address</u>	<u>City</u>	<u>Phone</u>
cagla	deliceirmak	1	karaoglanoglu	gime	8223520
ahmet	kara	2	yukan gime	gime	8152698
hatice	hoca	3	karaoglanoglu	gime	8223629

Page 1 of 1

start

Desktop

Printed

Screen 2.1

My Documents

7: Page 1

Electronic Shop Automation System Main Menu...

Supplier Reports...

SUPPLIER LIST

SupplierID	Name	Surname	CompanyName	Phone	Mob
1	ali	kara	arçelik	2265897	0533
2	selin	sari	sony	05338647951	8158
3	murat	tepe	beko	02128694563	0532

Print preview

Main menu

Total Count.....

PRESS F1 TO HELP

payments

PRESS F1 TO HELP

10

Print Preview


Close

ALL SUPPLIERS REPORT

<u>Company Name</u>	<u>Supplier ID</u>	<u>Name</u>	<u>Surname</u>	<u>Address</u>	<u>City</u>	<u>Phone</u>
arçelik	1	ali	kara	dereboyu	tefkosa	2265887
sony	2	selin	sari	gime merkez	gime	05338647951
beko	3	murat	tepe	yukan ayranci	ankara	02128694563

Page 1 of 1

start




Stock Reports...

STOCK REPORTS

Reports Choices

☒ All stocks
 ☐ Stocks Under Minimum Stock Level

ProductID	SupplierID	ProductName	Brand	Picture	CategoryID	Category
1	2	PS2	sony	(Blob)	4	play stat
2	2	20s-rw1	sony	(BLOB)	1	dvd play
3	1	kv-32	arçelik	(BLOB)	2	televizyc
4	1	home theatre	beko	(BLOB)	1	dvd play


Print preview

Main menu

Total..... 5
PRESS F1 TO HELP

Print Preview

Close

STOCK REPORT

All

<u>Product Name</u>	<u>Product ID</u>	<u>Brand</u>	<u>Category</u>	<u>Supplier</u>	<u>Quantity</u>	<u>Minimum Stock Level</u>
PS2	1	sony	play station	2	18	5
20s-rwl	2	sony	dvd player	2	8	6
kv-32	3	arçelik	televizyon	1	10	5
home theatre	4	beko	dvd player	1	2	10
el-sx	5	sony	müzik seti	2	30	10

Page 1 of 1

start

Print Preview

STOCKS UNDER MINIMUM STOCK LEVEL REPORT

<u>Product No</u>	<u>Product Name</u>	<u>Brand</u>	<u>Category</u>	<u>Supplier ID</u>	<u>Quantity</u>
4	home theatre	beko	dvd player	1	2

Page 1 of 1

start

7 Project 1

Accounting Reports...

ACCOUNTING REPORTS

Report Choices


☒ **Customer Payments**
☐ All Sales

☐ Supplier Payments
 ☐ All Purchases



Customer Select

Looking For.....

Choose.....

 **Filter**

c.CustomerNo	Name	Surname	Phone	Mobile	Address
1	cagla	deliceirmak	8223520	05338647950	karaoglanoglu
1	cagla	deliceirmak	8223520	05338647950	karaoglanoglu
1	cagla	deliceirmak	8223520	05338647950	karaoglanoglu
1	cagla	deliceirmak	8223520	05338647950	karaoglanoglu
3	hatice	hoca	8223629	05338627282	karaoglanoglu

 **Print preview**
 **Main menu**

Total..... 5

PRESS F1 TO HELP

Print Preview

CUSTOMER PAYMENTS

<u>Customer Name</u>	<u>Customer Surname</u>	<u>Customer ID</u>	<u>Address</u>	<u>City</u>	<u>Phone</u>
cagla	deliceirmak	1	karaoglanoglu	gime	8223520

<u>Invoice No</u>	<u>Date</u>	<u>Payment</u>	<u>Required Payment</u>
1	21.06.2006 22:30:13	1500	1200
1	21.06.2006 22:30:13	1300	1350
4	28.06.2006	600	550
1	28.06.2006	1500	1375
<u>Total</u>			

Page 1 of 1

start

Print Preview

Close

SUPPLIER PAYMENTS						
<u>Company Name</u>	<u>Supplier ID</u>	<u>Name</u>	<u>Surname</u>	<u>Address</u>	<u>City</u>	<u>Phone</u>
sony		selin	sarı	gime merkez	girne	05338647951
		<u>Bill No</u>	<u>Date</u>	<u>Payment</u>	<u>Required Payment</u>	
		3	21.06.2006 22:30:13	150	0	
		3	21.06.2006 22:30:13	100	150	
		3	21.06.2006 22:30:13	100	150	
		3	21.06.2006 22:30:13	100	125	
		3	21.06.2006 22:30:13	100	125	
		3	01.06.2006	150	125	
		1	01.06.2006	250	237,5	
				<u>Total</u>		

Page 1 of 1

start

Copy, Paste, Undo, Redo, Print, Close

Deploy, Uninstall, Screen Lock, Windows, Power Off



Print Preview

ALL SALES

<u>Invoice No</u>	<u>Product Name</u>	<u>Brand</u>	<u>Category</u>	<u>Product No</u>	<u>Quantity</u>	<u>Date</u>	<u>Total</u>
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
2	20s-rw1	sony		2	3	28.06.2006	726
2	el-sx	sony		5	3	28.06.2006	495
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
2	20s-rw1	sony		2	3	28.06.2006	726
2	el-sx	sony		5	3	28.06.2006	495
3	PS2	sony		1	1	28.06.2006	297

Page 1 of 1

start

Copy (9)

3

Display

Untitled

Screening

My Icons

Project

TP

Links

Print Preview

Close

ALL SALES BETWEEN TWO DATE

<u>Invoice No</u>	<u>Product Name</u>	<u>Brand</u>	<u>Category</u>	<u>Product No</u>	<u>Quantity</u>	<u>Date</u>	<u>Total</u>
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
2	20s-rw1	sony		2	3	28.06.2006	726
2	el-sx	sony		5	3	28.06.2006	495
1	el-sx	sony		5	3	28.06.2006	495
1	home theatre	beko		4	2	28.06.2006	5500
2	20s-rw1	sony		2	3	28.06.2006	726
2	el-sx	sony		5	3	28.06.2006	495
3	PS2	sony		1	1	28.06.2006	297

Page 1 of 1

start

Copy (9)

Google T

Untitled -

Screen 100

Windows

Project 1

Tip

100%

Print Preview

Close

ALL PURCHASES REPORT

<u>Bill No</u>	<u>Product Name</u>	<u>Brand</u>	<u>Category</u>	<u>Product no</u>	<u>Quantity</u>	<u>Date</u>	<u>Total</u>
1	el-sx	sony	müzik seti	5	5	01.06.2006	500
1	20s-rw1	sony	dvd player	2	3	01.06.2006	450
2	20s-rw1	sony	dvd player	2	3	01.06.2006	450
2	20s-rw1	sony	dvd player	2	3	01.06.2006	450
2	20s-rw1	sony	dvd player	2	3	01.06.2006	450
3	el-sx	sony	müzik seti	5	5	01.06.2006	500
3	el-sx	sony	müzik seti	5	5	01.06.2006	500
4	el-sx	sony	müzik seti	5	3	01.06.2006	300
Total Number 8					Total	3600	

Page 1 of 1

start

ALL PURCHASES BETWEEN TWO DATE							
<u>Bill No</u>	<u>Product Name</u>	<u>Brand</u>	<u>Category</u>	<u>Product no</u>	<u>Quantity</u>	<u>Date</u>	<u>Total</u>
1	el-sx	sony	müzik seti	5	5	01.06.2006500	
1	20s-rw1	sony	dvd player	2	3	01.06.2006450	
2	20s-rw1	sony	dvd player	2	3	01.06.2006450	
2	20s-rw1	sony	dvd player	2	3	01.06.2006450	
2	20s-rw1	sony	dvd player	2	3	01.06.2006450	
3	el-sx	sony	müzik seti	5	5	01.06.2006500	
3	el-sx	sony	müzik seti	5	5	01.06.2006500	
4	el-sx	sony	müzik seti	5	3	01.06.2006300	
Total Number 8					Total		3600

7 Users...

USERS

Looking For..... Enter Data.....  Find

UserID	Password	User_Name	User_Surname	Title	Address
1	admin	admin	admin	admin	admin

Navigation: << < > >>

User Id.....

Password.....

Name.....

Surname.....

Title.....

Address.....

City.....

Phone.....

Mobile.....

Users Accesibility

- ☒ Customer Proces
- ☒ Suppliers Process
- ☒ Stocks Process
- ☒ Accounting Process
- ☒ Reports Process
- ☒ Settings Proces

 New

 Update

 Cancel

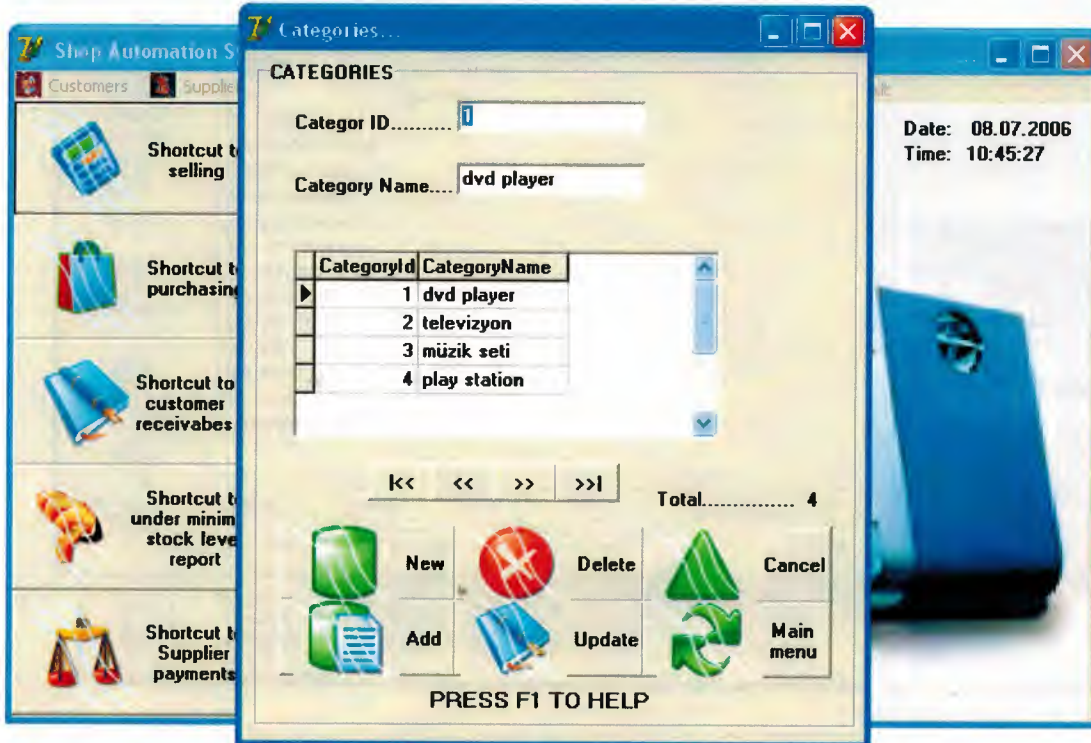
 Add

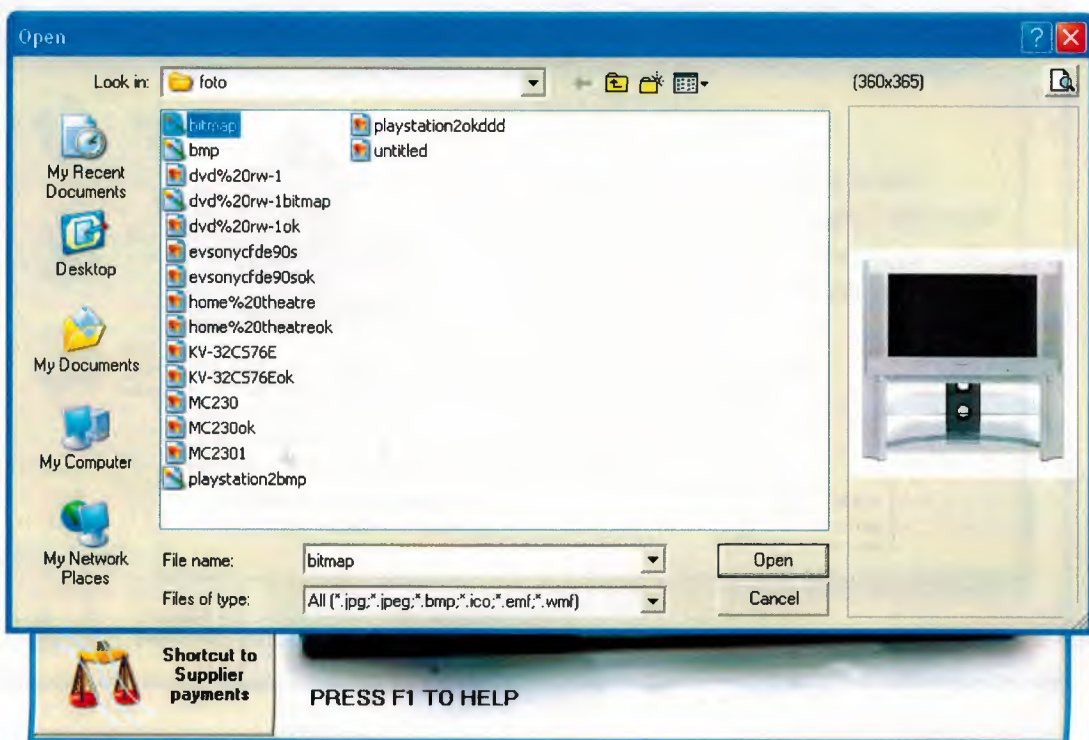
 Delete

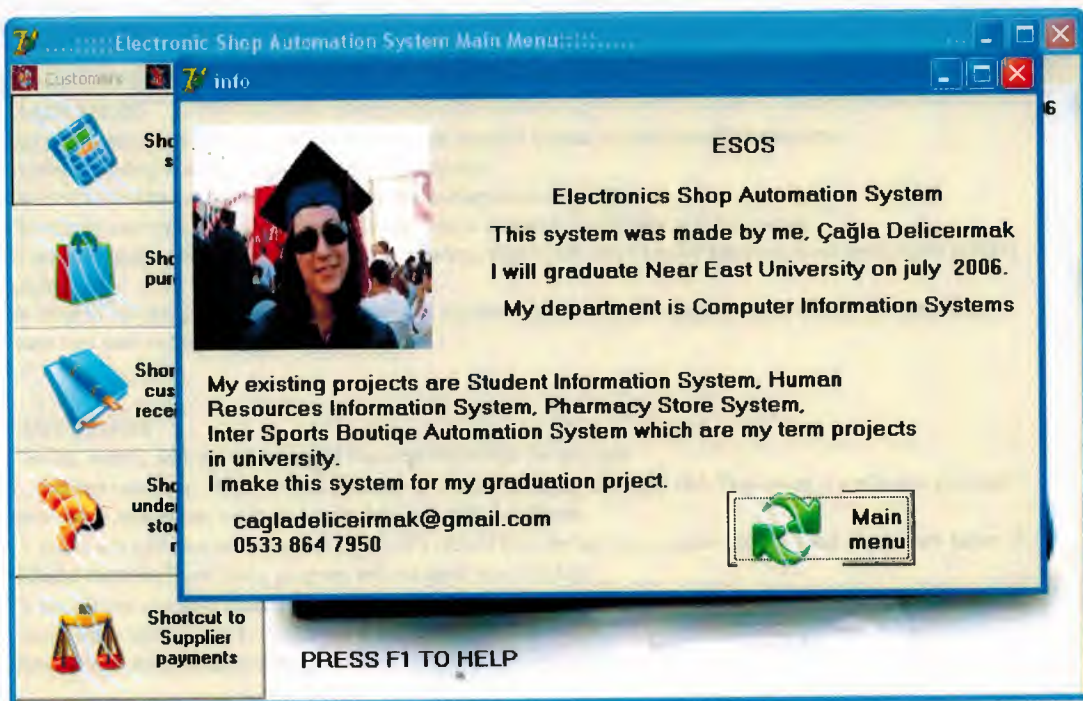
 Main menu

Total..... 1

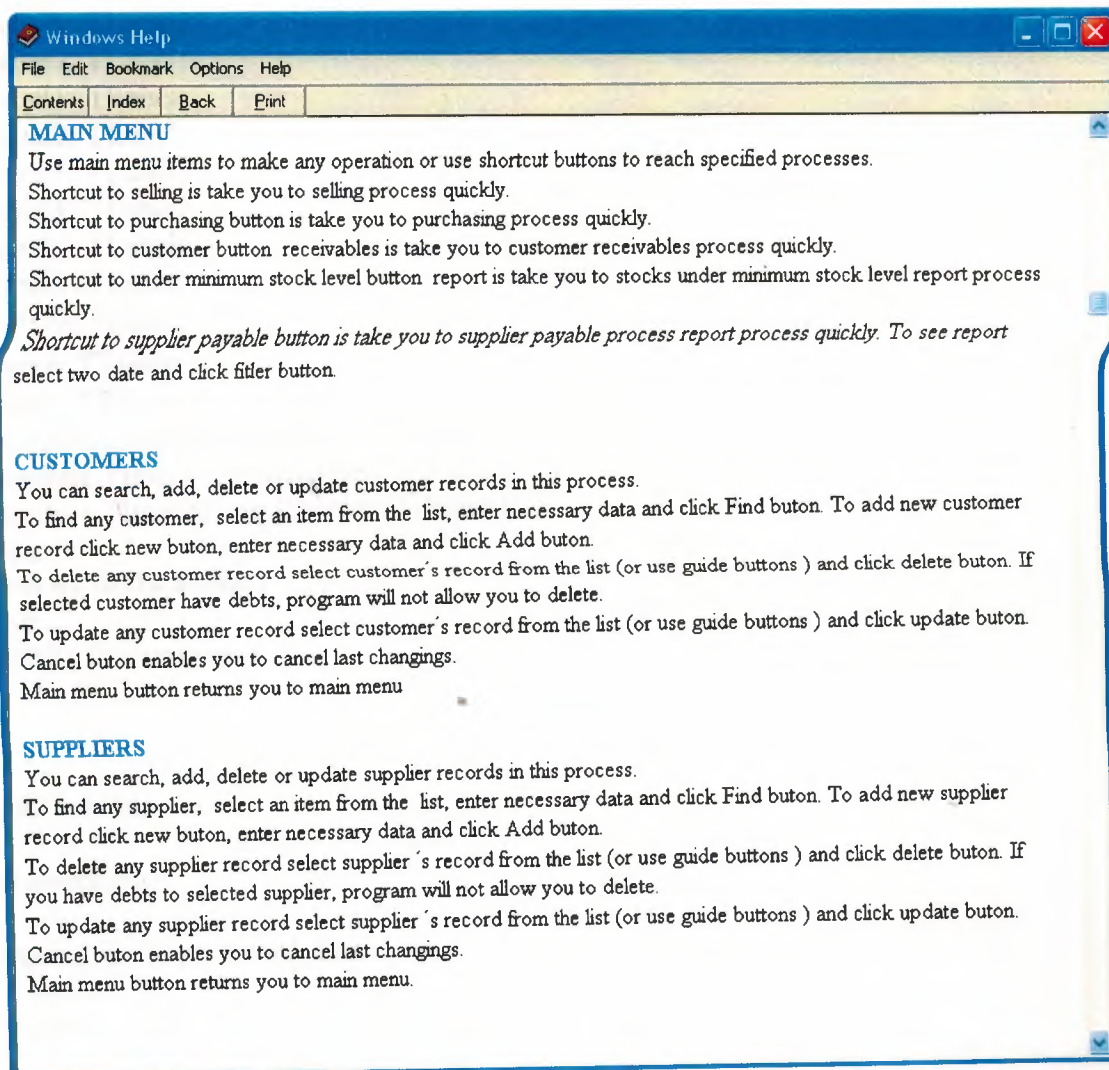
PRESS F1 TO HELP







H



APPENDIX II.

SOURCE CODES:

ABOUT FORM:

```
unit ab;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls;

type
  Tabout = class(TForm)
    Image1: TImage;
    Label6: TLabel;
    Label7: TLabel;
    Label1: TLabel;
    Label2: TLabel;
    Label4: TLabel;
    Label3: TLabel;
    Label5: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Timer1: TTimer;
    Timer2: TTimer;
    procedure FormCreate(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  about: Tabout;

implementation

uses main_menu, password_form;

{$R *.dfm}

procedure Tabout.FormCreate(Sender: TObject);
var
  region: HRgn;
begin
```

```

region:=CreateEllipticRgn(100,100,500,500);
SetWindowRgn(handle, region, true);
end;

```

```

procedure Tabout.Timer1Timer(Sender: TObject);
begin
password.Show;
about.Visible:=false;
timer1.Enabled:=false;
end;

```

```

procedure Tabout.Timer2Timer(Sender: TObject);
begin
label6.Caption:=copy(label6.Caption,2,length(label6.Caption)-1)+label6.Caption[1];
end;

end.

```

PASSWORD FORM:

```

unit password_form;

```

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, DB, ADODB, StdCtrls, Buttons, jpeg, ExtCtrls;

```

```

type

```

```

Tpassword = class(TForm)
Image1: TImage;
Label1: TLabel;
Label2: TLabel;
Edit1: TEdit;
Edit2: TEdit;
BitBtn2: TBitBtn;
BitBtn1: TBitBtn;
ADOQuery1: TADOQuery;
DataSource1: TDataSource;
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure FormCreate(Sender: TObject);

```

```

procedure FormShow(Sender: TObject);
procedure FormKeyPress(Sender: TObject; var Key: Char);

```

```

private

```



```

    { Private declarations }
public
    { Public declarations }
end;

var
    password: Tpassword;

implementation

uses ab, main_menu;

{$R *.dfm}

procedure Tpassword.BitBtn1Click(Sender: TObject);
begin
    with adoquery1 do
        begin
            close;
            sql.Clear;
            sql.Add('select * from users where User_Name=:p and Password=:p1');
            parameters.ParamByName('p').Value:=edit1.Text;
            parameters.ParamByName('p1').Value:=edit2.Text;
            execsql;
            open;
        end;

    if adoquery1.RecordCount<>0 then
        begin
            if datasource1.dataset.FieldValues['Customers']=true then
                begin
                    main.Customers1.Enabled:=true;
                end
            else if datasource1.dataset.FieldValues['Customers']=false then
                begin
                    main.Customers1.Enabled:=false;
                end;

            if datasource1.dataset.FieldValues['Suppliers']=true then
                begin
                    main.Suppliers1.Enabled:=true;
                end
            else if datasource1.dataset.FieldValues['Suppliers']=false then
                begin
                    main.Suppliers1.Enabled:=false;
                end;

            if datasource1.dataset.FieldValues['Stocks']=true then

```

```

begin
main.Stocks1.Enabled:=true;
end
else if datasource1.dataset.FieldValues['Stocks']=false then
begin
main.Stocks1.Enabled:=false;
end;

if datasource1.dataset.FieldValues['Accounting']=true then
begin
main.Accounting1.Enabled:=true;
end
else if datasource1.dataset.FieldValues['Accounting']=false then
begin
main.Accounting1.Enabled:=false;
end;

```

```

if datasource1.dataset.FieldValues['Reports']=true then
begin
main.Reports1.Enabled:=true;
end
else if datasource1.dataset.FieldValues['Reports']=false then
begin
main.Reports1.Enabled:=false;
end;

```

```

if datasource1.dataset.FieldValues['Settings']=true then
begin
main.Settings1.Enabled:=true;
end
else if datasource1.dataset.FieldValues['Settings']=false then
begin
main.Settings1.Enabled:=false;
end;

```

```

main.Show;

```

```

end
else
begin

```

```

showmessage('wrong username or password');

```

```

end;

```

```

end;

```

```

procedure Tpassword.BitBtn2Click(Sender: TObject);

```

```

begin
application.terminate;
end;

procedure Tpassword.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

procedure Tpassword.FormCreate(Sender: TObject);
var
region: HRgn;
begin
    { yuvarlak form olusturur }
    region:=CreateEllipticRgn(50,50,420,320);
    SetWindowRgn(handle, region, true);
end;

procedure Tpassword.FormShow(Sender: TObject);
begin
about.visible:=false;
end;

procedure Tpassword.FormKeyPress(Sender: TObject; var Key: Char);
begin
    if Key = #27 then halt;
end;

end.

```

MAIN MENU FORM:

unit main_menu;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ExtCtrls, ExtDlgs, Menus, StdCtrls, Buttons, jpeg;

type

Tmain = class(TForm)

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

BitBtn5: TBitBtn;

BitBtn4: TBitBtn;

MainMenu1: TMainMenu;

Customers1: TMenuItem;

Suppliers1: TMenuItem;

Stocks1: TMenuItem;

Accounting1: TMenuItem;

Sellingg1: TMenuItem;

Purchasing1: TMenuItem;

N1: TMenuItem;

CustomerReceivables1: TMenuItem;

SupplierPayables1: TMenuItem;

Reports1: TMenuItem;

CustomerReports1: TMenuItem;

SupplierReports1: TMenuItem;

StockReports1: TMenuItem;

AccountingReports1: TMenuItem;

Settings1: TMenuItem;

UserSettings1: TMenuItem;

N3: TMenuItem;

DefineCategory1: TMenuItem;

N4: TMenuItem;

ChangeBackground1: TMenuItem;

Exit1: TMenuItem;

OpenPictureDialog1: TOpenPictureDialog;

Timer1: TTimer;

```

Timer2: TTimer;
Exit2: TMenuItem;
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure FormShow(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Customers1Click(Sender: TObject);
procedure Suppliers1Click(Sender: TObject);
procedure Stocks1Click(Sender: TObject);
procedure DefineCategory1Click(Sender: TObject);
procedure Sellingg1Click(Sender: TObject);
procedure Purchasing1Click(Sender: TObject);
procedure CustomerReceivables1Click(Sender: TObject);
procedure SupplierPayables1Click(Sender: TObject);
procedure UserSettings1Click(Sender: TObject);
procedure ChangeBackground1Click(Sender: TObject);
procedure CustomerReports1Click(Sender: TObject);
procedure SupplierReports1Click(Sender: TObject);
procedure StockReports1Click(Sender: TObject);
procedure AccountingReports1Click(Sender: TObject);
procedure Exit1Click(Sender: TObject);
procedure Exit2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  main: Tmain;

implementation

uses suppliers_form, customers_form, categories_form, stocks_form,
  selling_form, purchasing_form, customer_receivables, suppliers_payables,
  users_form, customer_report, returned_goods_form, supplier_reports,
  stock_report, accounting_report, information;

{$R *.dfm}

procedure Tmain.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  application.Terminate;
end;

procedure Tmain.FormShow(Sender: TObject);
begin
  bitbtn1.Caption:='Shortcut to'+chr(13)+'selling';

```



```

bitbtn2.Caption:='Shortcut to'+chr(13)+'purchasing';
bitbtn3.Caption:='Shortcut to'+chr(13)+'customer'+chr(13)+ 'receivables';
bitbtn4.Caption:='Shortcut to'+chr(13)+'under minimum'+chr(13)+'stock
level'+chr(13)+'report';
bitbtn5.Caption:='Shortcut to'+chr(13)+'cash analyse'+chr(13)+'report';
end;

procedure Tmain.Timer2Timer(Sender: TObject);
begin
label2.Caption:=Timetostr(time);
label4.Caption:=Datetostr(Date);
end;
procedure Tmain.Timer1Timer(Sender: TObject);
begin
timer1.Interval:=100;
CAPTION:=COPY(CAPTION,2,LENGTH(CAPTION)-1)+CAPTION[1];
end;

procedure Tmain.Customers1Click(Sender: TObject);
begin
customers.showmodal;
end;

procedure Tmain.Suppliers1Click(Sender: TObject);
begin
suppliers.showmodal;
end;

procedure Tmain.Stocks1Click(Sender: TObject);
begin
stocks.showmodal;
end;

procedure Tmain.DefineCategory1Click(Sender: TObject);
begin
categories.ShowModal;
end;

procedure Tmain.Sellingg1Click(Sender: TObject);
begin
selling.showmodal;
end;

procedure Tmain.Purchasing1Click(Sender: TObject);
begin
purchasing.showmodal;
end;

procedure Tmain.CustomerReceivables1Click(Sender: TObject);

```



```
begin
Customer_receivable.showmodal;
end;
```

```
procedure Tmain.SupplierPayables1Click(Sender: TObject);
begin
supplier_payables.showmodal;
end;
```

```
procedure Tmain.UserSettings1Click(Sender: TObject);
begin
users.showmodal;
end;
```

```
procedure Tmain.ChangeBackground1Click(Sender: TObject);
begin
  if openpicturedialog1.Execute then
    try
      image1.Picture.LoadFromFile(openpicturedialog1.FileName);
    except
      showmessage('Picture file can not open') ;
    end;
end;
```

```
procedure Tmain.CustomerReports1Click(Sender: TObject);
begin
customer_rep.showmodal;
end;
```

```
procedure Tmain.SupplierReports1Click(Sender: TObject);
begin
supplier_rep.showmodal;
end;
```

```
procedure Tmain.StockReports1Click(Sender: TObject);
begin
stock_rep.showmodal;
end;
```

```
procedure Tmain.AccountingReports1Click(Sender: TObject);
begin
accounting_rep.showmodal;
end;
```

```
procedure Tmain.Exit1Click(Sender: TObject);
begin
info.showmodal;
end;
```

```
procedure Tmain.Exit2Click(Sender: TObject);
```

```
begin
application.Terminate;
end;
end.
```

CUSTOMERS FORM:

```
unit customers_form;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ADODB, DB, StdCtrls, DBCtrls, Mask, Grids, DBGrids, Buttons,
ExtCtrls;
```

```
type
```

```
Tcustomers = class(TForm)
  GroupBox1: TGroupBox;
  Panel1: TPanel;
  Label1: TLabel;
  Label5: TLabel;
  ComboBox1: TComboBox;
  Edit1: TEdit;
  BitBtn1: TBitBtn;
  Panel2: TPanel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  Label8: TLabel;
  Label9: TLabel;
  Label10: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  DBGrid1: TDBGrid;
  BitBtn2: TBitBtn;
  BitBtn3: TBitBtn;
  BitBtn4: TBitBtn;
  BitBtn5: TBitBtn;
  BitBtn6: TBitBtn;
  BitBtn7: TBitBtn;
  BitBtn8: TBitBtn;
  BitBtn9: TBitBtn;
  DBEdit1: TDBEdit;
  DBEdit2: TDBEdit;
  DBEdit3: TDBEdit;
  DBEdit5: TDBEdit;
  DBEdit6: TDBEdit;
  DBEdit7: TDBEdit;
```

```

DBEdit8: TDBEdit;
DBMemo1: TDBMemo;
BitBtn11: TBitBtn;
BitBtn12: TBitBtn;
ADOTable1: TADOTable;
DataSource1: TDataSource;
ADOQuery1: TADOQuery;
DataSource2: TDataSource;
ADOQuery2: TADOQuery;
DataSource3: TDataSource;
ADOQuery3: TADOQuery;
DataSource4: TDataSource;
procedure FormShow(Sender: TObject);
procedure ComboBox1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn12Click(Sender: TObject);
procedure DBEdit1KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit2KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit3KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit7KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure DBEdit5KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit6KeyPress(Sender: TObject; var Key: Char);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  customers: Tcustomers;

implementation

uses main_menu;

{$R *.dfm}

procedure Tcustomers.FormShow(Sender: TObject);
begin
  label12.Caption:=floattostr(adotable1.RecordCount) ;

```

```

dbgrid1.DataSource:=datasource1;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbedit6.DataSource:=datasource1;
dbedit7.DataSource:=datasource1;
dbedit8.DataSource:=datasource1;
dbmemo1.DataSource:=datasource1;
edit1.Clear;
combobox1.Text:='choose';

```

```

bitbtn9.Caption:='Main'+chr(13)+'Menu';
end;

```

```

procedure Tcustomers.ComboBox1Click(Sender: TObject);
begin
edit1.Clear;
end;

```

```

procedure Tcustomers.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if combobox1.ItemIndex=0 then
begin
if not (key in ['0'..'9',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end
else if combobox1.ItemIndex=1 then
begin
if not (key in ['A'..'Z','a'..'z',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;

```

```

end
else if combobox1.ItemIndex=2 then
begin
if not (key in ['A'..'Z','a'..'z',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;

```

```

end
else if combobox1.ItemIndex=4 then
begin

```

```

    if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanıycıyı uyar.
end;

```

```

    end
end;
procedure Tcustomers.BitBtn1Click(Sender: TObject);
begin

```

```

if combobox1.Text='Customer Number' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from customers where CustomerNo=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Name' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from customers where Name=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Surname' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from customers where Surname=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
end

```



```

else

if combobox1.Text='Phone' then
    begin
with adoquery1 do
begin
    close;
    sql.Clear;
    sql.Add('select * from customers where phone=:x');
    parameters.ParamByName('x').Value:= edit1.Text;
    execsql;
    open;
end;
end
else
if combobox1.Text='City' then
    begin
with adoquery1 do
begin
    close;
    sql.Clear;
    sql.Add('select * from customers where City=:x');
    parameters.ParamByName('x').Value:= edit1.Text;
    execsql;
    open;
end;
end
else
if combobox1.Text='Country' then
    begin
with adoquery1 do
begin
    close;
    sql.Clear;
    sql.Add('select * from customers where Country=:x');
    parameters.ParamByName('x').Value:= edit1.Text;
    execsql;
    open;
end;
end ;
if adoquery1.RecordCount=0 then
    showmessage('Customer Record Not Found');
    dbedit1.DataSource:=datasource2;
dbedit2.DataSource:=datasource2;
dbedit3.DataSource:=datasource2;

dbedit5.DataSource:=datasource2;
dbedit6.DataSource:=datasource2;
dbedit7.DataSource:=datasource2;
dbedit8.DataSource:=datasource2;

```



```
dbmemo1.DataSource:=datasource2;  
dbgrid1.DataSource:=datasource2;
```

```
end;
```

```
procedure Tcustomers.BitBtn11Click(Sender: TObject);  
begin  
adotable1.First;  
end;
```

```
procedure Tcustomers.BitBtn2Click(Sender: TObject);  
begin  
adotable1.Prior;  
end;
```

```
procedure Tcustomers.BitBtn3Click(Sender: TObject);  
begin  
adotable1.Next;  
end;
```

```
procedure Tcustomers.BitBtn12Click(Sender: TObject);  
begin  
adotable1.Last;  
end;
```

```
procedure Tcustomers.DBEdit1KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['0'..'9',#8]) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür  
Beep; //bip sesi ile kullanycyyy uyar.  
end;  
end;
```

```
procedure Tcustomers.DBEdit2KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['A'..'Z','a'..'z',#8]) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür  
Beep; //bip sesi ile kullanycyyy uyar.  
end;  
end;
```

```
procedure Tcustomers.DBEdit3KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['A'..'Z','a'..'z',#8]) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
```

```
Beep; //bip sesi ile kullanıycyyy uyar.  
end;  
end;
```

```
procedure Tcustomers.DBEdit7KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['A'..'Z','a'..'z','#8']) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür  
Beep; //bip sesi ile kullanıycyyy uyar.  
end;  
end;
```

```
procedure Tcustomers.BitBtn4Click(Sender: TObject);  
begin  
begin  
dbedit1.Enabled:=true;  
bitbtn4.Enabled:=false;  
dbedit1.DataSource:=datasource1;  
dbedit2.DataSource:=datasource1;  
dbedit3.DataSource:=datasource1;  
  
dbedit5.DataSource:=datasource1;  
dbedit6.DataSource:=datasource1;  
dbedit7.DataSource:=datasource1;  
dbedit8.DataSource:=datasource1;  
dbmemo1.DataSource:=datasource1;  
dbgrid1.DataSource:=datasource1;  
adotable1.append;  
end;  
end;
```

```
procedure Tcustomers.BitBtn5Click(Sender: TObject);  
begin  
  
if dbedit1.Text="" then  
begin  
messagedlg('Please fill all required fields!',mtwarning,[mbok],0);  
end  
else  
begin  
with adoquery1 do  
begin  
close;  
sql.Clear;  
sql.Add('select CustomerNo from customers where CustomerNo=:c1');  
parameters.ParamByName('c1').Value:=dbedit1.Text;  
execsql;  
open;  
end;
```

```

if datasource2.DataSet.RecordCount<>0 then
begin
showmessage('this customer number in use');
end
else
begin
if ((dbedit1.Text<>")and (dbedit2.Text<>")and (dbedit3.Text<>")and
(dbedit5.Text<>")and (dbedit6.Text<>")and(dbmemo1.Text<>")and (dbedit7.Text<>")and
(dbedit8.Text<>")) then
begin
adotable1.open;
adotable1.Post;

showmessage ('Customer record added succesfully. ');
adotable1.Insert;

bitbtn4.Enabled:=true;
end
else
begin
messagedlg('Please fill all required fields!',mtwarning,[mbok],0);
end;
end;
dbedit1.Enabled:=false;

label12.Caption:=floattostr(adotable1.RecordCount) ;

end;

end;

```

```

procedure Tcustomers.BitBtn6Click(Sender: TObject);
begin
if ((dbedit1.Text=") or (dbedit2.Text=") or (dbedit3.Text=")or (dbedit5.Text=")or
(dbedit6.Text=")or(dbmemo1.Text=")or (dbedit7.Text=")or (dbedit8.Text=")) then
begin
messagedlg('There is no added record !',mtwarning,[mbok],0);
end
else
begin
adotable1.Edit;
adotable1.UpdateRecord;
showmessage ('customer record updated succesfully. ');
end;
end;

```

```

procedure Tcustomers.BitBtn7Click(Sender: TObject);

```

```

var
x:word;
label 5;
begin
try
if ((dbedit1.Text=")or (dbedit2.Text=")or (dbedit3.Text=")or (dbedit5.Text=")or
(dbedit6.Text=")or(dbmemo1.Text=")or (dbedit7.Text=")or (dbedit8.Text=")) then
begin
messagedlg('you can not delete on empty datasheet',mtwarning,[mbok],0);
end
else
begin
x:=messagedlg('Are you sure to delete this customer?',mtwarning,[mbytes,mbno],0);
if x=mryes then
begin

with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from Invoice where CustomerNo=:p');
parameters.ParamByName('p').Value:=dbedit1.Text;
execsql;
open;
end;

while not adoquery2.Eof do
begin
if datasource3.DataSet.FieldValues['kapalı']=false then
begin
messagedlg('you can not delete this customer',mtwarning,[mbok],0);
goto 5;

end;

end;

if adoquery2.RecordCount=0 then
begin
with adoquery3 do
begin
close;
sql.Clear;
sql.Add('delete * from customers where CustomerNo=:k');
parameters.ParamByName('k').Value:=dbedit1.Text;
execsql;

end;

showmessage('customer record deleted successfully.');
```

label12.Caption:=floattostr(adotable1.RecordCount) ;

```

        end;
    end;

5:
end;

except
on EDatabaseError do showmessage('you can not delete on empty datasheet');
end;

end;
procedure Tcustomers.BitBtn8Click(Sender: TObject);
begin
    adotable1.Cancel;
    bitbtn4.Enabled:=true;
    dbedit1.Enabled:=false;
    dbedit1.DataSource:=datasource1;
    dbedit2.DataSource:=datasource1;
    dbedit3.DataSource:=datasource1;
    dbedit5.DataSource:=datasource1;
    dbedit6.DataSource:=datasource1;
    dbedit7.DataSource:=datasource1;
    dbedit8.DataSource:=datasource1;
    dbmemo1.DataSource:=datasource1;
    dbgrid1.DataSource:=datasource1;
    edit1.Clear;
    combobox1.Text:='choose';
end;
procedure Tcustomers.BitBtn9Click(Sender: TObject);
begin
    main.show;
    customers.Close;
end;
procedure Tcustomers.DBEdit5KeyPress(Sender: TObject; var Key: Char);
begin
    if not (key in ['0'..'9',#8]) then
    begin
        Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
        Beep; //bip sesi ile kullanıycıyı uyar.
    end;
end;
procedure Tcustomers.DBEdit6KeyPress(Sender: TObject; var Key: Char);
begin
    if not (key in ['0'..'9',#8]) then
    begin
        Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
        Beep; //bip sesi ile kullanıycıyı uyar.
    end;
end;
end;

```


end.

SUPPLIERS FORM:

unit suppliers_form;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ADODB, DB, StdCtrls, DBCtrls, Mask, Grids, DBGrids, Buttons,
ExtCtrls;

type

Tsuppliers = class(TForm)
 GroupBox1: TGroupBox;
 Panel1: TPanel;
 Label1: TLabel;
 Label14: TLabel;
 ComboBox1: TComboBox;
 Edit1: TEdit;
 BitBtn1: TBitBtn;
 Panel2: TPanel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 Label7: TLabel;
 Label8: TLabel;
 Label9: TLabel;
 Label10: TLabel;
 Label11: TLabel;
 Label12: TLabel;
 Label13: TLabel;
 DBGrid1: TDBGrid;
 DBEdit1: TDBEdit;
 DBEdit2: TDBEdit;
 DBEdit3: TDBEdit;
 DBEdit4: TDBEdit;
 DBEdit5: TDBEdit;
 DBEdit6: TDBEdit;
 DBEdit7: TDBEdit;
 DBEdit8: TDBEdit;
 DBMemo1: TDBMemo;
 BitBtn2: TBitBtn;
 BitBtn3: TBitBtn;
 BitBtn4: TBitBtn;


```

BitBtn5: TBitBtn;
BitBtn6: TBitBtn;
BitBtn7: TBitBtn;
BitBtn8: TBitBtn;
BitBtn9: TBitBtn;
BitBtn10: TBitBtn;
BitBtn11: TBitBtn;
ADOTable1: TADOTable;
DataSource1: TDataSource;
ADOQuery1: TADOQuery;
DataSource2: TDataSource;
ADOQuery2: TADOQuery;
DataSource3: TDataSource;
ADOQuery3: TADOQuery;
DataSource4: TDataSource;
procedure FormShow(Sender: TObject);
procedure ComboBox1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure DBEdit2KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit3KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit7KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure DBEdit5KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit6KeyPress(Sender: TObject; var Key: Char);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  suppliers: Tsuppliers;

implementation

uses main_menu;

{$R *.dfm}

procedure Tsuppliers.FormShow(Sender: TObject);

```

```

begin
label12.Caption:=floattostr(adotable1.RecordCount) ;
dbgrid1.DataSource:=datasource1;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit4.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbedit6.DataSource:=datasource1;
dbedit7.DataSource:=datasource1;
dbedit8.DataSource:=datasource1;
dbmemo1.DataSource:=datasource1;
edit1.Clear;
combobox1.Text:='choose';
bitbtn9.Caption:='Main'+chr(13)+'menu' ;

end;

procedure Tsuppliers.ComboBox1Click(Sender: TObject);
begin
edit1.Clear;
end;

procedure Tsuppliers.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  if combobox1.ItemIndex=0 then
    begin
      if not (key in ['0'..'9',#8]) then
        begin
          Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
          Beep; //bip sesi ile kullanycyyy uyar.
        end;
      end
    else if combobox1.ItemIndex=1 then
      begin
        if not (key in ['A'..'Z','a'..'z',#8]) then
          begin
            Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
            Beep; //bip sesi ile kullanycyyy uyar.
          end;
        end
      else if combobox1.ItemIndex=2 then
        begin
          if not (key in ['A'..'Z','a'..'z',#8]) then
            begin
              Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
              Beep; //bip sesi ile kullanycyyy uyar.
            end;
          end
        end
      end
end;

```

```

end
else if combobox1.ItemIndex=5 then
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanyyyy uyar.
end;

end
end;

```

```

procedure Tsuppliers.BitBtn1Click(Sender: TObject);
begin
try
bitbtn4.Enabled:=true;
adotable1.Cancel;
dbedit1.DataSource:=datasource2;
dbedit2.DataSource:=datasource2;
dbedit3.DataSource:=datasource2;
dbedit5.DataSource:=datasource2;
dbedit4.DataSource:=datasource2;
dbedit6.DataSource:=datasource2;
dbedit7.DataSource:=datasource2;
dbedit8.DataSource:=datasource2;
dbmemo1.DataSource:=datasource2;
dbgrid1.DataSource:=datasource2;
if combobox1.Text='SupplierID' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers where SupplierID=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Name' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers where Name=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;

```

```

end;
end
else
    if combobox1.Text='Surname' then
        begin
            with adoquery1 do
            begin
                close;
                sql.Clear;
                sql.Add('select * from suppliers where Surname=:x');
                parameters.ParamByName('x').Value:= edit1.Text;
                execsql;
                open;
            end;
        end
    else
        if combobox1.Text='Company Name' then
            begin
                with adoquery1 do
                begin
                    close;
                    sql.Clear;
                    sql.Add('select * from suppliers where CompanyName=:x');
                    parameters.ParamByName('x').Value:= edit1.Text;
                    execsql;
                    open;
                end;
            end
        else
            if combobox1.Text='Phone' then
                begin
                    with adoquery1 do
                    begin
                        close;
                        sql.Clear;
                        sql.Add('select * from suppliers where phone=:x');
                        parameters.ParamByName('x').Value:= edit1.Text;
                        execsql;
                        open;
                    end;
                end
            else
                if combobox1.Text='City' then
                    begin
                        with adoquery1 do
                        begin
                            close;
                            sql.Clear;
                            sql.Add('select * from suppliers where City=:x');
                            parameters.ParamByName('x').Value:= edit1.Text;

```

```

execsql;
open;
end;
end
else
if combobox1.Text='Country' then
    begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers where Country=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end ;
if adoquery1.RecordCount=0 then
showmessage('Supplier Record Not Found');
except
on EDatabaseError do messagedlg('You must enter data first !',mtwarning,[mbok],0);
end;
end;

```

```

procedure Tsuppliers.BitBtn10Click(Sender: TObject);
begin
adotable1.First;
end;

```

```

procedure Tsuppliers.BitBtn2Click(Sender: TObject);
begin
adotable1.prior;
end;

```

```

procedure Tsuppliers.BitBtn3Click(Sender: TObject);
begin
adotable1.next;
end;

```

```

procedure Tsuppliers.BitBtn11Click(Sender: TObject);
begin
adotable1.last;
end;

```

```

procedure Tsuppliers.DBEdit2KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyy uyar.

```



```
end;  
end;
```

```
procedure Tsuppliers.DBEdit3KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['A'..'Z','a'..'z','#8']) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür  
Beep; //bip sesi ile kullanıycyyy uyar.  
end;  
end;
```

```
procedure Tsuppliers.DBEdit7KeyPress(Sender: TObject; var Key: Char);  
begin  
if not (key in ['A'..'Z','a'..'z','#8']) then  
begin  
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür  
Beep; //bip sesi ile kullanıycyyy uyar.  
end;  
end;
```

```
procedure Tsuppliers.BitBtn4Click(Sender: TObject);  
var  
yenino,s:integer;  
begin  
bitbtn4.Enabled:=false;  
dbedit1.DataSource:=datasource1;  
dbedit2.DataSource:=datasource1;  
dbedit3.DataSource:=datasource1;  
dbedit4.DataSource:=datasource1;  
dbedit5.DataSource:=datasource1;  
dbedit6.DataSource:=datasource1;  
dbedit7.DataSource:=datasource1;  
dbedit8.DataSource:=datasource1;  
dbmemo1.DataSource:=datasource1;  
dbgrid1.DataSource:=datasource1;  
s:=DBEdit1.Field.AsInteger;//Access 'den alınan Otomatik no..DBEdit'e alıyoruz.  
yenino :=s;  
if not ADOtable1.eof Then  
Begin  
ADOtable1.Last;//Tablonun son kayıtına yönleneiyor.  
yenino := ADOtable1['SupplierID'];//Access'in Otomatik no su  
end;  
yenino := yenino + 1;//Otomatik No'nun Bir sonraki rakamı.  
Label13.Caption := FloatToStr(yenino);//Label'e alınıyor.  
begin  
ADOtable1.Append;  
ADOtable1.Edit;  
DBEdit1.Field.Asstring:=Label13.Caption;//Labelden alınıp Tekrar yeni no veriliyor  
end;
```


end;

procedure Tsuppliers.BitBtn5Click(Sender: TObject);

begin

try

if (((dbedit1.Text<>")and(dbedit2.Text<>")and (dbedit3.Text<>")and
(dbedit5.Text<>")and (dbedit6.Text<>")and(dbmemo1.Text<>")and (dbedit7.Text<>")and
(dbedit8.Text<>")) then

begin

adotable1.Post;

showmessage ('Supplier record added succesfully.');

adotable1.Insert;

bitbtn4.Enabled:=true;

end

else

begin

messedlg('Please fill all required fields!',mtwarning,[mbok],0);

end;

label12.Caption:=floattostr(adotable1.RecordCount) ;

except

on EDatabaseError do showmessage('You should add empty space before add any record.');

end;

end;

procedure Tsuppliers.BitBtn6Click(Sender: TObject);

begin

if (((dbedit1.Text=") or (dbedit2.Text=") or (dbedit3.Text=")or (dbedit5.Text=")or
(dbedit6.Text=")or(dbmemo1.Text=")or (dbedit7.Text=")or (dbedit8.Text=")) then

begin

messedlg('There is no added record !',mtwarning,[mbok],0);

end

else

begin

adotable1.Edit;

adotable1.UpdateRecord;

showmessage ('Supplier record updated succesfully.');

end;

end;

procedure Tsuppliers.BitBtn7Click(Sender: TObject);

var

x:word;

label 5;

begin

try

if (((dbedit1.Text=")or (dbedit2.Text=")or (dbedit3.Text=")or (dbedit5.Text=")or
(dbedit6.Text=")or(dbmemo1.Text=")or (dbedit7.Text=")or (dbedit8.Text=")) then

begin

messedlg('you can not delete on empty datasheet',mtwarning,[mbok],0);

```

end
else
begin
x:=messagedlg('Are you sure to delete this supplier?',mtwarning,[mbyes,mbno],0);
if x=mryes then
begin

    with adoquery2 do
    begin
    close;
    sql.Clear;
    sql.Add('select * from Bills where SupplierID=:p');
    parameters.ParamByName('p').Value:=dbedit1.Text;
    execsql;
    open;
    end;

        while not adoquery2.EOF do
        begin
        if datasource3.DataSet.FieldValues['kapali']=false then
        begin
        messagedlg('you can not delete this supplier',mtwarning,[mbok],0);
        goto 5;

        end;
        end;

        if adoquery2.RecordCount=0 then
        begin
        with adoquery3 do
        begin
        close;
        sql.Clear;
        sql.Add('delete * from suppliers where SupplierID=:k');
        parameters.ParamByName('k').Value:=dbedit1.Text;
        execsql;

        end;
        showmessage('supplier record deleted successfully. ');
        label12.Caption:=floattostr(adotable1.RecordCount) ;

        end;
        end;

5:
end;
except
on EDatabaseError do showmessage('you can not delete on empty datasheet');
end;

```

```

end;
procedure Tsuppliers.BitBtn8Click(Sender: TObject);
begin
bitbtn4.Enabled:=true;
adotable1.Cancel;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbedit4.DataSource:=datasource1;
dbedit6.DataSource:=datasource1;
dbedit7.DataSource:=datasource1;
dbedit8.DataSource:=datasource1;
dbmemo1.DataSource:=datasource1;
dbgrid1.DataSource:=datasource1;
edit1.Clear;
combobox1.Text:='choose';
end;

```

```

procedure Tsuppliers.BitBtn9Click(Sender: TObject);
begin
main.show;
suppliers.Close;
end;

```

```

procedure Tsuppliers.DBEdit5KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['0'..'9',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

```

```

procedure Tsuppliers.DBEdit6KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['0'..'9',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

end.

```

STOCKS FORM:

unit stocks_form;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ADODB, DB, DBCtrls, Mask, Grids, DBGrids, StdCtrls, Buttons,
ExtCtrls, ExtDlgs;

type

Tstocks = class(TForm)
 GroupBox1: TGroupBox;
 Panel1: TPanel;
 Label1: TLabel;
 Label17: TLabel;
 ComboBox1: TComboBox;
 Edit1: TEdit;
 BitBtn1: TBitBtn;
 Panel2: TPanel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 Label7: TLabel;
 Label8: TLabel;
 Label9: TLabel;
 Label10: TLabel;
 Label11: TLabel;
 Label12: TLabel;
 Label13: TLabel;
 Label14: TLabel;
 Label15: TLabel;
 Label16: TLabel;
 DBGrid1: TDBGrid;
 BitBtn2: TBitBtn;
 BitBtn3: TBitBtn;
 DBEdit1: TDBEdit;
 DBEdit2: TDBEdit;
 DBEdit3: TDBEdit;
 DBEdit4: TDBEdit;
 DBEdit5: TDBEdit;
 DBEdit6: TDBEdit;
 DBEdit7: TDBEdit;

```

BitBtn4: TBitBtn;
BitBtn5: TBitBtn;
BitBtn6: TBitBtn;
BitBtn7: TBitBtn;
BitBtn8: TBitBtn;
BitBtn9: TBitBtn;
BitBtn10: TBitBtn;
BitBtn11: TBitBtn;
DBEdit8: TDBEdit;
DBEdit9: TDBEdit;
ComboBox2: TComboBox;
BitBtn12: TBitBtn;
Edit2: TEdit;
ComboBox3: TComboBox;
DBEdit10: TDBEdit;
ADOTable1: TADOTable;
DataSource1: TDataSource;
ADOQuery1: TADOQuery;
DataSource2: TDataSource;
ADOQuery2: TADOQuery;
DataSource3: TDataSource;
ADOQuery3: TADOQuery;
DataSource4: TDataSource;
ADOQuery4: TADOQuery;
DataSource5: TDataSource;
ADOQuery5: TADOQuery;
DataSource6: TDataSource;
BitBtn13: TBitBtn;
Image1: TImage;
OpenPictureDialog1: TOpenPictureDialog;
ADOQuery6: TADOQuery;
DataSource7: TDataSource;
procedure FormShow(Sender: TObject);
procedure ComboBox1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure BitBtn12Click(Sender: TObject);
procedure BitBtn13Click(Sender: TObject);
procedure DataSource1DataChange(Sender: TObject; Field: TField);
procedure ComboBox2Change(Sender: TObject);

```



```

    procedure ComboBox3Change(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    stocks: Tstocks;

implementation

uses main_menu, categories_form;

{$R *.dfm}

procedure Tstocks.FormShow(Sender: TObject);

begin
    dbedit3.DataSource:=datasource1;
    dbedit10.DataSource:=datasource1;
    label10.Caption:=floattostr(adotable1.RecordCount) ;
    combobox2.Items.Clear;
    with adoquery2 do
    begin
        sql.Clear;
        sql.add('SELECT CategoryName FROM Categories');
        open;
    end;
    datasource3.DataSet.First;
    while not(datasource3.DataSet.Eof) do
    begin
        combobox2.Items.Add(datasource3.DataSet.FieldValues['CategoryName']);
        datasource3.DataSet.Next;
    end;

    combobox3.Items.Clear;
    with adoquery4 do
    begin
        sql.Clear;
        sql.add('SELECT CompanyName FROM suppliers');
        open;
    end;
    datasource5.DataSet.First;
    while not(datasource5.DataSet.Eof) do
    begin
        combobox3.Items.Add(datasource5.DataSet.FieldValues['CompanyName']);
        datasource5.DataSet.Next;
    end;
end;

```



```

end;
label10.Caption:=floattostr(adotable1.RecordCount) ;
dbgrid1.DataSource:=datasource1;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbedit6.DataSource:=datasource1;
dbedit7.DataSource:=datasource1;
dbedit8.DataSource:=datasource1;
dbedit9.DataSource:=datasource1;
dbedit10.DataSource:=datasource1;

edit1.Clear;
bitbtn12.Caption:='Add new'+chr(13)+'category' ;
bitbtn7.Caption:='Main'+chr(13)+'menu' ;
bitbtn13.Caption:='Insert'+chr(13)+'picture' ;

end;

procedure Tstocks.ComboBox1Click(Sender: TObject);
begin
edit1.Clear;
end;

procedure Tstocks.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if combobox1.ItemIndex=0 then
begin
if not (key in ['0'..'9',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end
else if combobox1.ItemIndex=2 then
begin
if not (key in ['A'..'Z','a'..'z',#8]) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;
end;

procedure Tstocks.BitBtn1Click(Sender: TObject);
begin
try

```

```

adotable1.Cancel;
dbedit1.DataSource:=datasource2;
dbedit2.DataSource:=datasource2;
dbedit3.DataSource:=datasource2;
dbedit4.DataSource:=datasource2;
dbedit5.DataSource:=datasource2;
dbedit6.DataSource:=datasource2;
dbedit7.DataSource:=datasource2;
dbgrid1.DataSource:=datasource2;

if combobox1.Text='Product ID' then
begin

with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from stocks where ProductID=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;

end
else
if combobox1.Text='Product Name' then
begin

with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from stocks where ProductName=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Category' then
begin

with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from stocks where CategoryName=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;

```

```

end;
end ;
if adoquery1.RecordCount=0 then
showmessage('Stock Record Not Found');

except
on EDatabaseError do messagedlg('You must enter data first !',mtwarning,[mbok],0);
end;
end;

```

```

procedure Tstocks.BitBtn10Click(Sender: TObject);
var
yenino,s:integer;
begin
bitbtn10.Enabled:=false;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit4.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbedit6.DataSource:=datasource1;
dbedit7.DataSource:=datasource1;

dbgrid1.DataSource:=datasource1;
s:=DBEdit1.Field.AsInteger;//Access 'den alınan Otomatik no..DBEdit'e alıyoruz.
yenino :=s;
if not ADOtable1.eof Then
Begin
ADOtable1.Last;//Tablonun son kayıtına yöneleniyor.
yenino := ADOtable1['ProductID'];//Access'in Otomatik no su
end;
yenino := yenino + 1;//Otomatik No'nun Bir sonraki rakamı.
Label11.Caption := FloatToStr(yenino);//Label'e alınıyor.
begin
ADOtable1.Append;
ADOtable1.Edit;
DBEdit1.Field.Asstring:=Label11.Caption;//Labelden alınıp Tekrar yeni no veriliyor
end;
end;

```

```

procedure Tstocks.BitBtn11Click(Sender: TObject);
begin
if (((dbedit1.Text<>"")and (dbedit10.Text<>"")and (dbedit2.Text<>"")and
(combobox2.Text<>"") and (dbedit3.Text<>"")and (dbedit5.Text<>"")and (dbedit6.Text<>"")and
(dbedit7.Text<>"") and (combobox3.Text<>"")) then

```

```

begin
adotable1.Open;
datasource1.DataSet.FieldValues['CategoryName']:=combobox2.Text;
datasource1.DataSet.FieldValues['CategoryID']:=dbedit3.Text;
datasource1.DataSet.FieldValues['SupplierID']:=dbedit10.Text;

adotable1.FieldName('Picture').AsString:=openpicturedialog1.FileName;
adotable1.Post;

    showmessage ('stock record added succesfully. ');
    label10.Caption:=floattostr(adotable1.RecordCount) ;
    bitbtn10.Enabled:=true;
end
else
begin
    messagedlg('Please fill all required fields!',mtwarning,[mbok],0);
end;

    dbedit3.DataSource:=datasource1;

end;

procedure Tstocks.BitBtn4Click(Sender: TObject);
begin
    if ((dbedit1.Text="")and (dbedit10.Text="")and (dbedit2.Text="")and (combobox2.Text="")
and (dbedit3.Text="")and (dbedit5.Text="")and (dbedit6.Text="")and (dbedit7.Text="") and
(combobox3.Text="")) then
        begin
            messagedlg('There is no record',mtwarning,[mbok],0);
        end
    else
        begin
            adotable1.Edit;
            adotable1.UpdateRecord;
            showmessage ('stock record updated succesfully. ');
        end;
    end;
end;

procedure Tstocks.BitBtn6Click(Sender: TObject);
var
x:word;
begin
try
    with adoquery5 do
        begin
            close;
            sql.Clear;

```



```

        sql.Add('select * from stocks as s inner join Returned_goods as r on
r.ProductID=s.ProductID where r.ProductID=:p');
        parameters.ParamByName('p').Value:=dbedit1.Text;
        execsql;
        open;
        end;

        if adoquery5.RecordCount<>0 then
        begin
            messagedlg('you can not delete this stock',mtwarning,[mbok],0);
        end
        else
        begin
            if ((dbedit1.Text="")and (dbedit10.Text="")and (dbedit2.Text="")and (combobox2.Text=")
and(dbedit3.Text="")and (dbedit5.Text="")and (dbedit6.Text="")and (dbedit7.Text=") and
(combobox3.Text=")) then
            begin
                messagedlg('you can not delete on empty datasheet',mtwarning,[mbok],0);
            end
            else
            begin
                x:=messagedlg('Are you sure to delete this stock?',mtwarning,[mbyes,mbno],0);
                if x=mryes then
                begin
                    if datasource1.DataSet.FieldValues['Quantity']=0 then
                    begin
                        with adoquery6 do
                        begin
                            close;
                            sql.Clear;
                            sql.Add('delete * from stocks where ProductID=:m');
                            parameters.ParamByName('m').Value:=dbedit1.Text;
                            execsql;
                        end;
                        showmessage('stock record deleted successfully.');
```

end

```

                    else
                    begin
                        messagedlg('you can not delete this product',mtwarning,[mbok],0);
                    end;
                label10.Caption:=floattostr(adotable1.RecordCount) ;
            end;
        end;
        end;
        except
        on EDatabaseError do showmessage('you can not delete on empty datasheet');
        end;

```

end;

```
procedure Tstocks.BitBtn5Click(Sender: TObject);
begin
  adotable1.Cancel;
  dbedit1.DataSource:=datasource1;
  dbedit2.DataSource:=datasource1;
  dbedit3.DataSource:=datasource1;
  dbedit5.DataSource:=datasource1;
  dbedit4.DataSource:=datasource1;
  dbedit6.DataSource:=datasource1;
  dbedit7.DataSource:=datasource1;
  dbedit8.DataSource:=datasource1;
  dbedit9.DataSource:=datasource1;
  dbedit10.DataSource:=datasource1;
  bitbtn10.Enabled:=true;
  dbgrid1.DataSource:=datasource1;;
end;
```

end;

```
procedure Tstocks.BitBtn7Click(Sender: TObject);
begin
  main.show;
  close;
end;
```

```
procedure Tstocks.BitBtn8Click(Sender: TObject);
begin
  adotable1.First;
end;
```

```
procedure Tstocks.BitBtn2Click(Sender: TObject);
begin
  adotable1.prior;
end;
```

```
procedure Tstocks.BitBtn3Click(Sender: TObject);
begin
  adotable1.next;
end;
```

```
procedure Tstocks.BitBtn9Click(Sender: TObject);
begin
  adotable1.last;
end;
```

```
procedure Tstocks.BitBtn12Click(Sender: TObject);
begin
```



```

categories.showmodal;
end;

procedure Tstocks.BitBtn13Click(Sender: TObject);
begin
openpicturedialog1.Execute;
image1.Picture.LoadFromFile(openpicturedialog1.FileName);
end;

procedure Tstocks.DataSource1DataChange(Sender: TObject; Field: TField);
begin
image1.Picture.LoadFromFile(adotable1.FieldValues['Picture']);
end;

procedure Tstocks.ComboBox2Change(Sender: TObject);
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from categories where CategoryName=x');
parameters.ParamByName('x').Value:=combobox2.Text;
execsql;
open;

end;

dbedit3.DataSource:=datasource3;
end;
procedure Tstocks.ComboBox3Change(Sender: TObject);
begin
with adoquery4 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers where CompanyName=x');
parameters.ParamByName('x').Value:=combobox3.Text;
execsql;
open;

end;

dbedit10.DataSource:=datasource5;
end;
end.

```

SELLING FORM:

```
unit selling_form;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, DB, ADODB, ExtCtrls, Grids, DBGrids, StdCtrls, ComCtrls,  
Buttons, Mask, DBCtrls;
```

```
type
```

```
Tselling = class(TForm)  
  GroupBox1: TGroupBox;  
  Label4: TLabel;  
  Label5: TLabel;  
  Label6: TLabel;  
  Label7: TLabel;  
  Label8: TLabel;  
  Label9: TLabel;  
  Label10: TLabel;  
  Label11: TLabel;  
  Label15: TLabel;  
  Panel1: TPanel;  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  Label14: TLabel;  
  DBText1: TDBText;  
  DBEdit1: TDBEdit;  
  DBEdit2: TDBEdit;  
  BitBtn1: TBitBtn;  
  BitBtn2: TBitBtn;  
  DateTimePicker1: TDateTimePicker;  
  BitBtn4: TBitBtn;  
  BitBtn5: TBitBtn;  
  Edit1: TEdit;  
  Edit2: TEdit;  
  DBEdit4: TDBEdit;  
  DBGrid1: TDBGrid;  
  RadioGroup1: TRadioGroup;  
  BitBtn7: TBitBtn;  
  Panel2: TPanel;  
  Label12: TLabel;  
  Label13: TLabel;  
  Label18: TLabel;
```

```

Label17: TLabel;
Edit3: TEdit;
Edit4: TEdit;
Edit6: TEdit;
Edit5: TEdit;
BitBtn6: TBitBtn;
BitBtn8: TBitBtn;
ADOTable1: TADOTable;
DataSource1: TDataSource;
ADOTable2: TADOTable;
DataSource2: TDataSource;
ADOTable3: TADOTable;
DataSource3: TDataSource;
ADOQuery1: TADOQuery;
DataSource4: TDataSource;
ADOTable4: TADOTable;
DataSource5: TDataSource;
ADOQuery2: TADOQuery;
DataSource6: TDataSource;
ADOTable5: TADOTable;
DataSource7: TDataSource;
ADOQuery3: TADOQuery;
ADOQuery3Expr1000: TBCTField;
ADOQuery3Expr1001: TFloatField;
DataSource8: TDataSource;
ADOQuery4: TADOQuery;
DataSource9: TDataSource;
ADOQuery5: TADOQuery;
DataSource10: TDataSource;
ADOQuery6: TADOQuery;
DataSource11: TDataSource;
Edit7: TEdit;
BitBtn3: TBitBtn;
procedure FormShow(Sender: TObject);
procedure DBEdit1KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit2KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure RadioGroup1Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure Edit4KeyPress(Sender: TObject; var Key: Char);
procedure Edit3KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn8Click(Sender: TObject);
procedure Edit2KeyPress(Sender: TObject; var Key: Char);
private
  { Private declarations }
public
  { Public declarations }

```

```

end;

var
    selling: Tselling;

implementation

uses customer_list, stock_list, fatura_report, main_menu,
    customer_Res_first_fatura;

{$R *.dfm}

procedure Tselling.FormShow(Sender: TObject);
begin
    dbtext1.Visible:=false;
    dbedit1.Clear;
    dbedit2.Clear;
    edit2.Clear;
    edit3.Clear;
    edit4.Clear;
    edit5.Clear;
    edit6.Clear;
    label6.Visible:=false;
    label8.Visible:=false;
    label10.Visible:=false;
    radiogroup1.ItemIndex:=-1;
    dbgrid1.DataSource:=datasource1;
    bitbtn4.visible:=false;
    bitbtn5.visible:=false;
    bitbtn7.visible:=false;
    bitbtn6.visible:=false;

    bitbtn4.Caption:='Add to'+chr(13)+'sales';
    bitbtn7.Caption:='Calculate'+chr(13)+'total';
    bitbtn8.Caption:='Main'+chr(13)+'menu';

end;

procedure Tselling.DBEdit1KeyPress(Sender: TObject; var Key: Char);
begin
    if ord(key)=13 then
    begin
        with adoquery4 do
        begin
            close;

```

```

yenino := yenino + 1; //Otomatik No'nun Bir sonraki rakamı.
Label11.Caption := FloatToStr(yenino); //Label'e alınıyor.
begin
    //ADOTable4.Append;
    //adotable4.edit;
    //dbEdit4.Field.Asstring:=Label11.Caption; //Labelden alınıp Tekrar yeni no veriliyor
    dbedit4.Text:=label11.Caption;

```

```

end;

```

```

dbgrid1.DataSource:=datasource5;

```

```

dbedit1.DataSource:=custtomer_list.DataSource1;
custtomer_list.showmodal;
end;

```

```

procedure Tselling.BitBtn2Click(Sender: TObject);
begin

```

```

    label6.Visible:=false;
    label8.Visible:=false;
    label10.Visible:=false;

```

```

    edit2.Clear;
    dbedit2.DataSource:=stocck_list.DataSource1;
    stocck_list.showmodal;
end;

```

```

procedure Tselling.BitBtn4Click(Sender: TObject);
begin

```

```

    bitbtn7.visible:=true;
    adotable4.Open;
    adotable4.insert;
    datasource5.DataSet.FieldValues['CustomerNo']:=dbedit1.Text;
    datasource5.DataSet.FieldValues['ProductID']:=dbedit2.Text;
    datasource5.DataSet.FieldValues['Quantity']:=edit2.Text;
    datasource5.DataSet.FieldValues['InvoiceNo']:=Label11.Caption;

```

```

    datasource5.DataSet.FieldValues['ProductName']:=stocck_list.DataSource1.DataSet.FieldValues['ProductName'];

```

```

    datasource5.DataSet.FieldValues['Brand']:=stocck_list.DataSource1.DataSet.FieldValues['Brand'];

```

```

    datasource5.DataSet.FieldValues['UnitPrice']:=stocck_list.DataSource1.DataSet.FieldValues['UnitPrice'];

```

```

    datasource5.DataSet.FieldValues['Date']:=datetostr(datetimepicker1.Date);
    datasource5.DataSet.FieldValues['Total']:=label10.Caption;

```



```

adotable4.Post;

bitbtn4.Visible:=false;

end;

procedure Tselling.BitBtn7Click(Sender: TObject);
var
  c,f:integer;
begin
    with adoquery3 do
    begin
      close;
      sql.Clear;
      sql.Add('select sum(total),sum(vat) from stock_add where InvoiceNo=:p');
      parameters.ParamByName('p').Value:=label11.Caption;
      execsql;
      open;

      end;

      c:=adotable4.RecordCount;
      label15.Caption:=inttostr(c);
      for f:=0 to c do
      begin
        label6.Caption:=datasource8.DataSet.FieldValues['Expr1000'];
        label8.Caption:=datasource8.DataSet.FieldValues['expr1001'];
        label10.Caption:=floattostr(strtfloat(label6.Caption)+strtfloat(label8.caption));

        end;
      end;

procedure Tselling.RadioGroup1Click(Sender: TObject);
begin
  if radiogroup1.ItemIndex=1 then
  begin
    panel2.enabled:=true;

  end
  else
  begin
    panel2.enabled:=false;
  end;
end;

```



```

end;
bitbtn6.Visible:=true;
end;

```

```

procedure Tselling.BitBtn6Click(Sender: TObject);

```

```

var
x,y:integer;
a,b:integer;
m:word;
begin

```

```

    m:=messagedlg('Are you sure to save selling
information?',mtconfirmation,[mbytes,mbno],0);
    if m=mryes then
    begin

```

```

        y:=adotable4.RecordCount;
        edit1.Text:=datasource4.DataSet.FieldValues['Quantity']-strtoint(edit2.Text);
    try
        if radiogroup1.ItemIndex=0 then
        begin
            adotable4.First;
            for x:=1 to y do
            begin

                adotable1.Open;
                adotable1.Insert;

```

```

datasource1.DataSet.FieldValues['CustomerNo']:=datasource5.DataSet.FieldValues['CustomerNo'];

```

```

datasource1.DataSet.FieldValues['ProductID']:=datasource5.DataSet.FieldValues['ProductId'];
;

```

```

datasource1.DataSet.FieldValues['Quantity']:=datasource5.DataSet.FieldValues['Quantity'];

```

```

datasource1.DataSet.FieldValues['InvoiceNo']:=datasource5.DataSet.FieldValues['InvoiceNo'];
];

```

```

datasource1.DataSet.FieldValues['ProductName']:=datasource5.DataSet.FieldValues['ProductName'];

```

```

datasource1.DataSet.FieldValues['Brand']:=datasource5.DataSet.FieldValues['Brand'];

```

```

datasource1.DataSet.FieldValues['UnitPrice']:=datasource5.DataSet.FieldValues['UnitPrice'];

```

```

datasource1.DataSet.FieldValues['Date']:=datasource5.DataSet.FieldValues['Date'];

```

```

datasource1.DataSet.FieldValues['Total']:=datasource5.DataSet.FieldValues['Total'];
    adotable1.Post;
    adotable4.Next;

    end;

    adotable5.Open;
    adotable5.Insert;

datasource7.DataSet.FieldValues['InvoiceNo']:=datasource5.DataSet.FieldValues['InvoiceNo'
];

datasource7.DataSet.FieldValues['CustomerNo']:=datasource5.DataSet.FieldValues['Custome
rNo'];
    datasource7.DataSet.FieldValues['Total']:=label10.Caption;

datasource7.DataSet.FieldValues['Date']:=datasource5.DataSet.FieldValues['Date'];
    datasource7.DataSet.FieldValues['taksit']:=false;
    datasource7.DataSet.FieldValues['kapalı']:=true;
    adotable5.Post;

    edit7.Text:=datasource5.DataSet.FieldValues['ProductID'];

with adoquery6 do
begin
    close;
    sql.Clear;
    sql.Add('update stocks set quantity=:k where ProductID=:k1');
    parameters.ParamByName('k').Value:=edit1.Text;
    parameters.ParamByName('k1').Value:=edit7.Text;
    execsql;
end;
else
begin
    adotable4.First;
    for x:=1 to y do
        begin
            adotable1.Open;
            adotable1.insert;

datasource1.DataSet.FieldValues['CustomerNo']:=datasource5.DataSet.FieldValues['Custome
rNo'];

datasource1.DataSet.FieldValues['ProductID']:=datasource5.DataSet.FieldValues['ProductId'
];

datasource1.DataSet.FieldValues['Quantity']:=datasource5.DataSet.FieldValues['Quantity'];

```

```

datasource1.DataSet.FieldValues['InvoiceNo']:=datasource5.DataSet.FieldValues['InvoiceNo'];

datasource1.DataSet.FieldValues['ProductName']:=datasource5.DataSet.FieldValues['Product
Name'];

datasource1.DataSet.FieldValues['Brand']:=datasource5.DataSet.FieldValues['Brand'];

datasource1.DataSet.FieldValues['UnitPrice']:=datasource5.DataSet.FieldValues['UnitPrice'];

datasource1.DataSet.FieldValues['Date']:=datasource5.DataSet.FieldValues['Date'];

datasource1.DataSet.FieldValues['Total']:=datasource5.DataSet.FieldValues['Total'];
    adotable1.Post;
    adotable4.Next;
end;

adotable2.Open;
adotable2.Insert;

datasource2.DataSet.FieldValues['InvoiceNo']:=datasource5.DataSet.FieldValues['InvoiceNo'];

datasource2.DataSet.FieldValues['CustomerNo']:=datasource5.DataSet.FieldValues['Custome
rNo'];
    datasource2.DataSet.FieldValues['Date']:=datasource5.DataSet.FieldValues['Date'];
    datasource2.DataSet.FieldValues['TotalPrice']:=label10.Caption;
    datasource2.DataSet.FieldValues['Number_Of_Installment']:=edit4.Text;
    datasource2.DataSet.FieldValues['Payment']:=edit3.Text;
    datasource2.DataSet.FieldValues['RemainingPayment']:= edit6.Text;
    datasource2.DataSet.FieldValues['InstallmentNo']:= 1;
    datasource2.DataSet.FieldValues['Required_Payment']:= edit5.Text;
    adotable2.post;

    edit7.Text:=datasource5.DataSet.FieldValues['ProductID'];
    with adoquery6 do
    begin
        close;
        sql.Clear;
        sql.Add('update stocks set quantity=:k where ProductID=:k1');
        parameters.ParamByName('k').Value:=edit1.Text;
        parameters.ParamByName('k1').Value:=edit7.Text;
        execsql;
    end;

adotable5.Open;
adotable5.Insert;

```

```
datasource7.DataSet.FieldValues['InvoiceNo']:=datasource5.DataSet.FieldValues['InvoiceNo'];
```

```
datasource7.DataSet.FieldValues['CustomerNo']:=datasource5.DataSet.FieldValues['CustomerNo'];
```

```
datasource7.DataSet.FieldValues['Total']:=label10.Caption;
```

```
datasource7.DataSet.FieldValues['Date']:=datasource5.DataSet.FieldValues['Date'];
```

```
datasource7.DataSet.FieldValues['taksit']:=true;
```

```
datasource7.DataSet.FieldValues['kapalı']:=false;
```

```
adotable5.Post;
```

```
end;
```

```
except
```

```
on EDatabaseError do showmessage('You must select the payment type');
```

```
end;
```

```
with adoquery2 do
```

```
begin
```

```
close;
```

```
sql.Clear;
```

```
sql.Add('delete * from stock_add');
```

```
execsql;
```

```
end;
```

```
dbedit1.Clear;
```

```
dbedit2.Clear;
```

```
edit2.Clear;
```

```
edit3.Clear;
```

```
edit4.Clear;
```

```
edit5.Clear;
```

```
edit6.Clear;
```

```
showmessage('selling information added to your system');
```

```
with adoquery5 do
```

```
begin
```

```
close;
```

```
sql.Clear;
```

```
sql.Add('select * from Invoice as i inner join s_stock as s on s.InvoiceNo=i.InvoiceNo
```

```
where s.InvoiceNo=:l');
```

```
parameters.ParamByName('l').Value:=dbedit4.Text;
```

```
execsql;
```

```
open;
```

```
end;
```

```
//fatura.ADOQuery1.Active:=true;
```



```

if radiogroup1.ItemIndex=0 then
begin
    fatura.QuickRep1.Preview;
end
else
begin
    fatura.QuickRep1.Preview;
    cust_rec_first_fatura.quickrep1.preview;
end;

end;

with adoquery2 do
begin
    close;
    sql.Clear;
    sql.Add('delete * from stock_add');
    execsql;

end;

end;

procedure Tselling.Edit4KeyPress(Sender: TObject; var Key: Char);
begin
    if Ord(Key)=13 then
        edit5.Text:=floattostr(strtfloat(label10.Caption) /strtfloat(edit4.Text)) ;
    end;

procedure Tselling.Edit3KeyPress(Sender: TObject; var Key: Char);
begin
    if Ord(Key)=13 then
        edit6.Text:=floattostr(strtfloat(label10.Caption)-strtfloat(edit3.Text));
    end;

procedure Tselling.BitBtn8Click(Sender: TObject);
begin
    with adoquery2 do
    begin
        close;
        sql.Clear;
        sql.Add('delete * from stock_add');
        execsql;

    end;

main.show;
close;
end;

```

```

procedure Tselling.Edit2KeyPress(Sender: TObject; var Key: Char);
var
  miktar:integer;
begin
  try

    dbtext1.Visible:=true;
    edit1.Text:=dbedit2.Text;
    if (dbedit1.Text='') or (dbedit2.Text='') then
      begin
        messagedlg('Please enter the values!',mtwarning,[mbok],0);
      end
    else
      begin
        with adoquery1 do
          begin
            close;
            sql.Clear;
            sql.Add('select * from stocks where ProductID=:p');
            parameters.ParamByName('p').Value:=edit1.Text;
            execsql;
            open;
          end;

          if Ord(Key)=13 then
            begin
              if strtoint(edit2.Text)<=0 then
                showmessage('you cant sell this quantity');
              if strtoint(edit2.Text)>stocck_list.DataSource1.DataSet.FieldValues['Quantity'] then
                begin
                  showmessage('you can not sell this quantity') ;
                  bitbtn4.visible:=false;
                  bitbtn5.visible:=false;
                  bitbtn6.visible:=false ;
                end
              else
                begin

                  miktar:=strtoint(dbtext1.Caption);

                  label6.Visible:=true;
                  label8.Visible:=true;
                  label10.Visible:=true;

                  bitbtn4.Visible:=true;
                  bitbtn4.visible:=true;
                  bitbtn5.visible:=true;

```



```
label6.Caption:=floattostr(stocck_list.DataSource1.DataSet.FieldValues['UnitPrice']*strtofloat(edit2.Text));
```

```
label8.Caption:=floattostr(strtofloat(label6.Caption)*(stocck_list.DataSource1.DataSet.FieldValues['vat']/100));
```

```
label10.Caption:=floattostr(strtofloat(label6.Caption)+strtofloat(label8.caption));
```

```
end;
```

```
end;
```

```
end;
```

```
except
```

```
on econverterror do
```

```
showmessage('you have to enter value');
```

```
end;
```

```
end;
```

```
end.
```

PURCHASING FORM:

```
unit purchasing_form;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, DB, ADODB, ExtCtrls, Grids, DBGrids, StdCtrls, ComCtrls,  
Buttons, Mask, DBCtrls;
```

```
type
```

```
Tpurchasing = class(TForm)  
  GroupBox1: TGroupBox;  
  Label4: TLabel;  
  Label5: TLabel;  
  Label6: TLabel;  
  Label7: TLabel;  
  Panel1: TPanel;  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  Label8: TLabel;  
  DBText1: TDBText;  
  DBEdit1: TDBEdit;  
  DBEdit2: TDBEdit;  
  BitBtn2: TBitBtn;  
  BitBtn3: TBitBtn;  
  DateTimePicker1: TDateTimePicker;  
  BitBtn4: TBitBtn;  
  BitBtn5: TBitBtn;  
  Edit1: TEdit;  
  Edit2: TEdit;  
  DBGrid1: TDBGrid;  
  DBEdit4: TDBEdit;  
  BitBtn6: TBitBtn;  
  BitBtn7: TBitBtn;  
  RadioGroup1: TRadioGroup;  
  Panel2: TPanel;  
  Label10: TLabel;  
  Label11: TLabel;  
  Label12: TLabel;  
  Label13: TLabel;  
  Edit3: TEdit;  
  Edit4: TEdit;  
  Edit5: TEdit;  
  Edit6: TEdit;  
  BitBtn8: TBitBtn;  
  ADOTable1: TADOTable;
```

```

DataSource1: TDataSource;
ADOTable2: TADOTable;
DataSource2: TDataSource;
ADOTable3: TADOTable;
DataSource3: TDataSource;
ADOQuery1: TADOQuery;
ADOQuery1Expr1000: TBCDField;
DataSource4: TDataSource;
ADOQuery2: TADOQuery;
DataSource5: TDataSource;
ADOTable4: TADOTable;
DataSource6: TDataSource;
ADOQuery3: TADOQuery;
DataSource7: TDataSource;
procedure FormShow(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn6Click(Sender: TObject);
procedure RadioGroup1Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure Edit3KeyPress(Sender: TObject; var Key: Char);
procedure Edit5KeyPress(Sender: TObject; var Key: Char);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  purchasing: Tpurchasing;

implementation

uses suppliers_list, stocks_form, main_menu, stock_list;

{$R *.dfm}

procedure Tpurchasing.FormShow(Sender: TObject);

var
  yenino,s:integer;
begin
  //datasource5.DataSet.Refresh;
  s:=DBEdit4.Field.AsInteger;//Access 'den alınan Otomatik no..DBEdit'e alıyoruz.
  yenino :=s;

```

```

dbedit4.DataSource:=datasource3;

if not ADOtable2.eof Then
  Begin
    ADOtable2.Last;//Tablonun son kaydına yöneliyor.
    yenino := ADOtable2['BillNo'];//Access'in Otomatik no su
    end;
    yenino := yenino + 1;//Otomatik No'nun Bir sonraki rakamı.
    Label5.Caption := FloatToStr(yenino);//Label'e alınıyor.
    begin
      //ADOtable4.Append;
      //adotable4.edit;
      //dbEdit4.Field.Asstring:=Label11.Caption;//Labelden alınıp Tekrar yeni no veriliyor
      dbedit4.Text:=label5.Caption;

    end;
    dbedit1.Clear;
    dbedit2.Clear;
    edit1.Clear;
    edit3.Clear;
    edit4.Clear;
    edit5.Clear;
    edit6.Clear;
    bitbtn4.visible:=false;
    bitbtn5.visible:=false;
    bitbtn6.visible:=false;
    bitbtn7.visible:=false;

    panel2.enabled:=false ;
    radiogroup1.ItemIndex:=-1;
    dbgrid1.DataSource:=datasource1;
    label7.Visible:=false;

    bitbtn3.Caption:='Add new'+chr(13)+'product';
    bitbtn4.Caption:='Add to'+chr(13)+'purchases';
    bitbtn6.Caption:='Calculate'+chr(13)+'total';
    bitbtn8.Caption:='Main'+chr(13)+'menu';

  end;

procedure Tpurchasing.BitBtn2Click(Sender: TObject);
begin
  dbgrid1.DataSource:=datasource3;

  label7.Visible:=false;
  edit1.Clear;
  dbedit2.DataSource:=stocck_list.DataSource1;
  dbedit1.DataSource:=stocck_list.DataSource1;

```

```

stocck_list.showmodal;
end;

procedure Tpurchasing.BitBtn3Click(Sender: TObject);
begin
stocks.showmodal;
end;

procedure Tpurchasing.BitBtn4Click(Sender: TObject);
begin
    adotable3.Open;
    adotable3.Insert;
    datasource3.DataSet.FieldValues['ProductId']:=dbedit2.Text;
    datasource3.DataSet.FieldValues['SupplierId']:=dbedit1.Text;
    datasource3.DataSet.FieldValues['BillNo']:=label5.Caption;

datasource3.DataSet.FieldValues['ProductName']:=stocck_list.DataSource1.DataSet.FieldValues['ProductName'];

datasource3.DataSet.FieldValues['Brand']:=stocck_list.DataSource1.DataSet.FieldValues['Brand'];

datasource3.DataSet.FieldValues['CategoryName']:=stocck_list.DataSource1.DataSet.FieldValues['CategoryName'];

datasource3.DataSet.FieldValues['UnitCost']:=stocck_list.DataSource1.DataSet.FieldValues['UnitCost'];

datasource3.DataSet.FieldValues['UnitPrice']:=stocck_list.DataSource1.DataSet.FieldValues['UnitPrice'];

datasource3.DataSet.FieldValues['CategoryName']:=stocck_list.DataSource1.DataSet.FieldValues['CategoryName'];
    datasource3.DataSet.FieldValues['Total']:=label7.Caption;
    datasource3.DataSet.FieldValues['Quantity']:=edit1.Text;
    datasource3.DataSet.FieldValues['Date']:=datetostr(datetimepicker1.Date);
    adotable3.Post;

    bitbtn6.visible:=true;
    bitbtn4.visible:=false;

end;
procedure Tpurchasing.Edit1KeyPress(Sender: TObject; var Key: Char);
var
miktar:integer;
begin
try

```



```
dbtext1.Visible:=true;
edit2.Text:=dbedit2.Text;
```

```
if (dbedit1.Text='') or (dbedit2.Text='') then
begin
    messagedlg('Please enter the values!',mtwarning,[mbok],0);
end
else
begin
    with adoquery2 do
    begin
        close;
        sql.Clear;
        sql.Add('select * from stocks where ProductID=:p');
        parameters.ParamByName('p').Value:=edit2.Text;
        execsql;
        open;
    end;
```

```
if Ord(Key)=13 then
    begin
        if strtoint(edit1.Text)<=0 then
        begin
            showmessage('you can not buy this quantity');
        end
        else
        begin
```

```
bitbtn4.visible:=true;
bitbtn5.visible:=false;
```

```
label7.Visible:=true;
miktar:=strtoint(dbtext1.Caption);
bitbtn4.Enabled:=true;
bitbtn5.Enabled:=true;
```

```
label7.Caption:=floattostr(stocck_list.DataSource1.DataSet.FieldValues['UnitCost']*strtofloat
(edit1.Text));
```

```
    end;
end;
end;
except
on econvertererror do
    showmessage('you have to enter value');
```

```

    end;
end;

procedure Tpurchasing.BitBtn6Click(Sender: TObject);
var
c,f:integer;
begin
    with adoquery1 do
        begin
            close;
            sql.Clear;
            sql.Add('select sum(total) from p_add where BillNo=:p');
            parameters.ParamByName('p').Value:=label5.Caption;
            execsql;
            open;

            end;

            c:=adotable3.RecordCount;
            label5.Caption:=inttostr(c);
            for f:=0 to c do
                begin
                    label7.Caption:=datasource4.DataSet.FieldValues['Expr1000'];

                end;
            end;
end;

```

```

procedure Tpurchasing.RadioGroup1Click(Sender: TObject);
begin
    if radiogroup1.ItemIndex=1 then
        begin
            panel2.enabled:=true;

            end
        else
            begin
                panel2.enabled:=false;

            end;
            bitbtn7.visible:=true;

end;

```

```

procedure Tpurchasing.BitBtn7Click(Sender: TObject);
var
x,y:integer;

```

```

s:=word;
begin

s:=messagedlg('Are you sure?',mtconfirmation,[mbytes,mbno],0);
if s=mryes then
begin
y:=adotable3.RecordCount;
edit2.Text:=datasource5.DataSet.FieldValues['Quantity']+strtoint(edit1.Text);
    try

        if radiogroup1.ItemIndex=0 then
        begin
            adotable3.First;
            for x:=1 to y do
            begin
                adotable1.Open;
                adotable1.Insert;

datasource1.DataSet.FieldValues['ProductId']:=datasource3.DataSet.FieldValues['ProductId'];

datasource1.DataSet.FieldValues['SupplierId']:=datasource3.DataSet.FieldValues['SupplierId'
];
            datasource1.DataSet.FieldValues['BillNo']:=datasource3.DataSet.FieldValues['BillNo'];

datasource1.DataSet.FieldValues['ProductName']:=datasource3.DataSet.FieldValues['Product
Name'];
            datasource1.DataSet.FieldValues['Brand']:=datasource3.DataSet.FieldValues['Brand'];

datasource1.DataSet.FieldValues['CategoryName']:=datasource3.DataSet.FieldValues['Categ
oryName'];

datasource1.DataSet.FieldValues['UnitCost']:=datasource3.DataSet.FieldValues['UnitCost'];

datasource1.DataSet.FieldValues['UnitPrice']:=datasource3.DataSet.FieldValues['UnitPrice'];

datasource1.DataSet.FieldValues['Quantity']:=datasource3.DataSet.FieldValues['Quantity'];
            datasource1.DataSet.FieldValues['Total']:=datasource3.DataSet.FieldValues['Total'];
            datasource1.DataSet.FieldValues['Date']:=datasource3.DataSet.FieldValues['Date'];

            adotable1.Post;
            adotable3.Next;
            end;

            adotable2.Open;
            adotable2.Insert;

            datasource2.DataSet.FieldValues['BillNo']:=datasource3.DataSet.FieldValues['BillNo'];

datasource2.DataSet.FieldValues['SupplierId']:=datasource3.DataSet.FieldValues['SupplierId'
];

```

```

datasource2.DataSet.FieldValues['Total']:=label7.Caption;
datasource2.DataSet.FieldValues['Date']:=datasource3.DataSet.FieldValues['Date'];
datasource2.DataSet.FieldValues['taksit']:=false;
datasource2.DataSet.FieldValues['kapalı']:=true;

```

```

adotable2.Post;

```

```

adoquery2.Edit;
datasource5.DataSet.FieldValues['Quantity']:=edit2.Text;
adoquery2.Post;
end
else
begin
    adotable3.First;
for x:=1 to y do
begin
    adotable1.Open;
    adotable1.Insert;

```

```

datasource1.DataSet.FieldValues['ProductId']:=datasource3.DataSet.FieldValues['ProductId'];

```

```

datasource1.DataSet.FieldValues['SupplierId']:=datasource3.DataSet.FieldValues['SupplierId'];

```

```

    datasource1.DataSet.FieldValues['BillNo']:=datasource3.DataSet.FieldValues['BillNo'];

```

```

datasource1.DataSet.FieldValues['ProductName']:=datasource3.DataSet.FieldValues['Product
Name'];

```

```

    datasource1.DataSet.FieldValues['Brand']:=datasource3.DataSet.FieldValues['Brand'];

```

```

datasource1.DataSet.FieldValues['CategoryName']:=datasource3.DataSet.FieldValues['Categ
oryName'];

```

```

datasource1.DataSet.FieldValues['UnitCost']:=datasource3.DataSet.FieldValues['UnitCost'];

```

```

datasource1.DataSet.FieldValues['UnitPrice']:=datasource3.DataSet.FieldValues['UnitPrice'];

```

```

datasource1.DataSet.FieldValues['Quantity']:=datasource3.DataSet.FieldValues['Quantity'];

```

```

    datasource1.DataSet.FieldValues['Total']:=datasource3.DataSet.FieldValues['Total'];

```

```

    datasource1.DataSet.FieldValues['Date']:=datasource3.DataSet.FieldValues['Date'];

```

```

adotable1.Post;
adotable3.Next;
end;

```

```

    adotable2.Open;
    adotable2.Insert;

```

```

datasource2.DataSet.FieldValues['BillNo']:=datasource3.DataSet.FieldValues['BillNo'];

```



```
datasource2.DataSet.FieldValues['SupplierId']:=datasource3.DataSet.FieldValues['SupplierId'];
```

```
datasource2.DataSet.FieldValues['Total']:=label7.Caption;
```

```
datasource2.DataSet.FieldValues['Date']:=datasource3.DataSet.FieldValues['Date'];
```

```
datasource2.DataSet.FieldValues['taksit']:=true;
```

```
datasource2.DataSet.FieldValues['kapalı']:=false;
```

```
adotable2.Post;
```

```
adoquery2.Edit;
```

```
datasource5.DataSet.FieldValues['Quantity']:=edit2.Text;
```

```
adoquery2.Post;
```

```
adotable4.Open;
```

```
adotable4.Insert;
```

```
datasource6.DataSet.FieldValues['BillNo']:=datasource3.DataSet.FieldValues['BillNo'];
```

```
datasource6.DataSet.FieldValues['SupplierID']:=datasource3.DataSet.FieldValues['SupplierID'];
```

```
datasource6.DataSet.FieldValues['Date']:=datasource3.DataSet.FieldValues['Date'];
```

```
datasource6.DataSet.FieldValues['Total']:=label7.Caption;
```

```
datasource6.DataSet.FieldValues['Num_of_installment']:=edit3.Text;
```

```
datasource6.DataSet.FieldValues['Payment']:=edit5.Text;
```

```
datasource6.DataSet.FieldValues['RemainingPayment']:= edit6.Text;
```

```
datasource6.DataSet.FieldValues['InstallmentNo']:= 1;
```

```
datasource6.DataSet.FieldValues['Required_payment']:= edit4.Text;
```

```
adotable4.Post;
```

```
end;
```

```
except
```

```
on EDatabaseError do showmessage('You must select the payment type');
```

```
end;
```

```
with adoquery2 do
```

```
begin
```

```
close;
```

```
sql.Clear;
```

```
sql.Add('delete * from p_add');
```

```
execsql;
```

```
end;
```



```

dbedit1.Clear;
dbedit2.Clear;
edit1.Clear;
  showmessage('purchasing information added to your system');
close;
end;

with adoquery2 do
begin
  close;
  sql.Clear;
  sql.Add('delete * from p_add');
  execsql;
end;

end;

```

```

procedure Tpurchasing.BitBtn8Click(Sender: TObject);
begin
  main.show;
close;
end;

```

```

procedure Tpurchasing.Edit3KeyPress(Sender: TObject; var Key: Char);
begin
  if Ord(Key)=13 then
    edit4.Text:=floattostr(strtfloat(label7.Caption) /strtfloat(edit3.Text)) ;
end;

```

```

procedure Tpurchasing.Edit5KeyPress(Sender: TObject; var Key: Char);
begin
  if Ord(Key)=13 then
    edit6.Text:=floattostr(strtfloat(label7.Caption)-strtfloat(edit5.Text));
end;
end.

```

CUSTOMER RECEIVABLES FORM:

unit customer_receivables;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, DB, ADODB, Grids, DBGrids, DBCtrls, StdCtrls, Buttons, ExtCtrls;

type

TCustomer_receivable = class(TForm)

 GroupBox1: TGroupBox;

 Panel1: TPanel;

 Label1: TLabel;

 Label2: TLabel;

 ComboBox1: TComboBox;

 Edit1: TEdit;

 BitBtn1: TBitBtn;

 Panel2: TPanel;

 Label3: TLabel;

 Label4: TLabel;

 Label5: TLabel;

 DBText1: TDBText;

 DBText2: TDBText;

 Label6: TLabel;

 Label7: TLabel;

 GroupBox2: TGroupBox;

 DBGrid1: TDBGrid;

 GroupBox3: TGroupBox;

 DBGrid2: TDBGrid;

 Edit2: TEdit;

 BitBtn2: TBitBtn;

 BitBtn3: TBitBtn;

 BitBtn4: TBitBtn;

 ADOQuery1: TADOQuery;

 DataSource2: TDataSource;

 ADOQuery2: TADOQuery;

 DataSource3: TDataSource;

 ADOQuery3: TADOQuery;

 DataSource1: TDataSource;

 procedure ComboBox1Click(Sender: TObject);

 procedure Edit1KeyPress(Sender: TObject; var Key: Char);

 procedure BitBtn1Click(Sender: TObject);

 procedure DBGrid1CellClick(Column: TColumn);

 procedure BitBtn2Click(Sender: TObject);

 procedure BitBtn3Click(Sender: TObject);

 procedure BitBtn4Click(Sender: TObject);

 procedure FormClose(Sender: TObject; var Action: TCloseAction);

private

```

    { Private declarations }
public
    { Public declarations }
end;

var
    Customer_receivable: TCustomer_receivable;

implementation

uses customer_payments;

{$R *.dfm}

procedure TCustomer_receivable.ComboBox1Click(Sender: TObject);
begin
    edit1.Clear;
end;

procedure TCustomer_receivable.Edit1KeyPress(Sender: TObject;
    var Key: Char);
begin
    if combobox1.ItemIndex=0 then
        begin
            if not (key in ['0'..'9',#8]) then
                begin
                    Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
                    Beep; //bip sesi ile kullanycyyy uyar.
                end;
            end
        else if combobox1.ItemIndex=1 then
            begin
                if not (key in ['0'..'9',#8]) then
                    begin
                        Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
                        Beep; //bip sesi ile kullanycyyy uyar.
                    end;
                end
            else if combobox1.ItemIndex=2 then
                begin
                    if not (key in ['A'..'Z','a'..'z',#8]) then
                        begin
                            Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
                            Beep; //bip sesi ile kullanycyyy uyar.
                        end;
                    end
                else if combobox1.ItemIndex=3 then
                    begin

```

```

    if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanıycıyı uyar.
end;

    end
end;
procedure TCustomer_receiveable.BitBtn1Click(Sender: TObject);
begin
try
dbgrid1.DataSource:=datasource3;
if combobox1.Text='Customer Number' then
    begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from customer_rec where CustomerNo=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Invoice Number' then
    begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from customer_rec where InvoiceNo=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Name' then
    begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from customers as c inner join customer_rec as i on
c.CustomerNo=i.CustomerNo where c.Surname=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end;

```

```

end
else

if combobox1.Text='Surname' then
    begin
    with adoquery2 do
    begin
        close;
        sql.Clear;
        sql.Add('select * from customers as c inner join customer_rec as i on
c.CustomerNo=i.CustomerNo where c.Surname=:x');
        parameters.ParamByName('x').Value:= edit1.Text;
        execsql;
        open;

    end;

    end ;
    if adoquery2.RecordCount=0 then
        messagedlg('Record not found',mtwarning,[mbok],0);

    except
    on edatabaseerror do messagedlg('Record not found',mtwarning,[mbok],0);
    end;
    end;

procedure TCustomer_receivable.DBGrid1 CellClick(Column: TColumn);
begin
edit2.Text:=datasource3.DataSet.FieldValues['InvoiceNo'];
with adoquery1 do
begin
    close;
    sql.Clear;
    sql.Add('select * from Installment where InvoiceNo=:p');
    parameters.ParamByName('p').Value:=edit2.text;
    execsql;
    open;
end;

    with adoquery3 do
    begin
        close;
        sql.Clear;
        sql.Add('select sum(Payment) from Installment where InvoiceNo=:x');
        parameters.ParamByName('x').Value:=edit2.Text;
        execsql;
        open;
    end;

```



```
label6.Caption:=floattostr(strtfloat(dbtext1.Caption)-strtfloat(dbtext2.Caption));
end;
```

```
procedure TCustomer_receivable.BitBtn2Click(Sender: TObject);
begin
label7.Caption:=datasource3.DataSet.FieldValues['CustomerNo'];
```

```
with custoemr_payment.adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from customers where CustomerNo=:p');
parameters.ParamByName('p').Value:=label7.Caption;
execsql;
open;
end;
```

```
with custoemr_payment.adoquery2 do
begin
close;
sql.Clear;
sql.Add('select max(InstallmentNo),min(RemainingPayment) from Installment where
InvoiceNo=:p');
parameters.ParamByName('p').Value:=custoemr_payment.dbtext4.Caption;
execsql;
open;
end;
```

```
if custoemr_payment.DataSource2.DataSet.FieldValues['Expr1001']=0 then
begin
messagedlg('This account was closed!',mtwarning,[mbok],0);
end
else
begin
custoemr_payment.show;
end;
end;
```

```
procedure TCustomer_receivable.BitBtn3Click(Sender: TObject);
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from customer_rec');
```

```
execsql;  
open;  
end;  
dbgrid1.DataSource:= datasource3;  
end;
```

```
procedure TCustomer_receivable.BitBtn4Click(Sender: TObject);  
begin  
close;  
end;
```

```
procedure TCustomer_receivable.FormClose(Sender: TObject;  
var Action: TCloseAction);  
begin  
adoquery1.Close;  
adoquery2.Close;  
adoquery3.Close;  
custoemr_payment.ADOQuery1.Close;  
custoemr_payment.ADOQuery2.Close;  
custoemr_payment.ADOQuery3.Close;  
custoemr_payment.ADOQuery4.Close;
```

```
end;
```

```
end.
```

CUSTOMER PAYMENT FORM:

```
unit customer_payments;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, ADODB, DB, StdCtrls, Buttons, ComCtrls, DBCtrls, ExtCtrls;
```

```
type
```

```
Tcustoemr_payment = class(TForm)  
  GroupBox1: TGroupBox;  
  Panel1: TPanel;  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  Label4: TLabel;  
  Label5: TLabel;  
  Label6: TLabel;  
  DBText1: TDBText;  
  DBText2: TDBText;  
  DBText3: TDBText;  
  DBText4: TDBText;  
  DBText5: TDBText;  
  Label12: TLabel;  
  Panel2: TPanel;  
  Label7: TLabel;  
  Label8: TLabel;  
  Label9: TLabel;  
  Label10: TLabel;  
  Label11: TLabel;  
  DBText6: TDBText;  
  DBText7: TDBText;  
  DBText8: TDBText;  
  Label13: TLabel;  
  Label14: TLabel;  
  DateTimePicker1: TDateTimePicker;  
  Edit1: TEdit;  
  Edit2: TEdit;  
  BitBtn1: TBitBtn;  
  BitBtn2: TBitBtn;  
  BitBtn3: TBitBtn;  
  ADOQuery1: TADOQuery;  
  DataSource1: TDataSource;  
  ADOQuery2: TADOQuery;  
  DataSource2: TDataSource;  
  ADOQuery3: TADOQuery;  
  DataSource3: TDataSource;  
  ADOTable1: TADOTable;
```

```

DataSource4: TDataSource;
ADOQuery4: TADOQuery;
DataSource5: TDataSource;
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure FormShow(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  custoemr_payment: Tcustoemr_payment;

implementation

uses customer_receivables, customer_rec_fatura;

{$R *.dfm}

procedure Tcustoemr_payment.FormClose(Sender: TObject;
  var Action: TCloseAction);

begin
  with Customer_receivable.adoquery1 do
  begin
    close;
    sql.Clear;
    sql.Add('select * from Installment as ins inner join customer_rec as inv on
    ins.InvoiceNo=inv.InvoiceNo');
    execsql;
    open;
  end;
end;

procedure Tcustoemr_payment.FormShow(Sender: TObject);
begin
  edit1.Clear;
  edit2.Clear;
  edit1.Enabled:=true;
  with adoquery3 do
  begin
    close;

```

```

sql.Clear;
sql.Add('select * from Installment where InvoiceNo=:x and InstallmentNo=:y');
parameters.ParamByName('x').Value:=dbtext4.Caption;
parameters.ParamByName('y').Value:=dbtext5.Caption;
execsql;
open;
end;

with adoquery4 do
begin
close;
sql.Clear;
sql.Add('select * from customers where CustomerNo=:h');
parameters.ParamByName('h').Value:=dbtext1.Caption;
execsql;
open;
end;
label12.Caption:=inttostr(strtoint(dbtext5.Caption)+1 );
if strtoint(label12.Caption)=DataSource3.DataSet.FieldValues['Number_Of_Installment'] then
begin
showmessage('This is last installment');
dbtext7.Caption:=datasource2.DataSet.FieldValues['Expr1001'];
edit1.Text:=dbtext7.Caption;
edit1.Enabled:=false;
edit2.Text:=floattostr(datasource3.DataSet.FieldValues['RemainingPayment']-
strtofloat(edit1.Text));
label14.Caption:='0';
end ;

end;

procedure Tcustoemr_payment.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if Ord(Key)=13 then
if (strtoint(edit1.Text)<=0) then
begin
showmessage('Not valid Price');
end
else
begin
edit2.Text:=floattostr(datasource3.DataSet.FieldValues['RemainingPayment']-
strtofloat(edit1.Text));
if (strtofloat(edit2.Text)<=0) then
begin
showmessage('you may make mistake');
edit2.Clear;
end;
end;
end;
end;

```



```

procedure Tcustoemr_payment.BitBtn1Click(Sender: TObject);
var
a:word;
begin
    label13.Caption:=inttostr(datasource3.DataSet.FieldValues['Number_Of_Installment']-
    strtoint(label12.Caption));

if strtoint(label12.Caption)=DataSource3.DataSet.FieldValues['Number_Of_Installment'] then
begin
label14.Caption:='0';
end
else
begin
label14.Caption:=floattostr(strtfloat(edit2.Text)/strtfloat(label13.Caption));
end;

a:=messagedlg('Are You Sure?',mtconfirmation,[mbytes,mbno],0);
if a=mryes then
begin
    adotable1.Open;
    adotable1.Insert;
    datasource4.DataSet.FieldValues['InvoiceNo']:=dbtext4.Caption;
    datasource4.DataSet.FieldValues['CustomerNo']:=dbtext1.Caption;
    datasource4.DataSet.FieldValues['Date']:=datetimepicker1.Date;
    datasource4.DataSet.FieldValues['TotalPrice']:=dbtext6.Caption;
    datasource4.DataSet.FieldValues['Number_Of_Installment']:=dbtext8.Caption;
    datasource4.DataSet.FieldValues['Payment']:=edit1.Text;
    datasource4.DataSet.FieldValues['RemainingPayment']:= edit2.Text;
    datasource4.DataSet.FieldValues['InstallmentNo']:= label12.Caption;
    datasource4.DataSet.FieldValues['Required_Payment']:= label14.Caption;

    adotable1.post;
    close;
end
else
begin
    edit1.Clear;
    edit2.Clear;
end;
    cust_rec_fatura.quickrep1.preview;
end;

procedure Tcustoemr_payment.BitBtn2Click(Sender: TObject);
begin
    edit1.Clear;
    edit2.Clear;

```

```

end;
procedure Tcustoemr_payment.BitBtn3Click(Sender: TObject);
begin
close;
end;
end.

```

SUPPLIER PAYABLES FORM:

```

unit suppliers_payables;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, DB, ADODB, Grids, DBGrids, DBCtrls, StdCtrls, Buttons, ExtCtrls;

type
  Tsupplier_payables = class(TForm)
    GroupBox1: TGroupBox;
    Panel1: TPanel;
    Label1: TLabel;
    Label2: TLabel;
    ComboBox1: TComboBox;
    Edit1: TEdit;
    BitBtn1: TBitBtn;
    Panel2: TPanel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    DBText1: TDBText;
    DBText2: TDBText;
    Label6: TLabel;
    GroupBox2: TGroupBox;
    DBGrid1: TDBGrid;
    GroupBox3: TGroupBox;
    DBGrid2: TDBGrid;
    Edit2: TEdit;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    BitBtn4: TBitBtn;
    ADOQuery1: TADOQuery;
    DataSource2: TDataSource;
    ADOQuery2: TADOQuery;
    DataSource3: TDataSource;

```

```

ADOQuery3: TADOQuery;
DataSource1: TDataSource;
procedure ComboBox1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure DBGrid1CellClick(Column: TColumn);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  supplier_payables: Tsupplier_payables;

implementation

uses suppliers_payments;

{$R *.dfm}

procedure Tsupplier_payables.ComboBox1Click(Sender: TObject);
begin
  edit1.Clear;
end;

procedure Tsupplier_payables.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  if combobox1.ItemIndex=0 then
    begin
      if not (key in ['0'..'9',#8]) then
        begin
          Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
          Beep; //bip sesi ile kullanycyyy uyar.
        end;
      end
    else if combobox1.ItemIndex=1 then
      begin
        if not (key in ['0'..'9',#8]) then
          begin
            Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
            Beep; //bip sesi ile kullanycyyy uyar.
          end;
        end
      end
    else if combobox1.ItemIndex=2 then

```

```

begin
  if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;

```

```

end
end;

```

```

procedure Tsupplier_payables.BitBtn1Click(Sender: TObject);
begin
dbgrid1.DataSource:=datasource3;
if combobox1.Text='Bill No' then
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from Sup_payments where BillNo=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Supplier No' then
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from Sup_payments where SupplierID=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Company Name' then
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers as s inner join Sup_payments as p on
p.SupplierID=s.SupplierID where s.CompanyName=:x');
parameters.ParamByName('x').Value:= edit1.Text;

```

```

    execsql;
    open;
end;

end ;

end;

procedure Tsupplier_payables.DBGrid1 CellClick(Column: TColumn);
begin
edit2.Text:=datasource3.DataSet.FieldValues['BillNo'];
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from Sup_payments where BillNo=:p');
parameters.ParamByName('p').Value:=edit2.Text;
execsql;
open;
end;

with adoquery3 do
begin
close;
sql.Clear;
sql.Add('select sum(Payment) from Sup_payments where BillNo=:x');
parameters.ParamByName('x').Value:=edit2.Text;
execsql;
open;
end;
label6.Caption:=floattostr(strtfloat(dbtext1.Caption)-strtfloat(dbtext2.Caption));
end;

procedure Tsupplier_payables.BitBtn2Click(Sender: TObject);
begin
with supplier_payment.adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from Suppliers where SupplierID=:p');
parameters.ParamByName('p').Value:=supplier_payment.dbtext1.Caption;
execsql;
open;
end;
with supplier_payment.adoquery2 do
begin
close;

```



```

sql.Clear;
sql.Add('select max(InstallmentNo),min(RemainingPayment) from Sup_payments where
BillNo=:p');
parameters.ParamByName('p').Value:=supplier_payment.dbtext4.Caption;
execsql;
open;
end;
if supplier_payment.DataSource2.DataSet.FieldValues['Expr1001']=0 then
begin
messedlg('This account was closed!',mtwarning,[mbok],0);
end
else
begin
supplier_payment.showmodal;
end;
end;

procedure Tsupplier_payables.BitBtn3Click(Sender: TObject);
begin
with adoquery2 do
begin
close;
sql.Clear;
sql.Add('select * from Sup_payments');
execsql;
open;
end;
dbgrid1.DataSource:= datasource3;
end;
procedure Tsupplier_payables.BitBtn4Click(Sender: TObject);
begin
close;
end;
procedure Tsupplier_payables.FormClose(Sender: TObject;
var Action: TCloseAction);
begin
adoquery1.Close;
adoquery2.Close;
adoquery3.Close;
supplier_payment.ADOQuery1.Close;
supplier_payment.ADOQuery2.Close;
supplier_payment.ADOQuery3.Close;
supplier_payment.ADOQuery4.Close;

end;

end.

```

SUPPLIER PAYMENTS FORM:

unit suppliers_payments;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ADODB, DB, StdCtrls, Buttons, ComCtrls, DBCtrls, ExtCtrls;

type

Tsupplier_payment = class(TForm)

GroupBox1: TGroupBox;

Panel1: TPanel;

Label1: TLabel;

Label2: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

DBText1: TDBText;

DBText2: TDBText;

DBText4: TDBText;

DBText5: TDBText;

Label12: TLabel;

Edit3: TEdit;

Panel2: TPanel;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

Label10: TLabel;

Label11: TLabel;

DBText6: TDBText;

DBText7: TDBText;

DBText8: TDBText;

Label13: TLabel;

Label14: TLabel;

DateTimePicker1: TDateTimePicker;

Edit1: TEdit;

Edit2: TEdit;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

ADOQuery1: TADOQuery;

DataSource1: TDataSource;

ADOQuery2: TADOQuery;

DataSource2: TDataSource;

ADOQuery3: TADOQuery;

DataSource3: TDataSource;

ADOTable1: TADOTable;

DataSource4: TDataSource;

```

ADOQuery4: TADOQuery;
DataSource5: TDataSource;
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure FormShow(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  supplier_payment: Tsupplier_payment;

implementation

{$R *.dfm}

procedure Tsupplier_payment.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
begin
with supplier_payment.adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from Bills as ins inner join Sup_payments as inv on ins.BillNo=inv.BillNo');
execsql;
open;
end;
end;
end;

procedure Tsupplier_payment.FormShow(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Text:=dbtext1.Caption;

with adoquery3 do
begin
close;
sql.Clear;
sql.Add('select * from Sup_payments where BillNo=:x and InstallmentNo=:y');
parameters.ParamByName('x').Value:=dbtext4.Caption;

```

```

parameters.ParamByName('y').Value:=dbtext5.Caption;
execsql;
open;
end;
with adoquery4 do
begin
close;
sql.Clear;
sql.Add('select * from suppliers where supplierID=:h');
parameters.ParamByName('h').Value:=edit3.Text;
execsql;
open;
end;
label12.Caption:=inttostr(strtoint(dbtext5.Caption)+1 );
if strtoint(label12.Caption)=DataSource3.DataSet.FieldValues['Num_of_installment'] then
begin
showmessage('This is last installment');
dbtext7.Caption:=datasource2.DataSet.FieldValues['Expr1001'];
edit1.Text:=dbtext7.Caption;
edit1.Enabled:=false;
edit2.Text:=floattostr(datasource3.DataSet.FieldValues['RemainingPayment']-
strtoint(edit1.Text));
label14.Caption:='0';
end ;

end;
procedure Tsupplier_payment.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if Ord(Key)=13 then
if strtoint(edit1.Text)<=0 then
begin
showmessage('Not valid price');
end
else
begin
edit2.Text:=floattostr(datasource3.DataSet.FieldValues['RemainingPayment']-
strtoint(edit1.Text));
if (strtoint(edit2.Text)<=0) then
begin
showmessage('you may make mistake');
edit2.Clear;
end;
end;
end;

procedure Tsupplier_payment.BitBtn1Click(Sender: TObject);
var
a:word;
begin

```

```
label13.Caption:=inttostr(datasource3.DataSet.FieldValues['Num_of_installment']-  
strtoint(label12.Caption));
```

```
if strtoint(label12.Caption)=DataSource3.DataSet.FieldValues['Num_of_installment'] then  
begin  
label14.Caption:='0';  
end  
else  
begin  
label14.Caption:=floattostr(strtfloat(edit2.Text)/strtfloat(label13.Caption));  
end;
```

```
a:=messagedlg('Are You Sure?',mtconfirmation,[mbytes,mbno],0);  
if a=mryes then  
begin  
adotable1.Open;  
adotable1.Insert;  
datasource4.DataSet.FieldValues['BillNo']:=dbtext4.Caption;  
datasource4.DataSet.FieldValues['SupplierID']:=dbtext1.Caption;  
datasource4.DataSet.FieldValues['Date']:=datetimepicker1.Date;  
datasource4.DataSet.FieldValues['Total']:=dbtext6.Caption;  
datasource4.DataSet.FieldValues['Num_of_installment']:=dbtext8.Caption;  
datasource4.DataSet.FieldValues['Payment']:=edit1.Text;  
datasource4.DataSet.FieldValues['RemainingPayment']:= edit2.Text;  
datasource4.DataSet.FieldValues['InstallmentNo']:= label12.Caption;  
datasource4.DataSet.FieldValues['Required_payment']:= label14.Caption;
```

```
adotable1.post;  
close;  
end  
else  
begin  
edit1.Clear;  
edit2.Clear;  
end;  
end;  
procedure Tsupplier_payment.BitBtn2Click(Sender: TObject);  
begin  
edit1.Clear;  
edit2.Clear;  
end;
```

```
procedure Tsupplier_payment.BitBtn3Click(Sender: TObject);  
begin  
close;  
end;  
  
end.
```


CUSTOMER REPORTS FORM:

```
unit customer_report;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, DB, ADODB, StdCtrls, Buttons, Grids, DBGrids, ExtCtrls;
```

```
type
```

```
Tcustomer_rep = class(TForm)
```

```
  GroupBox1: TGroupBox;
```

```
  Panel1: TPanel;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  DBGrid1: TDBGrid;
```

```
  BitBtn1: TBitBtn;
```

```
  BitBtn2: TBitBtn;
```

```
  ADOQuery1: TADOQuery;
```

```
  DataSource1: TDataSource;
```

```
  RadioGroup1: TRadioGroup;
```

```
  ADOTable1: TADOTable;
```

```
  DataSource2: TDataSource;
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
  procedure FormShow(Sender: TObject);
```

```
  procedure BitBtn2Click(Sender: TObject);
```

```
  procedure RadioGroup1Click(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
end;
```

```
var
```

```
  customer_rep: Tcustomer_rep;
```

```
implementation
```

```
uses customer_named_report
```

```
, cuss_debt_rep;
```

```
{ $R *.dfm }
```

```
procedure Tcustomer_rep.BitBtn1Click(Sender: TObject);
```

```
begin
```

```
  if radiogroup1.ItemIndex=0 then
```

```

begin
customer_named_rep.ADOTable1.Active:=false;
customer_named_rep.ADOTable1.Active:=true;
customer_named_rep.QuickRep1.Preview;
end
else
if radiogroup1.ItemIndex=1 then
begin
cuss_debt.quickrep1.preview;
end;
end;

procedure Tcustomer_rep.FormShow(Sender: TObject);
begin
bitbtn2.Caption:='Main'+chr(13)+'menu' ;
bitbtn1.Caption:='Print'+chr(13)+'preview' ;
end;
procedure Tcustomer_rep.BitBtn2Click(Sender: TObject);
begin
close;
end;
procedure Tcustomer_rep.RadioGroup1Click(Sender: TObject);
var
x:integer;
begin
if radiogroup1.ItemIndex=0 then
begin
dbgrid1.DataSource:=datasource2;
x:=datasource2.DataSet.RecordCount;
label2.Caption:=inttostr(x);
end
else if radiogroup1.ItemIndex=1 then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select distinct i.CustomerNo,c.Name,c.Surname,c.Phone, c.Address, c.City from
customers as c inner join Invoice as i on i.CustomerNO=c.CustomerNO where i.taksit=true
and i.kapali=false');
execsql;
open;
end;

end;
end;
end.

```

SUPPLIER REPORT FORM:

unit supplier_reports;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, DB, ADODB, StdCtrls, Buttons, Grids, DBGrids, ExtCtrls;

type

Tsupplier_rep = class(TForm)
 GroupBox1: TGroupBox;
 Panel1: TPanel;
 Label1: TLabel;
 Label2: TLabel;
 Label3: TLabel;
 DBGrid1: TDBGrid;
 BitBtn1: TBitBtn;
 BitBtn2: TBitBtn;
 ADOQuery1: TADOQuery;
 DataSource1: TDataSource;
 procedure FormShow(Sender: TObject);
 procedure BitBtn1Click(Sender: TObject);
 procedure BitBtn2Click(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var

supplier_rep: Tsupplier_rep;

implementation

uses supplier_named_report

;

{ \$R *.dfm }

procedure Tsupplier_rep.FormShow(Sender: TObject);
begin
 bitbtn2.Caption:='Main'+chr(13)+'menu' ;
 bitbtn1.Caption:='Print'+chr(13)+'preview' ;
end;

procedure Tsupplier_rep.BitBtn1Click(Sender: TObject);
begin

```

begin
supplier_named_rep.ADOTable1.Active:=false;
supplier_named_rep.ADOTable1.Active:=true;;
supplier_named_rep.QuickRep1.Preview;
end

```

```

end;
procedure Tsupplier_rep.BitBtn2Click(Sender: TObject);
begin
close;
end;

end.

```

STOCK REPORTS FORM:

```
unit stock_report;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, DB, ADODB, StdCtrls, Buttons, Grids, DBGrids, ExtCtrls;
```

```
type
```

```

Tstock_rep = class(TForm)
  GroupBox1: TGroupBox;
  RadioGroup1: TRadioGroup;
  Panel1: TPanel;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  DBGrid1: TDBGrid;
  BitBtn1: TBitBtn;
  BitBtn2: TBitBtn;
  ADOQuery1: TADOQuery;
  DataSource1: TDataSource;
  ADOQuery2: TADOQuery;
  DataSource2: TDataSource;
  procedure RadioGroup1Click(Sender: TObject);
  procedure BitBtn2Click(Sender: TObject);
  procedure FormShow(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

var
  stock_rep: Tstock_rep;

implementation

uses all_stock_report, stock_under_min_report, stock_over_min_report,
  stock_at_min_report, stock_at_zero_report, stock_under_minimum;

{$R *.dfm}

procedure Tstock_rep.RadioGroup1Click(Sender: TObject);
var
  x:integer;
begin
  if radiogroup1.ItemIndex=0 then
  begin
    with adoquery1 do
    begin
      close;
      sql.Clear;
      sql.Add('select * from stocks ');
      execsql;
      open;
    end;
    dbgrid1.DataSource:=datasource1;
    x:=datasource1.DataSet.RecordCount;
    label2.Caption:=inttostr(x);

  end
  else
  begin
    with adoquery2 do
    begin
      close;
      sql.Clear;
      sql.Add('select * from stocks where MinimumStockLevel>Quantity') ;
      execsql;
      open;
    end;
    dbgrid1.DataSource:=datasource2;
    x:=datasource2.DataSet.RecordCount;
    label2.Caption:=inttostr(x);

  end;
end;

```



```

end;

procedure Tstock_rep.BitBtn2Click(Sender: TObject);
begin
close;
end;

procedure Tstock_rep.FormShow(Sender: TObject);
begin
bitbtn2.Caption:='Main'+chr(13)+'menu' ;
bitbtn1.Caption:='Print'+chr(13)+'preview' ;

end;

procedure Tstock_rep.BitBtn1Click(Sender: TObject);
begin
if radiogroup1.ItemIndex=0 then
begin
all_stock_rep.ADOTable1.Active:=false;
all_stock_rep.ADOTable1.Active:=true;
all_stock_rep.QuickRep1.Preview;
end
else if radiogroup1.ItemIndex=1 then
begin
stock_under_min.quickrep1.preview;
end ;

end;

end.

```

ACCOUNTING REPORTS FORM:

```
unit accounting_report;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, DB, ADODB, StdCtrls, ComCtrls, Buttons, Grids, DBGrids, ExtCtrls;
```

```
type
```

```
Taccounting_rep = class(TForm)  
  GroupBox1: TGroupBox;  
  RadioGroup1: TRadioGroup;  
  Panel1: TPanel;  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  DBGrid1: TDBGrid;  
  BitBtn1: TBitBtn;  
  BitBtn3: TBitBtn;  
  GroupBox2: TGroupBox;  
  Label4: TLabel;  
  Label5: TLabel;  
  DateTimePicker1: TDateTimePicker;  
  DateTimePicker2: TDateTimePicker;  
  BitBtn2: TBitBtn;  
  GroupBox3: TGroupBox;  
  Label6: TLabel;  
  Label7: TLabel;  
  ComboBox1: TComboBox;  
  ComboBox2: TComboBox;  
  BitBtn4: TBitBtn;  
  GroupBox4: TGroupBox;  
  Label8: TLabel;  
  Label9: TLabel;  
  BitBtn5: TBitBtn;  
  ComboBox3: TComboBox;  
  ComboBox4: TComboBox;  
  ADOQuery1: TADOQuery;  
  DataSource1: TDataSource;  
  ADOQuery2: TADOQuery;  
  ADOQuery3: TADOQuery;  
  DataSource2: TDataSource;  
  DataSource3: TDataSource;  
  GroupBox5: TGroupBox;  
  Label10: TLabel;  
  Label11: TLabel;  
  DateTimePicker3: TDateTimePicker;  
  DateTimePicker4: TDateTimePicker;
```

```

BitBtn6: TBitBtn;
procedure RadioGroup1Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);

procedure ComboBox1Select(Sender: TObject);
procedure ComboBox4Select(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  accounting_rep: Taccounting_rep;

implementation

uses customer_payment_report, supplier_payment_report, all_sales_report,
  all_purchases_report, supplier_payment_analyse_report,
  customer_payment_analyse_report, all_sales_between_report,
  all_purchases_between_report, cash_analyse_report;

{$R *.dfm}

procedure Taccounting_rep.RadioGroup1Click(Sender: TObject);
var
  x:integer;
begin
  if radiogroup1.ItemIndex=0 then
    begin

supplier_payment_analyse_rep.ADOQuery1.Close;
all_sales_rep.ADOQuery1.Close;
all_purchases_rep.ADOQuery1.Close;
all_purchases_between_rep.ADOQuery1.Close;
all_sales_between_rep.ADOQuery1.Close;
groupbox3.Visible:=true;
groupbox2.Visible:=false;
groupbox4.Visible:=false;
groupbox5.Visible:=false;
      with adoquery1 do
        begin
          close;

```

```

    sql.Clear;
    sql.Add('select * from customers as c inner join Installment as i on
i.CustomerNo=c.CustomerNo ');
    execsql;
    open;
    end;
    dbgrid1.DataSource:=datasource1;
    x:=datasource1.DataSet.RecordCount;
    label2.Caption:=inttostr(x);
end
else if radiogroup1.ItemIndex=1 then
begin
    customer_payment_analyse_rep.ADOQuery1.Close;

    all_sales_rep.ADOQuery1.Close;
    all_purchases_rep.ADOQuery1.Close;
    all_purchases_between_rep.ADOQuery1.Close;
    all_sales_between_rep.ADOQuery1.Close;

    groupbox4.Visible:=true;
    groupbox2.Visible:=false;
    groupbox3.Visible:=false;
    groupbox5.Visible:=false;
    with adoquery1 do
    begin
        close;
        sql.Clear;
        sql.Add('select * from Suppliers as s inner join Sup_payments as b on
b.SupplierID=s.SupplierID ');
        execsql;
        open;
        end;
        dbgrid1.DataSource:=datasource1;
        x:=datasource1.DataSet.RecordCount;
        label2.Caption:=inttostr(x);
    end
    else if radiogroup1.ItemIndex=2 then
    begin
        customer_payment_analyse_rep.ADOQuery1.Close;
        supplier_payment_analyse_rep.ADOQuery1.Close;

        all_purchases_rep.ADOQuery1.Close;
        all_purchases_between_rep.ADOQuery1.Close;
        all_sales_between_rep.ADOQuery1.Close;
        groupbox4.Visible:=false;
        groupbox2.Visible:=true;
        groupbox3.Visible:=false;
        groupbox5.Visible:=false;
        with adoquery1 do
        begin

```

```

close;
sql.Clear;
sql.Add('select * from Invoice');
execsql;
open;
end;
dbgrid1.DataSource:=datasource1;
x:=datasource1.DataSet.RecordCount;
label2.Caption:=inttostr(x);
end
else if radiogroup1.ItemIndex=3 then
begin
customer_payment_analyse_rep.ADOQuery1.Close;
supplier_payment_analyse_rep.ADOQuery1.Close;
all_sales_rep.ADOQuery1.Close;

all_purchases_between_rep.ADOQuery1.Close;
all_sales_between_rep.ADOQuery1.Close;
groupbox4.Visible:=false;
groupbox2.Visible:=false;
groupbox3.Visible:=false;
groupbox5.Visible:=true;
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from Bills');
execsql;
open;
end;
dbgrid1.DataSource:=datasource1;
x:=datasource1.DataSet.RecordCount;
label2.Caption:=inttostr(x);
end
else
if radiogroup1.ItemIndex=4 then
begin
groupbox4.Visible:=false;
groupbox2.Visible:=true;
groupbox3.Visible:=false;
end
else if radiogroup1.ItemIndex=5 then
begin
groupbox4.Visible:=false;
groupbox2.Visible:=false;
groupbox3.Visible:=false;
with adoquery1 do
begin
close;
sql.Clear;

```



```

    sql.Add('select * from Returned_goods');
    execsql;
    open;
end;
dbgrid1.DataSource:=datasource1;
x:=datasource1.DataSet.RecordCount;
label2.Caption:=inttostr(x);
end

end;

procedure Taccounting_rep.FormShow(Sender: TObject);
begin
bitbtn3.Caption:='Main'+chr(13)+'menu' ;
bitbtn1.Caption:='Print'+chr(13)+'preview' ;

end;

procedure Taccounting_rep.BitBtn1Click(Sender: TObject);
begin
if radiogroup1.ItemIndex=0 then
begin
customer_payment_rep.QuickRep1.Preview;
end
else if radiogroup1.ItemIndex=1 then
begin
supplier_payment_rep.QuickRep1.Preview;
end
else if radiogroup1.ItemIndex=2 then
begin
with all_sales_rep.ADOQuery1 do
begin
close;
sql.Clear;
sql.Add('select * from Invoice as i inner join s_stock as s on s.InvoiceNo=i.InvoiceNo
');
execsql;
open;
end;
all_sales_rep.QuickRep1.Preview;
end
else if radiogroup1.ItemIndex=3 then
begin
with all_purchases_rep.ADOQuery1 do
begin
close;
sql.Clear;
sql.Add('select * from Bills as b inner join p_stock as p on p.BillNo=b.BillNo');
execsql;
open;

```

```

        end;
        all_purchases_rep.QuickRep1.Preview;
        end ;
    end;

procedure Taccounting_rep.BitBtn3Click(Sender: TObject);
begin
    close;
end;

procedure Taccounting_rep.BitBtn5Click(Sender: TObject);
begin
    customer_payment_analyse_rep.ADOQuery1.Close;

    all_sales_rep.ADOQuery1.Close;
    all_purchases_rep.ADOQuery1.Close;
    all_purchases_between_rep.ADOQuery1.Close;
    all_sales_between_rep.ADOQuery1.Close;
    if combobox4.ItemIndex=0 then
        begin
            with supplier_payment_analyse_rep.adoquery1 do
                begin
                    close;
                    sql.clear;
                    sql.add('select * from Suppliers as s inner join Sup_payments as p on
p.SupplierID=s.SupplierID where s.SupplierID=:p');
                    parameters.ParamByName('p').Value:=combobox3.Text;
                    execsql;
                    open;
                    end;
            supplier_payment_analyse_rep.QuickRep1.Preview;
        end
    else if combobox4.ItemIndex=1 then
        begin
            with supplier_payment_analyse_rep.adoquery1 do
                begin
                    close;
                    sql.clear;
                    sql.add('select * from Suppliers as s inner join Sup_payments as p on
p.SupplierID=s.SupplierID where s.CompanyName=:p');
                    parameters.ParamByName('p').Value:=combobox3.Text;
                    execsql;
                    open;
                    end;
            supplier_payment_analyse_rep.QuickRep1.Preview;
        end
    else
        if combobox4.ItemIndex=2 then
            begin
                with supplier_payment_analyse_rep.adoquery1 do

```

```

begin
close;
sql.clear;
sql.add('select * from Suppliers as s inner join Sup_payments as p on
p.SupplierID=s.SupplierID where s.Name=:p');
parameters.ParamByName('p').Value:=combobox3.Text;
execsql;
open;
end;
supplier_payment_analyse_rep.QuickRep1.Preview;

end
else
if combobox4.ItemIndex=3 then
begin
with supplier_payment_analyse_rep.adoquery1 do
begin
close;
sql.clear;
sql.add('select * from Suppliers as s inner join Sup_payments as p on
p.SupplierID=s.SupplierID where s.Surname=:p');
parameters.ParamByName('p').Value:=combobox3.Text;
execsql;
open;
end;
supplier_payment_analyse_rep.QuickRep1.Preview;
end;
end;

```

```

procedure Taccounting_rep.BitBtn4Click(Sender: TObject);
begin

```

```

supplier_payment_analyse_rep.ADOQuery1.Close;
all_sales_rep.ADOQuery1.Close;
all_purchases_rep.ADOQuery1.Close;
all_purchases_between_rep.ADOQuery1.Close;
all_sales_between_rep.ADOQuery1.Close;
if combobox1.ItemIndex=0 then
begin
with customer_payment_analyse_rep.adoquery1 do
begin
close;
sql.clear;
sql.add('select * from customers as c inner join Installment as s on
s.CustomerNo=c.CustomerNo where c.CustomerNo=:p');
parameters.ParamByName('p').Value:=combobox2.Text;
execsql;
open;
end;

```

```

        customer_payment_analyse_rep.QuickRep1.Preview;
    end
    else if combobox1.ItemIndex=1 then
    begin
        with customer_payment_analyse_rep.adoquery1 do
        begin
            close;
            sql.clear;
            sql.add('select * from customers as c inner join Installment as s on
s.CustomerNo=c.CustomerNo where c.Name=:p');
            parameters.ParamByName('p').Value:=combobox2.Text;
            execsql;
            open;
            end;
            customer_payment_analyse_rep.QuickRep1.Preview;
        end
    else
        if combobox1.ItemIndex=2 then
        begin
            with customer_payment_analyse_rep.adoquery1 do
            begin
                close;
                sql.clear;
                sql.add('select * from customers as c inner join Installment as s on
s.CustomerNo=c.CustomerNo where c.Surname=:p');
                parameters.ParamByName('p').Value:=combobox2.Text;
                execsql;
                open;
                end;
                customer_payment_analyse_rep.QuickRep1.Preview;
            end ;
        end;
    end;

```

```

procedure Taccounting_rep.BitBtn2Click(Sender: TObject);
begin

```

```

    customer_payment_analyse_rep.ADOQuery1.Close;
    supplier_payment_analyse_rep.ADOQuery1.Close;
    all_sales_rep.ADOQuery1.Close;
    all_purchases_rep.ADOQuery1.Close;
    all_purchases_between_rep.ADOQuery1.Close;

```

```

    if radiogroup1.ItemIndex=2 then
    begin

```

```

        with all_sales_between_rep.ADOQuery1 do
        begin
            close;
            sql.Clear;

```



```

        sql.Add('select * from Invoice as i inner join s_stock as s on
s.InvoiceNo=i.InvoiceNo where i.Date>=:f and i.Date<=:f1');
        parameters.ParamByName('f').Value:=datetostr(datetimestr1.Datetime);
        parameters.ParamByName('f1').Value:=datetostr(datetimestr2.Datetime);
        execsql;
        open;
    end;
    all_sales_between_rep.quickrep1.preview;
end

```

```

end;

```

```

procedure Taccounting_rep.ComboBox4Select(Sender: TObject);
begin

```

```

    customer_payment_analyse_rep.ADOQuery1.Close;
    all_sales_between_rep.ADOQuery1.Close;
    all_sales_rep.ADOQuery1.Close;
    all_purchases_rep.ADOQuery1.Close;
    all_purchases_between_rep.ADOQuery1.Close;

```

```

if combobox4.ItemIndex=0 then

```

```

    begin
    combobox3.Items.Clear;
    with adoquery3 do
    begin
        sql.Clear;
        sql.add('SELECT SupplierID FROM Suppliers');
        open;
    end;
    datasource3.DataSet.First;
    while not(datasource3.DataSet.Eof) do
    begin
        combobox3.Items.Add(datasource3.DataSet.FieldValues['SupplierID']);
        datasource3.DataSet.Next;
    end;
    end

```

```

else if combobox4.ItemIndex=1 then

```

```

    begin
    combobox3.Items.Clear;
    with adoquery3 do
    begin
        sql.Clear;
        sql.add('SELECT CompanyName FROM Suppliers');
        open;
    end;

```



```

while not(datasource3.DataSet.Eof) do
begin
    combobox3.Items.Add(datasource3.DataSet.FieldValues['CompanyName']);
    datasource3.DataSet.Next;
end;
end
else if combobox4.ItemIndex=2 then
    begin
combobox3.Items.Clear;
with adoquery3 do
begin
    sql.Clear;
    sql.add('SELECT Name FROM Suppliers');
    open;
end;
while not(datasource3.DataSet.Eof) do
begin
    combobox3.Items.Add(datasource3.DataSet.FieldValues['Name']);
    datasource3.DataSet.Next;
end;
end
else if combobox4.ItemIndex=3 then
    begin
combobox3.Items.Clear;
with adoquery3 do
begin
    sql.Clear;
    sql.add('SELECT Surname FROM Suppliers');
    open;
end;
while not(datasource3.DataSet.Eof) do
begin
    combobox3.Items.Add(datasource3.DataSet.FieldValues['Surname']);
    datasource3.DataSet.Next;
end;
end;
end;
end;

```

```

procedure Taccounting_rep.ComboBox1Select(Sender: TObject);

```

```

begin
    supplier_payment_analyse_rep.ADOQuery1.Close;
    all_sales_between_rep.ADOQuery1.Close;
    all_sales_rep.ADOQuery1.Close;
    all_purchases_rep.ADOQuery1.Close;
    all_purchases_between_rep.ADOQuery1.Close;
    combobox2.Text:='choose';
    if combobox1.ItemIndex=0 then
    begin

```

```

combobox2.Items.Clear;
with adoquery2 do
begin
  sql.Clear;
  sql.add('SELECT CustomerNo FROM Customers');
  open;
end;
datasource2.DataSet.First;
while not(datasource2.DataSet.Eof) do
begin
  combobox2.Items.Add(datasource2.DataSet.FieldValues['CustomerNo']);
  datasource2.DataSet.Next;
end;
end
else if combobox1.ItemIndex=1 then
begin
combobox2.Items.Clear;
with adoquery2 do
begin
  sql.Clear;
  sql.add('SELECT Name FROM Customers');
  open;
end;
while not(datasource2.DataSet.Eof) do
begin
  combobox2.Items.Add(datasource2.DataSet.FieldValues['Name']);
  datasource2.DataSet.Next;
end;
end
else if combobox1.ItemIndex=2 then
begin
combobox2.Items.Clear;
with adoquery2 do
begin
  sql.Clear;
  sql.add('SELECT Surname FROM Customers');
  open;
end;
while not(datasource2.DataSet.Eof) do
begin
  combobox2.Items.Add(datasource2.DataSet.FieldValues['Surname']);
  datasource2.DataSet.Next;
end;
end;
end;
end;

```

```

procedure Taccounting_rep.BitBtn6Click(Sender: TObject);
begin
  customer_payment_analyse_rep.ADOQuery1.Close;
  supplier_payment_analyse_rep.ADOQuery1.Close;
  all_sales_rep.ADOQuery1.Close;
  all_purchases_rep.ADOQuery1.Close;

  all_sales_between_rep.ADOQuery1.Close;
  if radiogroup1.ItemIndex=3 then
    begin
      with all_purchases_between_rep.ADOQuery1 do
        begin
          close;
          sql.Clear;
          sql.Add('select * from Bills as b inner join p_stock as p on p.BillNo=b.BillNo where
b.Date>=:p and b.Date<=:p1');
          parameters.ParamByName('p').Value:=datetostr(datetimepicker3.Datetime);
          parameters.ParamByName('p1').Value:=datetostr(datetimepicker4.Datetime);
          execsql;
          open;
        end;
        all_purchases_between_rep.quickrep1.preview;
      end
    end;
end;

```

```

procedure Taccounting_rep.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
  customer_payment_analyse_rep.ADOQuery1.Close;
  supplier_payment_analyse_rep.ADOQuery1.Close;
  all_sales_rep.ADOQuery1.Close;
  all_purchases_rep.ADOQuery1.Close;
  all_purchases_between_rep.ADOQuery1.Close;
  all_sales_between_rep.ADOQuery1.Close;
end;

end.

```

USER SETTINGS FORM:

```
unit users_form;
```

```
interface
```

```
uses
```

```

  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ADODB, DB, StdCtrls, DBCtrls, Mask, Grids, DBGrids, Buttons,
  ExtCtrls;

```

```
type
```

```

Tusers = class(TForm)
  GroupBox1: TGroupBox;
  Panel1: TPanel;
  Label1: TLabel;
  Label2: TLabel;
  ComboBox1: TComboBox;
  Edit1: TEdit;
  BitBtn1: TBitBtn;
  Panel2: TPanel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  Label8: TLabel;
  Label9: TLabel;
  Label10: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  Label13: TLabel;
  Label14: TLabel;
  DBGrid1: TDBGrid;
  GroupBox2: TGroupBox;
  DBCheckBox1: TDBCheckBox;
  DBCheckBox2: TDBCheckBox;
  DBCheckBox3: TDBCheckBox;
  DBCheckBox4: TDBCheckBox;
  DBCheckBox5: TDBCheckBox;
  DBCheckBox6: TDBCheckBox;
  DBEdit1: TDBEdit;
  DBEdit2: TDBEdit;
  DBEdit3: TDBEdit;
  DBEdit4: TDBEdit;
  DBEdit5: TDBEdit;
  DBEdit7: TDBEdit;
  DBEdit8: TDBEdit;
  DBEdit9: TDBEdit;
  BitBtn2: TBitBtn;
  BitBtn3: TBitBtn;
  BitBtn4: TBitBtn;
  BitBtn5: TBitBtn;
  BitBtn6: TBitBtn;
  BitBtn7: TBitBtn;
  BitBtn8: TBitBtn;
  BitBtn9: TBitBtn;
  BitBtn10: TBitBtn;
  BitBtn11: TBitBtn;
  DBMemo1: TDBMemo;
  ADOTable1: TADOTable;
  DataSource1: TDataSource;

```



```

ADOQuery1: TADOQuery;
DataSource2: TDataSource;
ADOQuery2: TADOQuery;
DataSource3: TDataSource;
ADOQuery3: TADOQuery;
DataSource4: TDataSource;
procedure FormShow(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn1Click(Sender: TObject);
procedure DBEdit3KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit4KeyPress(Sender: TObject; var Key: Char);
procedure DBEdit7KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  users: Tusers;

implementation

{$R *.dfm}

procedure Tusers.FormShow(Sender: TObject);
begin
  label14.Caption:=floattostr(adotable1.RecordCount) ;
  bitbtn11.Caption:='Main'+chr(13)+'menu';
end;

procedure Tusers.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  if combobox1.ItemIndex=0 then
    begin
      if not (key in ['0'..'9','#8']) then
        begin
          Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
          Beep; //bip sesi ile kullanycyyy uyar.
        end;
      end;
    end;

```



```

end
else if combobox1.ItemIndex=1 then
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;

end
else if combobox1.ItemIndex=2 then
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end ;

end
else if combobox1.ItemIndex=3 then
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end ;
end;
end;

procedure Tusers.BitBtn1Click(Sender: TObject);
begin
dbedit1.DataSource:=datasource2;
dbedit2.DataSource:=datasource2;
dbedit3.DataSource:=datasource2;
dbedit3.DataSource:=datasource2;
dbedit5.DataSource:=datasource2;
dbmemo1.DataSource:=datasource2;
dbedit7.DataSource:=datasource2;
dbedit8.DataSource:=datasource2;
dbedit9.DataSource:=datasource2;
dbgrid1.DataSource:=datasource2;
dbcheckbox1.DataSource:=datasource2;
dbcheckbox2.DataSource:=datasource2;
dbcheckbox3.DataSource:=datasource2;
dbcheckbox4.DataSource:=datasource2;
dbcheckbox5.DataSource:=datasource2;
dbcheckbox6.DataSource:=datasource2;

if combobox1.Text='User ID' then

```

```

begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from users where UserID=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Name' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from users where User_Name=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Surname' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from users where User_Surname=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;
end
else
if combobox1.Text='Title' then
begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from users where Title=:x');
parameters.ParamByName('x').Value:= edit1.Text;
execsql;
open;
end;

```

```

end;
if adoquery1.RecordCount=0 then
messagedlg('Record not found!',mtwarning,[mbok],0);

```

```

end;

```

```

procedure Tusers.DBEdit3KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

```

```

procedure Tusers.DBEdit4KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

```

```

procedure Tusers.DBEdit7KeyPress(Sender: TObject; var Key: Char);
begin
if not (key in ['A'..'Z','a'..'z','#8']) then
begin
Key:=#0; //girilen karakter rakam veya backspace de?ilse null(#0)'a dönü?tür
Beep; //bip sesi ile kullanycyyy uyar.
end;
end;

```

```

procedure Tusers.BitBtn6Click(Sender: TObject);
var
yenino,s:integer;
begin
try
bitbtn6.Enabled:=false;
adotable1.cancel;
dbedit1.DataSource:=datasource1;
dbedit2.DataSource:=datasource1;
dbedit3.DataSource:=datasource1;
dbedit4.DataSource:=datasource1;
dbedit5.DataSource:=datasource1;
dbmemo1.DataSource:=datasource1;

```

```

dbedit7.DataSource:=datasource1;
dbedit8.DataSource:=datasource1;
dbedit9.DataSource:=datasource1;
dgrid1.DataSource:=datasource1;
dbcheckbox1.DataSource:=datasource1;
dbcheckbox2.DataSource:=datasource1;
dbcheckbox3.DataSource:=datasource1;
dbcheckbox4.DataSource:=datasource1;
dbcheckbox5.DataSource:=datasource1;
dbcheckbox6.DataSource:=datasource1;

```

```

s:=DBEdit1.Field.AsInteger;//Access 'den alınan Otomatik no..DBEdit'e alıyoruz.
yenino :=s;
if not ADOtable1.eof Then
  Begin
    ADOtable1.Last;//Tablonun son kaydına yönleneiyor.
    yenino := ADOtable1['UserID'];//Access'in Otomatik no su
    end;
    yenino := yenino + 1;//Otomatik No'nun Bir sonraki rakamı.
    Label12.Caption := FloatToStr(yenino);//Label'e alınıyor.
    begin
      ADOtable1.Append;
      ADOtable1.Edit;
      DBEdit1.Field.Asstring:=Label12.Caption;//Labelden alınıp Tekrar yeni no veriliyor
    end;
  except
    on EDatabaseError do showmessage('You have to fill required data');
  end;
end;

```

```

procedure Tusers.BitBtn7Click(Sender: TObject);

```

```

begin
try
  if ((dbedit1.Text<>"")and(dbedit2.Text<>"")and (dbedit3.Text<>"")and
(dbedit4.Text<>"")and (dbedit5.Text<>"")and(dbmemo1.Text<>"")and (dbedit7.Text<>"")and
(dbedit8.Text<>"") and (dbedit9.Text<>"")) then
    begin
      with adoquery2 do
      begin
        close;
        sql.Clear;
        sql.Add('select * from users where User_name=:p and password=:p1');
        parameters.ParamByName('p').Value:=dbedit3.Text;
        parameters.ParamByName('p1').Value:=dbedit2.Text;
        execsql;
      end;
    end;
  end;

```



```

        open;
    end;
    if adoquery2.RecordCount<>0 then
    begin
        messagedlg('This Username or password in use',mtwarning,[mbok],0);
        adotable1.Cancel;
    end
    else
    begin
        adotable1.Post;
        showmessage ('user added succesfully. ');
        adotable1.Insert;
        bitbtn6.Enabled:=true;
    end ;
end
else
begin
    messagedlg('Please fill all required fields!',mtwarning,[mbok],0);
end;

label14.Caption:=floattostr(adotable1.RecordCount) ;

except
on EDatabaseError do showmessage('You should add empty space before add any record. ');
end;

end;

procedure Tusers.BitBtn8Click(Sender: TObject);
begin
    if ((dbedit1.Text="" or (dbedit2.Text="" or (dbedit3.Text="" or (dbedit4.Text="" or
(dbedit5.Text="" or (dbmemo1.Text="" or (dbedit7.Text="" or (dbedit8.Text="" or
(dbedit9.Text="")) then
    begin
        messagedlg('There is no added record !',mtwarning,[mbok],0);
    end
    else
    begin
        adotable1.Edit;
        adotable1.UpdateRecord;
        showmessage ('User record updated succesfully. ');
    end;
end;

procedure Tusers.BitBtn9Click(Sender: TObject);
var
x:word;

```



```

begin
try
x:=messagedlg('Are you sure to delete this user?',mtwarning,[mbytes,mbno],0);
if x=mryes then
begin
if (((dbedit1.Text='') or (dbedit2.Text='') or (dbedit3.Text='') or (dbedit4.Text='') or
(dbedit5.Text='') or (dbmemo1.Text='') or (dbedit7.Text='') or (dbedit8.Text='') or
(dbedit9.Text='')) then
begin
messagedlg('you can not delete on empty datasheet',mtwarning,[mbok],0);
end
else
begin
with adoquery3 do
begin
close;
sql.Clear;
sql.Add('delete from users where UserID=:k');
parameters.ParamByName('k').Value:=dbedit1.Text;
execsql;
end;
showmessage('user deleted successfully.');
```

label14.Caption:=floattostr(adotable1.RecordCount) ;

```

end;
end;
except
on EDatabaseError do showmessage('you can not delete on empty datasheet');
end;

end;

procedure Tusers.BitBtn10Click(Sender: TObject);
begin
adotable1.Cancel;
bitbtn6.Enabled:=true;
end;

procedure Tusers.BitBtn11Click(Sender: TObject);
begin
close;
end;

procedure Tusers.BitBtn2Click(Sender: TObject);
begin
adotable1.First;
end;

```

```

procedure Tusers.BitBtn3Click(Sender: TObject);
begin
  adotable1.prior;
end;

procedure Tusers.BitBtn4Click(Sender: TObject);
begin
  adotable1.next;
end;

procedure Tusers.BitBtn5Click(Sender: TObject);
begin
  adotable1.last;
end;

end.

```

DEFINE CATEGORIES FORM:

```

unit categories_form;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ADODB, DB, StdCtrls, DBCtrls, Buttons, Grids, DBGrids, Mask;

type
  Tcategories = class(TForm)
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    DBGrid1: TDBGrid;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    BitBtn4: TBitBtn;
    BitBtn6: TBitBtn;
    BitBtn7: TBitBtn;
    BitBtn8: TBitBtn;
    BitBtn9: TBitBtn;
    BitBtn10: TBitBtn;
    Edit1: TEdit;
    BitBtn11: TBitBtn;
    ADOTable1: TADOTable;

```

```

DataSource1: TDataSource;
ADOQuery1: TADOQuery;
DataSource2: TDataSource;
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure FormShow(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  categories: Tcategories;

implementation

uses stocks_form;

{$R *.dfm}

procedure Tcategories.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  stocks.label10.Caption:=floattostr(stocks.adotable1.RecordCount) ;
  stocks.combobox2.Items.Clear;
  with stocks.adoquery2 do
  begin
    sql.Clear;
    sql.add('SELECT CategoryName FROM Categories');
    open;
  end;
  stocks.datasource3.DataSet.First;
  while not(stocks.datasource3.DataSet.Eof) do
  begin
    stocks.combobox2.Items.Add(stocks.datasource3.DataSet.FieldValues['CategoryName']);
    stocks.datasource3.DataSet.Next;
  end;
end;

```

```

procedure Tcategories.FormShow(Sender: TObject);
begin
label3.Caption:=floattostr(adotable1.RecordCount) ;
bitbtn11.Caption:='Main'+Chr(13)+'menu';

end;

procedure Tcategories.BitBtn6Click(Sender: TObject);
var
yenino,s:integer;
begin
s:=DBEdit1.Field.AsInteger;//Access 'den alınan Otomatik no..DBEdit'e alıyoruz.
yenino :=s;
if not ADOtable1.eof Then
Begin
ADOtable1.Last;//Tablonun son kayıtına yönleneiyor.
yenino := ADOtable1['CategoryID'];//Access'in Otomatik no su
end;
yenino := yenino + 1;//Otomatik No'nun Bir sonraki rakamı.
Label3.Caption := FloatToStr(yenino);//Label'e alınıyor.
begin
ADOtable1.Append;
ADOtable1.Edit;
DBEdit1.Field.Asstring:=Label3.Caption;//Labelden alınıp Tekrar yeni no veriliyor
end;
end;

procedure Tcategories.BitBtn7Click(Sender: TObject);
begin
if ((dbedit1.Text<>"")and (dbedit2.Text<>"")) then
begin
adotable1.Post;
showmessage ('category added succesfully. ');
adotable1.Insert;
label3.Caption:=floattostr(adotable1.RecordCount) ;
end
else
begin
messagedlg('Please fill all required fields!',mtwarning,[mbok],0);
end;

end;

procedure Tcategories.BitBtn8Click(Sender: TObject);
var
x:word;
begin
try
edit1.text:=dbedit1.text;
x:=messagedlg('Are you sure to delete this category?',mtwarning,[mbyes,mbno],0);
if x=mryes then

```

```

begin
with adoquery1 do
begin
close;
sql.Clear;
sql.Add('select * from stocks where CategoryID=:p');
parameters.ParamByName('p').Value:=edit1.text;
execsql;
open;
end;
if adoquery1.Found=true then
begin
showmessage('You can not delete this category!');
end
else
begin
adotable1.Delete;
showmessage('category deleted successfully. ');
label3.Caption:=floattostr(adotable1.RecordCount) ;
end;
end ;

except
on EDatabaseError do showmessage('you can not delete on empty datasheet');
end;

end;

procedure Tcategories.BitBtn9Click(Sender: TObject);
begin
adotable1.Edit;
adotable1.UpdateRecord;
showmessage ('category updated succesfully. ');
end;
procedure Tcategories.BitBtn10Click(Sender: TObject);
begin
adotable1.Cancel;
end;

procedure Tcategories.BitBtn11Click(Sender: TObject);
begin
CLOSE;
end;

procedure Tcategories.BitBtn1Click(Sender: TObject);
begin
adotable1.First;
end;

```



```
procedure Tcategories.BitBtn2Click(Sender: TObject);  
begin  
  adotable1.prior;  
end;
```

```
procedure Tcategories.BitBtn3Click(Sender: TObject);  
begin  
  adotable1.next;  
end;
```

```
procedure Tcategories.BitBtn4Click(Sender: TObject);  
begin  
  adotable1.last;  
end;
```

```
end.
```