

NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

**DORMITORY CONTROL SYSTEM
IN VISUAL BASIC PROGRAMMING**

**Graduation Project
COM-400**

Student: Erhan Öztürk (971387)

Supervisor: Mr. Ümit İlhan

Nicosia- 2002

ACKNOWLEDGEMENT

At first, thanks to Mr. Ümit İlhan, my advisor for the project. Also thanks to my parents as they bring me the chance to graduate with their economical aid and morale that they gave.

Erhan ÖZTÜRK

ABSTRACT

This project is prepared for controlling a dormitory of a school by using a program, so that the control of the dormitory was automated. With the power of automation the risk of confusion for the people is reduced. This program is written using Visual Basic 6.0 language, which is considered as one of the most intelligent visual languages. The data was manipulated using the manager option in Visual Basic and formatted with the Microsoft Access 7.0, which is a sophisticated database program.

In this project the program records the students in dormitory according to their room numbers and shows them. There are functions used for recording entering and leaving processes of students, manipulating the records and deletion process. The program shows the student's and their rooms, which student is inside, which student is outside, which student has permission or not permitted, which student is punished and additive controls for visitors.

The explanation of every command is given in the following sections. Even a person who didn't use Visual Basic can understand the function and the process easily.

TABLE OF CONTENTS

ACKNOWLEDGEMENT	i
ABSTRACT	ii
TABLE OF CONTENTS	iii
INTRODUCTION	v
1. APPLICATION OF VISUAL BASIC	1
1.1. Writing Code	4
1.2. Visual Basic Data Types	4
1.3. Creating a Database	5
2. DAO DATA CONTROL	5
2.1. DAO Data Control Properties	6
2.2. Recordset Object	7
2.3. Data Bound Controls	8
2.4. Data Bound Control Properties	9
2.5. DAO Data Control Events	10
2.6. DAO Data Control Methods	10
2.7. DAO Data Control Recordset Properties	11
2.8. DAO Data Control Recordset Methods	12
2.9. DAO Data Control Recordset Navigation	13
3. ADO DATA CONTROL	13
3.1. ADO Data Control Properties	14
3.2. ConnectionString Property	15
3.3. Recordset Object	16
3.4. Data Bound Controls	17
3.5. Data Bound Control Properties	18
3.6. ADO Data Control Events	19
3.7. ADO Data Control Method	20
3.8. ADO Data Control Recordset Properties	20
3.9. ADO Data Control Recordset Methods	21
3.10. ADO Data Control Recordset Navigation	22
4. DAO or ADO – WHAT’S THE DIFFERENCE?	22

5. COMMAND BUTTON CONTROL	23
6. LABEL CONTROL	23
7. TEXT BOX CONTROL	23
8. CHECK BOX CONTROL	23
9. OPTION BUTTON CONTROL	24
10.FRAME CONTROL	24
11.STUDENT DATABASE	25
12.VISITOR DATABASE	25
13.DORMITORY CONTROL SYSTEM	26
14.MAIN FORM	27
15.FILE	35
15.1. New Record	35
15.2. Update	44
15.3. Delete	55
16.LIST AND SEARCH	60
17.CONTROL	64
17.1. Check Student And Rooms	64
17.2. Search By Room Number	81
18.PERMIT	83
18.1. Permission Situation	83
18.2. List Of On Leaves	87
19.PENALTY	89
19.1. Operation Of Penalty	89
19.2. List Of Punishment	96
20.STUDENT CONTROL	98
21.VISITOR	101
CONCLUSION	110
REFERENCES	111

INTRODUCTION

I have learned Visual Basic program previous semester. It was the most exciting computer language product to hit market in quite a while. The press had rarely been excited by a product so what was all the type about? Exactly what is Visual Basic and what can it my do? Well, it is an essay -to-use, yet extra ordinarily powerful tool for developing Windows applications. Before Visual Basic was introduced, developing Windows applications was much harder then developing Dos applications. Programmers had to worry about much, such as what the mouse was doing, where the users was inside a menu, and whether he or she was clicking or double-clicking at a given place. Developnig a Windows application required expert C programmers and hundres of lines code for the simple task even the experts had trouble. Visual Basic makes it easy to design the screen you literally draw the user interface, almost as if you were using a point program. In addition, when you have finished drawing the interface, the command buttons and other controls that you have placed in a blank Window will automatically recognize user actions such as mouse movements and button clicks. Also comes with a menu design feature that makes creating both ordinary and pop-up a snap.

Because of reasons I want to developed my skills working on Visual basic. My project is dormitory control system. Visual Basic have a data manager I used database management which Microsoft Access 7.0. more sophisticated databases, like the ones you can begin to build with the data manager (build completely with Microsoft Access or the data access power of Visual Basic Professional) don't fit indexed card problem. This makes it easy to avoid the update problem. They have many other advantages as well. There really is no convenient way to describe the underlying structure of the databases that you can build using the Access engine supplied with Visula Basic; that is what actually lies on the user's hard disk.

I have learned how is connect database management and Visual Basic projects. In program, users may make any operations for example new account, update account, delete account, search account etc. Generally commercial programs are write in Visual programs for this reason basic project. My program is may use to commercial work.

A lot of company want to have visual programmers. So after I graduate "gradution project" studies to get many advantages to me.

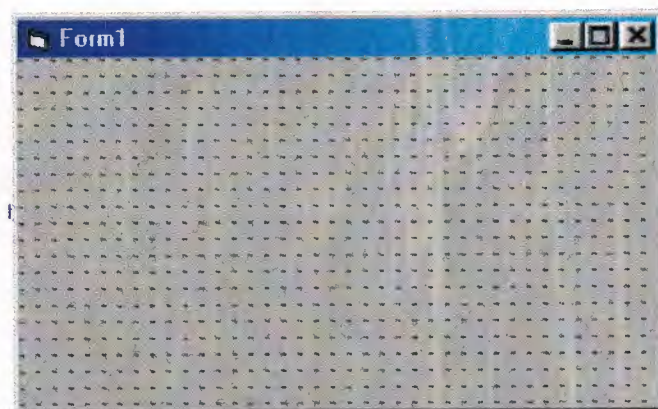
Application of Visual Basic

Six windows should appear when you start Visual Basic. If any of these windows do not appear, they may be accessed using the main window menu **View** item.

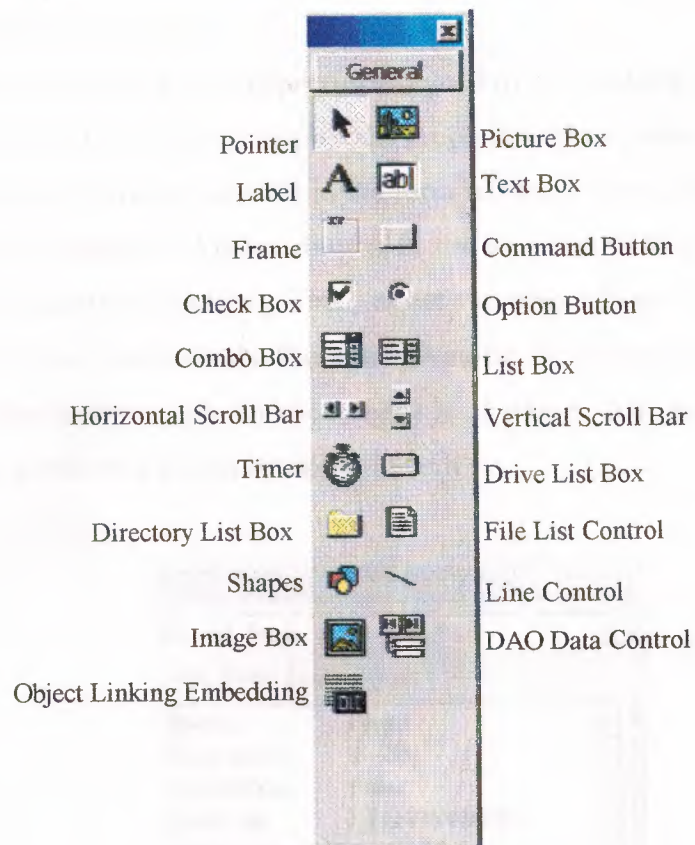
The **Main Window** consists of the title bar, menu bar, and toolbar. The title bar indicates the project name, the current Visual Basic operating mode, and the current form. The menu bar has drop-down menus from which you control the operation of the Visual Basic environment. The toolbar has buttons that provide shortcuts to some of the menu options (ToolTips indicate their function). The main window also shows the location of the current form relative to the upper left corner of the screen (measured in twips) and the width and length of the current form.



The **Form Window** is central to developing Visual Basic applications. It is where you draw your application.



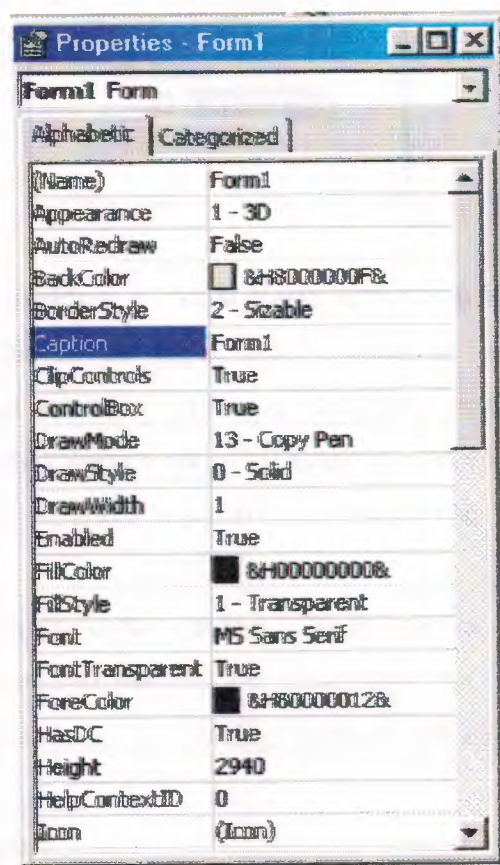
The **Toolbox** is the selection menu for controls (objects) used in your application.



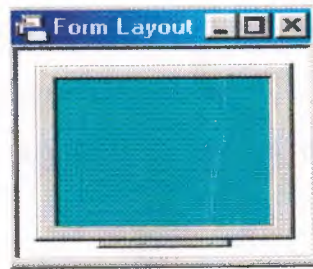
The **Properties Window** is used to establish initial property values for objects. The drop-down box at the top of the window lists all objects in the current form. Two views are available: **Alphabetic** and **Categorized**. Under this box are the available properties for the currently selected object.

Each form and control has **properties** assigned to it by default when you start a new project. There are two ways to display the properties of an object. The first way is to click on the object (form or control) in the form window. Then, click on the Properties Window or the Properties Window button in the tool bar. The second way is to first click on the Properties Window. Then, select the object from the **Object** box in the Properties Window. Shown is the Properties Window for a new application:

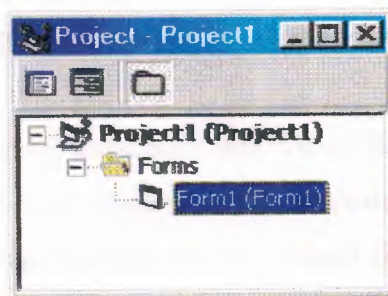
A very important property for each object is its **name**. The name is used by Visual Basic to refer to a particular object in code



The **Form Layout Window** shows where (upon program execution) your form will be displayed relative to your monitor's screen:



The **Project Explorer Window** displays a list of all forms and modules making up your application. You can also obtain a view of the **Form** or **Code** windows (window containing the actual Basic coding) from the Project Explorer window.



Writing Code

The last step in building a Visual Basic application is to write code using the **BASIC** language. This is the most time consuming task in any Visual Basic application, not just ones involving databases. As objects are added to the form, Visual Basic automatically builds a framework of all event procedures. We simply add code to the event procedures we want our application to respond to. And, if needed, we write general procedures.

Code is placed in the **code window**. At the top of the code window are two boxes, the **object** (or control) **list** and the **procedure list**. Select an object and the corresponding event procedure. A blank procedure will appear in the window where you write BASIC code

Visual Basic Data Types

- ⇒ Boolean (True or False)
- ⇒ Integer (Whole numbers)
- ⇒ Long (Large whole numbers)
- ⇒ Single (Floating point numbers)
- ⇒ Double (Large floating point numbers)

- ⇒ Currency
- ⇒ Date
- ⇒ Object (yes, objects can be variables!)
- ⇒ String (Used for many control properties)
- ⇒ Variant (A chameleon, becomes what it needs to be)

Creating a Database

Databases are created using commercial applications like Access, dBase, FoxPro, Oracle, and others. Each of these products has a design mode where you define a table and the fields that are part of the table. You can also enter records into the table using these applications. The books database was built with Access. In the first part of this course, we will work with existing databases and will not be concerned with creating a database.

It is possible to create databases with Visual Basic (we'll look at how to do this in the final chapter). For now, when we need to create our own database, we will use a product shipped with Visual Basic called the **Visual Data Manager**. It is a fairly easy-to-use application that will suit our needs quite well. If you know how to use Access, you could also use that when the time for creating a database arises.

DAO Data Control

- The **DAO data control** is selected from the Visual Basic toolbox window. It's icon looks like this:



- The DAO data control is the primary interface between a Visual Basic application and a database. It can be used without writing any code at all or, it can be a central part of a complex database management system.
- The data control (or tool) can access databases created by other programs besides Visual Basic (or Microsoft Access). Some other formats supported include dBase, FoxPro, and Paradox.
- The data control can perform the following tasks:
 1. Connect to a database.
 2. Open a specified database table.
 3. Create a virtual table based on a database query.

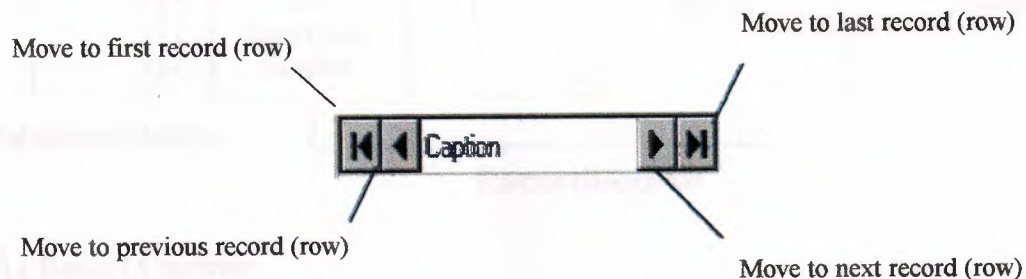
4. Pass database fields to other Visual Basic tools, for display or editing. Such tools are bound to the database, or **data bound controls**.
 5. Add new records, delete records, or update records.
 6. Trap any errors that may occur while accessing data.
 7. Close the database.
- As a rule, you need one data control for every database table, or virtual table, you need access to. One row of the table is accessible to each data control at any one time. This is referred to as the **current record**.

DAO Data Control Properties

- The DAO data control is connected to a database simply by setting a few properties. Important properties of this data control are:

Align	Determines where data control is displayed.
Caption	Phrase displayed on the data control.
Connect	Type of database. Default is Microsoft Access (or Jet).
DatabaseName	Returns or sets the name of the source database for the data control. Must be a fully qualified path and file name.
EditMode	Read-only at run-time. Indicates current state of editing for the current record.
Exclusive	Indicates whether the underlying database is opened for single-user or multi-user access.
ReadOnly	Indicates whether the data can be edited or not.
Recordset	A set of records defined by a data control's Connect, DatabaseName, and RecordSource properties. Run-time only.
RecordsetType	Indicates type of Recordset you want data control to create
RecordSource	Determines the table (or virtual table) the data control is attached to.
Visible	Establishes whether the data control appears on the form at run-time.

- When a DAO data control is placed on a form, it appears with the assigned caption and four arrow buttons:

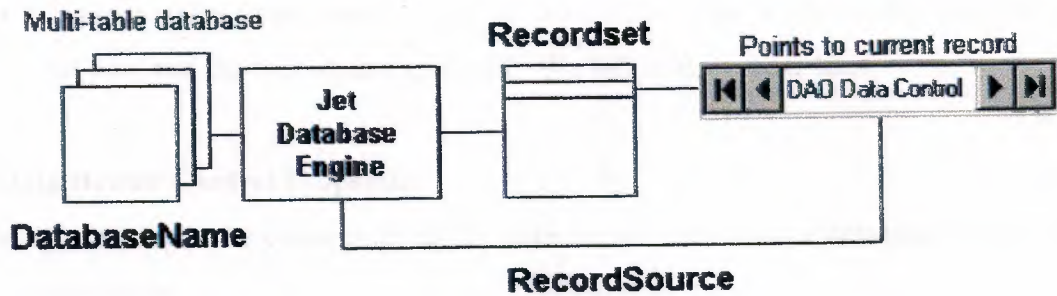


The arrows are used to navigate through the table records (rows). As indicated, the buttons can be used to move to the beginning of the table, the end of the table, or from record to record. In most applications, the data control never appears on the form – its **Visible** property is almost always **False**. In this case, moving from record to record is handled programmatically, a topic discussed later in this chapter.

- After placing a DAO data control on a form, set the **DatabaseName** property first. Simply click on the ellipsis in the property box and choose the database. Then, set the **RecordSource** property. Click on that property's drop-down arrow and a list of valid tables will be presented. Choose the desired table (or type in a valid SQL statement - studied in Chapter 5). This establishes the **Recordset** object. By following these steps carefully, we avoid run-time errors associated with inability to find referenced data.

Recordset Object

- The **Recordset** object is an important concept. When we set the **RecordSource** property (either select a table from the database or form a virtual table via a query), the data control (using the Jet engine) retrieves the needed records and places them in the **Recordset** object for our use. We will see that this object has its own **properties** and **methods** for our use.
- In summary, the relationship between the **data control**, its two primary properties (**DatabaseName** and **RecordSource**), and the **Recordset** object is:



Data Bound Controls

- The DAO data control allows us to easily connect to a database and form a Recordset. Yet, that control alone does not provide us with anyway to view the information in the database. To view the information, we use **data bound controls** that are special controls with properties established by database fields. A data bound control is needed for each field (column) in the Recordset (database table) you need to view. Most of the standard Visual Basic tools can be used as **data bound controls**.

- Standard data bound data controls are:

Label	Can be used to provide display-only access to a specified text data field. Caption property is data bound.
Text Box	Can be used to provide read/write access to a specified text data field. Probably, the most widely used data bound tool. Text property is data bound.
Check Box	Used to provide read/write access to a Boolean field. Value property is data bound.
Picture Box	Used to display a graphical image from a bitmap, icon, gif, jpeg, or metafile file. Provides read/write access to a image/binary data field. Picture property is data bound.
Image Box	Used to display a graphical image from a bitmap, icon, gif, jpeg, or metafile file (uses fewer resources than a picture box). Provides read/write access to a image/binary data field. Picture property is data bound.

- There are also three 'custom' data bound controls, the bound combo box, the bound list box, and the bound data grid tool. We will look at these later.

Data Bound Control Properties

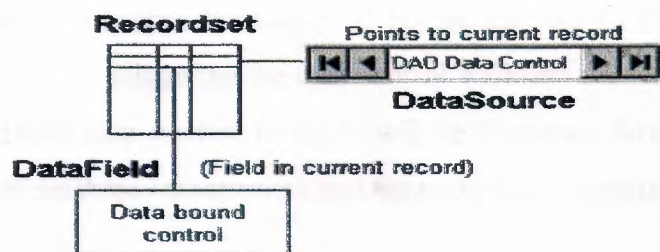
- To establish the connection of the data bound control to a database, we use a few properties:

DataChanged	Indicates whether a value displayed in a bound control has changed.
DataField	Specifies the name of a field in the table pointed to by the respective data control.
DataSource	Specifies which data control the control is bound to (indirectly specifying the database table).

- If the data in any data bound control is changed and the user moves to another record in the database, the database will **automatically be updated** with the new data (assuming it is not ReadOnly). Be aware of this - it is an extremely powerful feature of the data control, but also a potential source of problems.
- To make using bound controls easy, follow these steps (in order listed) in placing the controls on a form:
 1. Draw the bound control on the same form as the data control to which it will be bound.
 2. Set the **DataSource** property. Click on the drop-down arrow to list the data controls on your form. Choose one.
 3. Set the **DataField** property. Click on the drop-down arrow to list the fields associated with the selected data control records. Make your choice.
 4. Set all other properties, as needed.

Again, by following these steps in order, we avoid potential data access errors.

- The relationships between a data bound control (**DataSource** and **DataField** properties) and the DAO data control (**Recordset** property) are:



DAO Data Control Events

- Like other controls, the DAO data control has **events** that are triggered at various times during database access. In these events, we write BASIC code to perform specific needed tasks. In this chapter, we will not be using these event procedures, but we will define them to make our definition of the data control complete.

- Important DAO data control events:

Error	Triggered when a data access error occurs and Visual Basic code is not being executed.
Reposition	Triggered after data control pointer moves to a new record. Use to update information from non-data bound controls.
Validate	Event triggered when the pointer is about to move away from the current record. This event can be used to cancel an update of a record or a move to a new record.

- These events will be discussed further when we begin development of database management techniques in a later chapter.

DAO Data Control Methods

- To complete our definition of the DAO data control, we present some important methods. These methods perform certain actions on the data control:

Refresh	Requeries the database based on contents of the RecordSource property.
UpdateControls	Restores the value of bound controls to original values (if no update has been performed).

UpdateRecord Saves the values of bound controls to the database without triggering the data control **Validate** event.

- Like events, DAO data control methods will be discussed further when we begin development of database management techniques in a later chapter.

DAO Data Control Recordset Properties

- The **Recordset** object of the data control has its own set of **properties**. These properties can only be accessed at **run-time**. To refer to a Recordset property, use a 'double-dot' notation. For example, if you have a data control named **datExample**, to refer to a property named **PropertyName**, the notation is:

datExample.Recordset.PropertyName

- Important data control Recordset properties are:

AbsolutePosition Long integer that either gets or sets the position of the current record.

BOF Returns True when the current record is positioned before any data.

Bookmark Sets or returns a bookmark to the current record. Used as a place marker in database management tasks.

EditMode Indicates the state of editing for the current record.

EOF Returns True when the current record is positioned past any data.

PercentPosition Single data type that sets or gets the position of the current record as a percentage of total records. Used for status indicators.

RecordCount The total number of records in the Recordset.

Updatable Read-only at run-time. If True, records in the Recordset can be modified. If False, records are read-only.

- We will look at the BOF and EOF properties in the section on Recordset Navigation. Other properties will be examined later in this course.

DAO Data Control Recordset Methods

- The data control **Recordset** also has its own set of **methods** that perform functions on the Recordset. These methods are invoked using the double-dot notation introduced for the Recordset properties. So, for a data control (**datExample**) and method (**MethodName**), you invoke the method via:

datExample.Recordset.MethodName

- Important Recordset methods are:

AddNew	Adds a new record to the Recordset. All fields are set to null and this record becomes the current record.
CancelUpdate	Used to cancel any pending updates (either with Edit or AddNew method)
Close	Closes a Recordset.
Delete	The current record is deleted from the Recordset.
Edit	Places the current record in the Recordset into edit mode.
MoveFirst	Moves the current record pointer to the first record in the Recordset.
MoveLast	Moves the current record pointer to the last record in the Recordset.
MoveNext	Moves the current record pointer to the next record in the Recordset.
MovePrevious	Moves the current record pointer to the previous record in the Recordset.
Requery	Updates the data in a Recordset object by re-executing the query on which the object is based.
Update	Saves the current contents of all data bound controls.

- We will look at the four 'Move' methods in the next section on Recordset Navigation. Other properties will be reviewed later in this course.

DAO Data Control Recordset Navigation

- We have seen that, on the form, the DAO data control has four arrows that allow the user to move to the first, next, previous, and last records in the recordset. Unfortunately, this control does not have a familiar look to a user and it may not be clear just exactly what functions the arrows perform. For this reason, we usually set the data control's **Visible** property to **False** and provide a programmatic approach to moving from record to record, or **recordset navigation**.
- Four Recordset methods replicate the capabilities of the arrow buttons on the data control: **MoveFirst**, **MoveNext**, **MovePrevious**, and **MoveLast**. For each function we need, a command button is added to the form, with a **Click** event procedure attached to the corresponding Recordset method.
- When programmatically navigating through the records, you need to be aware of the position of the current record. For example, if you are at the first record and try a **MovePrevious** method, you will move past the beginning of the file. You can use the **BOF** property to see you are at the beginning of file and disallow such a move. You need a similar check at the end of a file to disallow an invalid **MoveNext** method.

ADO Data Control

- The **ADO** (ActiveX Data Object) **data control** is the primary interface between a Visual Basic application and a database. It can be used without writing any code at all! Or, it can be a central part of a complex database management system. The ADO data control does not appear in the standard Visual Basic toolbox - it must be added. Select **Project** from the main menu, then click **Components**. The Components window will appear. Select **Microsoft ADO Data Control**, then click **OK**. The control will be added to your toolbox. Its icon appears as:



- The data control (or tool) can access databases created by other programs besides Visual Basic (or Microsoft Access). Some other formats supported include dBase, FoxPro, and Paradox.

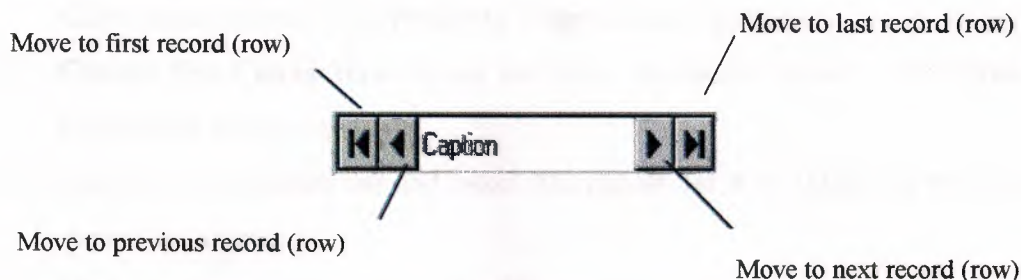
- The data control can perform the following tasks:
 1. Connect to a database.
 2. Open a specified database table.
 3. Create a virtual table based on a database query.
 4. Pass database fields to other Visual Basic tools, for display or editing. Such tools are bound to the database, or **data bound controls**.
 5. Add new records, delete records, or update records.
 6. Trap any errors that may occur while accessing data.
 7. Close the database.
- As a rule, you need one data control for every database table, or virtual table, you need access to. One row of the table is accessible to each data control at any one time. This is referred to as the **current record**.

ADO Data Control Properties

- The ADO data control is connected to a database simply by setting a few properties. Important properties of this data control are:

Align	Determines where data control is displayed.
Caption	Phrase displayed on the data control.
CommandType	Establishes source of Recordset (table or query).
ConnectionString	Contains the information used to establish a connection to a database.
EditMode	Read-only at run-time. Indicates current state of editing for the current record.
LockType	Indicates the type of locks placed on records during editing (default setting makes databases read-only).
Recordset	A set of records defined by a data control's ConnectionString and RecordSource properties. Run-time only.
RecordSource	Determines the table (or virtual table) the data control is attached to.
Visible	Establishes whether the data control appears on the form at run-time.

- When an ADO data control is placed on a form, it appears with the assigned caption and four arrow buttons:



The arrows are used to navigate through the table records (rows). As indicated, the buttons can be used to move to the beginning of the table, the end of the table, or from record to record. In most applications, the data control never appears on the form – its **Visible** property is almost always **False**. In this case, moving from record to record is handled programmatically, a topic discussed later in this chapter.

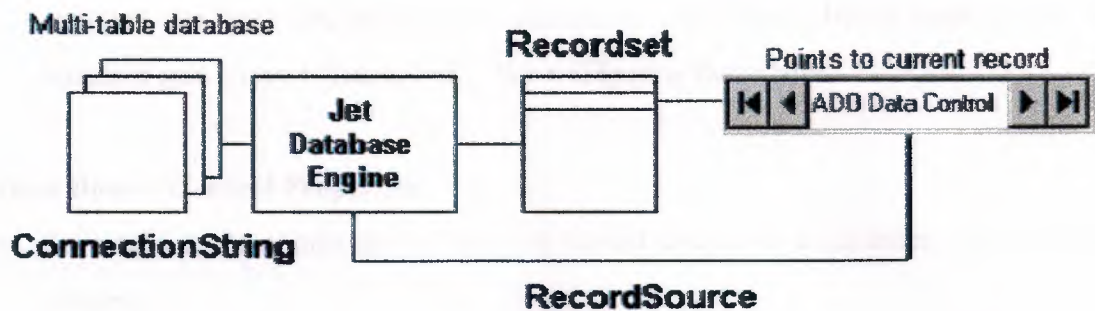
ConnectionString Property

- After placing a data control on a form, you set the **ConnectionString** property. The ADO data control can connect to a variety of database types. There are three ways to connect to a database: using a data link, using an ODBC data source, or using a connection string. For now, we will look only at connection to a Microsoft Access database using a **connection string**.
- Visual Basic can build the connection string for us. This process is best illustrated by example. We will use the books database (**BIBLIO.MDB**) discussed in Chapter 2. This database is shipped with Visual Basic and is usually installed in the Visual Basic main directory. Using Windows Explorer, find this file. Make a copy of the database and place it in a working directory (you decide on a name – we use c:\vbdb\working) where you will build your applications. We do this to insure there is always a valid copy of BIBLIO.MDB on your computer. You will see that the power of the ADO control also opens up the possibility of doing damage to a database (we, of course, will try to minimize this possibility). So, we are just living by the adage, “Better safe, than sorry.”
- Now, the steps to create our example connection string are:

1. Start a new Visual Basic project and place an ADO Data Control on the form.
2. Go to the **Properties** Window, click on the **Connection String**. Click on the ellipsis that appears. The **Property Pages** window appears.
3. Choose **Use Connection String** and click the **Build** button. The **Data Link Properties** window appears.
4. Choose the **Provider** tab and select **Microsoft Jet 3.51 OLE DB Provider** (an Access database).
5. Click the **Next** button to go to the **Connection** tab.
6. Click the ellipsis and use the **Select Access Database** dialog box to choose the **BIBLIO.MDB** file in your working directory. Click **Open**.
7. Click **Test Connection**. Then, click **OK** (assuming it passed). Click **OK** in the Property Pages window. The connection string is built and assigned to the **ConnectionString** property of the data control.

Recordset Object

- The **Recordset** object is an important concept. When we set the **RecordSource** property (either select a table from the database or form a virtual table via a query), the data control (using the Jet engine) retrieves the needed records and places them in the **Recordset** object for our use. We will see that this object has its own **properties** and **methods** for our use.
- In this chapter, the Recordset will be one of the native tables in the database. Continuing with the **BIBLIO.MDB** example, let's connect to the **Titles** table:
 1. Go to the **Properties** window for the data control. Select **2-adCmdTable** for the **CommandType** property for the data control (this tells the data control we will be using a native table). Now, click the **RecordSource** property. Click on the ellipsis that appears. The **Property Pages** window appears.
 2. Click the drop-down button under **Tables or Stored Procedure Name**. Choose **Titles**.
 3. Click **OK** in the Property Pages window. The **RecordSource** property of the data control is set, establishing the **Recordset** as the **Titles** table of the books database.
- In summary, the relationship between the **data control**, its two primary properties (**Connection String** and **RecordSource**), and the **Recordset** object is:



Data Bound Controls

- The ADO data control allows us to easily connect to a database and form a Recordset. Yet, that control alone does not provide us with any way to view the information in the database. To view the information, we use **data bound controls** that are special controls with properties established by database fields. A data bound control is needed for each field (column) in the Recordset (database table) you need to view. Most of the standard Visual Basic tools can be used as **data bound controls**.

Standard data bound data controls are:

Label	Can be used to provide display-only access to a specified text data field. Caption property is data bound.
Text Box	Can be used to provide read/write access to a specified text data field. Probably, the most widely used data bound tool. Text property is data bound.
Check Box	Used to provide read/write access to a Boolean field. Value property is data bound.
Picture Box	Used to display a graphical image from a bitmap, icon, gif, jpeg, or metafile file. Provides read/write access to a image/binary data field. Picture property is data bound.
Image Box	Used to display a graphical image from a bitmap, icon, gif, jpeg, or metafile file (uses fewer resources than a picture box). Provides read/write access to a image/binary data field. Picture property is data bound.

- There are also three 'custom' data bound controls, data bound versions of the standard list box (**DataList**), the standard combo box (**DataCombo**), and the standard grid control (**DataGrid**). We will look at these later.

Data Bound Control Properties

- To establish the connection of the data bound control to a database, we use a few properties:

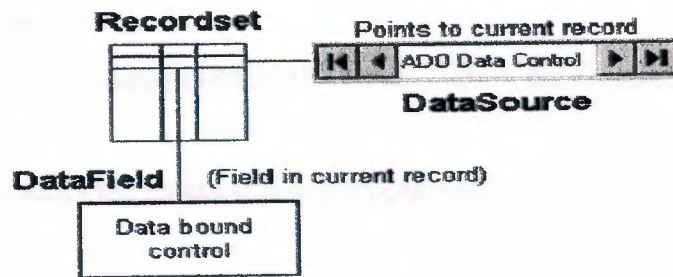
DataChanged	Indicates whether a value displayed in a bound control has changed.
DataField	Specifies the name of a field in the table pointed to by the respective data control.
DataSource	Specifies which data control the control is bound to (indirectly specifying the database table).

- If the data in any data bound control is changed and the user moves to another record in the database, the database will **automatically** be **updated** with the new data (assuming it is not **ReadOnly**). Be aware of this - it is an extremely powerful feature of the data control, but also a potential source of problems.
- To make using bound controls easy, follow these steps (in order listed) in placing the controls on a form:

1. Draw the bound control on the same form as the data control to which it will be bound.
2. Set the **DataSource** property. Click on the drop-down arrow to list the data controls on your form. Choose one.
3. Set the **DataField** property. Click on the drop-down arrow to list the fields associated with the selected data control records. Make your choice.
4. Set all other properties, as needed.

Again, by following these steps in order, we avoid potential data access errors.

- The relationships between a data bound control (**DataSource** and **DataField** properties) and the ADO data control (**Recordset** property) are:



ADO Data Control Events

- Like other controls, the ADO data control has **events** that are triggered at various times during database access. In these events, we write BASIC code to perform specific needed tasks. In this chapter, we will not be using these event procedures, but we will define them to make our definition of the data control complete.

Important ADO data control events:

WillMove

Triggers before record pointer is moved from one row to another.

MoveComplete

Event triggered after record pointer has been moved from one row to another.

EndofRecordset

Triggers when the record pointer is moved past the last record in the recordset.

WillChangeRecordset

Triggers before a change is made to the recordset. Used to trap unwanted changes.

RecordsetChangeComplete

Triggers after a change is made to recordset.

WillChangeRecord

Triggers before updates for the current row are sent to the data source.

RecordChangeComplete

Triggers after updates for the current row are sent to the data source.

WillChangeField

Triggers before the current field in the recordset is updated.

FieldChangeComplete

Triggers after the current field in the recordset has been updated.

- These events will be discussed further when we begin development of database management techniques in a later chapter.

ADO Data Control Method

- To complete our definition of the ADO data control, we present a single method:

Refresh Requeries the database based on contents of the **RecordSource** property.

- Like events, this ADO data control method will be discussed further when we begin development of database management techniques in a later chapter.

ADO Data Control Recordset Properties

- The **Recordset** object of the data control has its own set of **properties**. These properties can only be accessed at **run-time**. To refer to a Recordset property, use a 'double-dot' notation. For example, if you have a data control named **datExample**, to refer to a property named **PropertyName**, the notation is:

datExample.Recordset.PropertyName

- Important data control Recordset properties are:

AbsolutePosition	Long integer that either gets or sets the position of the current record.
BOF	Returns True when the current record is positioned before any data.
Bookmark	Sets or returns a bookmark to the current record. Used as a place marker in database management tasks.
EditMode	Indicates the state of editing for the current record.
EOF	Returns True when the current record is positioned past any data.
RecordCount	The total number of records in the Recordset.

- We will look at the BOF and EOF properties in the section on Recordset Navigation. Other properties will be examined later in this course.

ADO Data Control Recordset Methods

- The data control **Recordset** also has its own set of **methods** that perform functions on the Recordset. These methods are invoked using the double-dot notation introduced for the Recordset properties. So, for a data control (**datExample**) and method (**MethodName**), you invoke the method via:

datExample.Recordset.MethodName

- Important Recordset methods are:

AddNew	Adds a new record to the Recordset. All fields are set to null and this record becomes the current record.
CancelUpdate	Used to cancel any pending updates (either while editing or using the AddNew method)
Close	Closes a Recordset.
Delete	The current record is deleted from the Recordset.
Move	Moves the current record pointer a specified number of records forward or backward.
MoveFirst	Moves the current record pointer to the first record in the Recordset.
MoveLast	Moves the current record pointer to the last record in the Recordset.
MoveNext	Moves the current record pointer to the next record in the Recordset.
MovePrevious	Moves the current record pointer to the previous record in the Recordset.
Requery	Updates the data in a Recordset object by re-executing the query on which the object is based.
Update	Saves the current contents of all data bound controls.

- We will look at the four 'Move' methods in the next section on Recordset Navigation. Other properties will be reviewed later in this course.

ADO Data Control Recordset Navigation

- We have seen that, on the form, the ADO data control has four arrows that allow the user to move to the first, next, previous, and last records in the Recordset. Unfortunately, this control does not have a familiar look to a user and it may not be clear just exactly what functions the arrows perform. For this reason, we usually set the data control's **Visible** property to **False** and provide a programmatic approach to moving from record to record, or **Recordset navigation**.
- Four Recordset methods replicate the capabilities of the arrow buttons on the data control: **MoveFirst**, **MoveNext**, **MovePrevious**, and **MoveLast**. For each function we need, a command button is added to the form, with a **Click** event procedure attached to the corresponding Recordset method.
- When programmatically navigating through the records, you need to be aware of the position of the current record. For example, if you are at the first record and try a **MovePrevious** method, you will move past the beginning of the file. You can use the **BOF** property to see you are at the beginning of file and disallow such a move. You need a similar check at the end of a file to disallow an invalid **MoveNext** method.

DAO or ADO – What's the Difference?

- Use of the two controls is nearly identical – you can see that in the fact that the two sets of notes are nearly identical. The primary difference between the two data controls is in the properties used to connect to a database. The **DAO** control connects to the database using the **DatabaseName** property while the **ADO** control uses the **ConnectionString**.
- To construct a Recordset, the **DAO** control uses the **RecordSource** property. The **ADO** control requires setting two properties: **RecordSource** and **CommandType**.
- The **ADO** data control offers more event procedures than the **DAO** control to allow more complete control over database management.
- So, the question still may be – which control should you use? You should understand the use of both data controls because you will see them both as you progress as a Visual Basic programmer. For simple projects, the **DAO** control is adequate. For more detailed projects and for all new projects, we would recommend

the ADO data control. It is new technology and will receive the bulk of Microsoft's support with future releases of Visual Basic. And, as seen in the next section, it has some hidden powers we have yet to see!

Command Button Control



- The **command button** is probably the most widely used control. It is used to begin, interrupt, or end a particular process. With **databases**, it is used to **navigate** among records, **add** records, and **delete** records.

Label Control



- A **label** is a control you use to display text. The text in a label can be changed at run-time in response to events. It is widely used in **database** applications for **information display**.

Text Box Control



A **text box** is used to display information entered at design time, by a user at run-time, or assigned within code. The displayed text may be edited. This is the tool used in **database** applications for **editing** fields

Check Box Control



Check boxes provide a way to make choices from a list of potential candidates. Some, all, or none of the choices in a group may be selected. With **databases**, check boxes are used for many kinds of **choices**

Option Button Control



- **Option buttons** provide the capability to make a mutually exclusive choice among a group of potential candidate choices. Hence, option buttons work as a group, only one of which can have a True (or selected) value. Option buttons on a form work as an independent group as do groups of options buttons within frames. Option buttons are not data bound controls, yet they can still be used for a variety of **options** in database interfaces.

Frame Control



- **Frames** provide a way of grouping related controls on a form. Option buttons within a frame act independently of other option buttons in an application.

STUDENT DATABASE

FIELD NAME	TYPE	SIZE
Stdnumber	NUMERIC	--
Stdname	CHARACTER	15
Stdsurname	CHARACTER	15
Department	CHARACTER	25
Mothername	CHARACTER	15
Fathername	CHARACTER	15
Birthday	DATE	--
Birthplace	DATE	--
Tel1	NUMERIC	--
Tel2	NUMERIC	--
Recorddate	DATE	--
Roomno	NUMERIC	--
Situationboolean	BOOLEAN	--
Situation	CHARACTER	5
Penaltydate	DATE	--
Reasonpenalty	CHARACTER	75
Givepenalty	BOOLEAN	--
Permitboolean	BOOLEAN	--
Permissiondate	DATE	--
Address	CHARACTER	40

VISITOR DATABASE

Name	CHARACTER	15
Surname	CHARACTER	15
Licenceno	CHARACTER	10
Whovis	CHARACTER	15
Visitedroom	NUMERIC	--
Vistime	TIME	--

DORMITORY CONTROL SYSTEM

The most common problem for the school dormitory's today is security. This problem occurs due to the control of person who enters and leaves. In this program the control of students who enters and leaves and also the control of visitors is automated.

The student that is going to leave the dormitory will give his or her number, to the user. The user inputs the number to the corresponding space and search process is done. If student is recorded he or she can enter or leave the dormitory and corresponding checkbox is clicked. By that way the process is recorded to the database.

Addition to process of entrances and leaving the personal informations of the students can be recorded, updated or deleted, if student will be permitted the permission can be recorded and a list of permitted students can be seen, if a student will be given a punishment the punishment can be recorded and a list of punished students can be seen, the students who are inside or outside can be seen or the control visitors can be manipulated.

As a result this program automates the dormitory control and increases the security of a dormitory by using records and database which gives more information about visitors and students to the user. If the student can be given security cards as their identification card and the control can only be done with a program the security will be more successful.

MAIN FORM

The body of program starts with main form. You can access to other forms by using main form. The main form consists of main menu, toolbar, and searching part for entering-leaving of students to the dormitory.

In searching part you can write entering or leaving of student number to textbox then you press search button, if there is student number in database, it is shown in textboxes. Using checkbox, if it is on clicked checks entering or leaving of student, student is inside else student is outside. In toolbar there are some icons to access forms, which is needed. Main menu includes eight parts. These are file, list and search, control, permit, penalty, student control, visitor and exit. Shown in figure 1.

The screenshot shows a Windows-style application window titled "Near East University Dormitory". The window has a blue title bar with the date and time "22.12.2001 18:33:41". Below the title bar is a menu bar with the following items: "File", "Lists And Search", "Control", "Permit", "Penalty", "Student Control", "Visitor", and "Exit". Below the menu bar is a toolbar containing various icons for file operations and student management. The main area of the window is divided into several sections. At the top, there is a "Student Number" label followed by a text box containing "971387" and a "Search" button. Below this is a section titled "Entering / Leaving" which contains a table with four rows: "Number", "Name", "Surname", and "Room No". The values in these rows are "921213", "AYDIN", "TAŞTAN", and "10" respectively. To the right of this table is a checkbox labeled "Situation" which is checked, and a text box below it containing the word "Inside". At the bottom of the window, there is a large text label "Near East University Dormitory" and a small logo. In the bottom right corner, there is a label "Total Student Number" followed by a text box containing the number "3".

Entering / Leaving	
Number	921213
Name	AYDIN
Surname	TAŞTAN
Room No	10

**Near East University
Dormitory**

Total Student Number 3

Figure 1.

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub exit1_Click()
```

```
End
```

```
End Sub
```

```
Private Sub listeleme_Click()
```

```
Load Form1
```

```
Form1.Show
```

```
Unload main
```

```
main.Hide
```

```
Form1.Text2.SetFocus
```

```
Form1.Text2.Text = ""
```

```
End Sub
```

```
Private Sub ListsofPenalty_Click()
```

```
Data1.Refresh
```

```
Load punishedlist
```

```
punishedlist.Show
```

```
main.Hide
```

```
Unload main
```

```
End Sub
```

```
Private Sub Check1_Click()
```

```
Data1.Recordset.Edit
```

```
    If Check1.Value = Checked Then
```

```
        Text5.Text = "Inside"
```

```
        Data1.Recordset.situation = "Inside"
```

```
    ElseIf Check1.Value = Unchecked Then
```

```
        Text5.Text = "Outside"
```

```
        Data1.Recordset.situation = "Outside"
```

```
    End If
```

```
End Sub
```

```
Private Sub exit2_Click()
```

```
End
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
Dim a, b
```

```
If Data1.Recordset.BOF Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
    If Text1.Text = "" Then
```

```
        MsgBox "Enter student number!"
```

```
        Text1.SetFocus
```

```
    Else
```

```
        notgo = Data1.Recordset.Bookmark
```

```
        stdnumber$ = Text1.Text
```

```
        bul = "stdnumber =" & stdnumber$ & ""
```

```
        Data1.Recordset.FindFirst bul
```

```
        a = Val(Text1.Text)
```

```
        b = Val(Text9.Text)
```

```
        If a <> b Then
```

```
            MsgBox "This student number is not found"
```

```
            Text1.SelStart = 0
```

```
            Text1.SelLength = Len(Text1.Text)
```

```
            Text1.SetFocus
```

```
        Data1.Recordset.Bookmark = notgo
```

```
    Else:
```

```
        Text1.SelStart = 0
```

```
        Text1.SelLength = Len(Text1.Text)
```

```
        Text1.SetFocus
```

```
    If Data1.Recordset.situation = "Inside" Then
```

```
        Check1.Value = Checked
```

```
    ElseIf (Data1.Recordset.situation = "Outside") Or (Data1.Recordset.situation = "")
```

```
Then
```

```
    Check1.Value = Unchecked
```

End If

End If

End If

End If

End Sub

Private Sub Data1_Reposition()

Text4.Text = Str(Data1.Recordset.RecordCount)

End Sub

Private Sub Form_Load()

Call Toolbar1.RestoreToolbar("x", "x", "Toolbar1")

Data1.Refresh

main.Caption = Time() + Date

If Data1.Recordset.BOF Then

Else

Data1.Recordset.MoveLast

If Data1.Recordset.situation = "Inside" Then

Check1.Value = Checked

ElseIf (Data1.Recordset.situation = "Outside") Or (Data1.Recordset.situation = "") Then

Check1.Value = Unchecked

End If

Text5.Locked = True

Text6.Locked = True

Text7.Locked = True

Text8.Locked = True

Text9.Locked = True

Text5.FontBold = True

Text5.FontItalic = True

Text6.FontBold = True

Text6.FontItalic = True

Text7.FontBold = True

Text7.FontItalic = True

Text8.FontBold = True

Text8.FontItalic = True

Text9.FontBold = True

Text9.FontItalic = True

End If

End Sub

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

Label5.ForeColor = RGB(0, 0, 0)

End Sub

Private Sub Frame1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

Label5.ForeColor = RGB(0, 0, 0)

End Sub

Private Sub PermissionSituation_Click()

Load permitform

permitform.Show

permitform.Text6.SetFocus

main.Hide

Unload main

End Sub

Private Sub Listonleave_Click()

Data1.Refresh

Load onleave

onleave.Show

main.Hide

Unload main

End Sub

Private Sub Label5_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

Label5.ForeColor = RGB(0, 0, 255)

End Sub

Private Sub SearchbyRoomNumber_Click()

Data1.Refresh

Load byroomnumber

byroomnumber.Show

main.Hide

Unload main

byroomnumber.Text5.SetFocus

End Sub

Private Sub Checkstudentsandrooms_Click()

Data1.Refresh

Load floorcontrol

floorcontrol.Show

main.Hide

Unload main

End Sub

Private Sub StudentControl_Click()

inside.datPrimaryRS.Refresh

Load inside

inside.Show

Unload main

main.Hide

End Sub

Private Sub Toolbar1_ButtonClick(ByVal Button As MSComctlLib.Button)

Select Case Button.Key

Case "new": addnew.Show

main.Hide

Case "update": updateform.Show

main.Hide

```

Case "delete": deleteform.Show
main.Hide
Case "listandsearch": Form1.Show
main.Hide
Case "checkstudentandroom": floorcontrol.Show
main.Hide
Case "searchbroomnumber": byroomnumber.Show
main.Hide
Case "permit": permitform.Show
main.Hide
Case "listofpermit": onleave.Show
main.Hide
Case "penalty": penaltyform.Show
main.Hide
Case "listofpenalty": punishedlist.Show
main.Hide
Case "inside": inside.Show
main.Hide
Case "visitor": visitorform.Show
main.Hide
End Select
End Sub

```

```

Private Sub Toolbar1_Change()
Call Toolbar1.SaveToolbar("x", "x", "Toolbar1")
End Sub

```

```

Private Sub update_Click()
Load updateform
updateform.Show
main.Hide
Unload main
End Sub

```



```
Private Sub delete_Click()
```

```
Data1.Refresh
```

```
deleteform.Data1.Refresh
```

```
Load deleteform
```

```
deleteform.Show
```

```
main.Hide
```

```
Unload main
```

```
deleteform.Text13.Text = ""
```

```
deleteform.Text13.SetFocus
```

```
End Sub
```

```
Private Sub Text1_KeyPress(KeyAscii As Integer)
```

```
If KeyAscii = vbKeyReturn Then
```

```
Command1.Value = True
```

```
End If
```

```
If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Or  
KeyAscii = vbKeyReturn Then
```

```
Exit Sub
```

```
Else
```

```
KeyAscii = 0
```

```
End If
```

```
End Sub
```

```
Private Sub Timer1_Timer()
```

```
main.Caption = Time() + Date
```

```
End Sub
```

```
Private Sub OperationoffPenalty_Click()
```

```
Load penaltyform
```

```
penaltyform.Show
```

```
main.Hide
```

```
Unload main
```

```
End Sub
```

```

Private Sub newrecord_Click()
Data1.Refresh
addnew.Data1.Refresh
Load addnew
addnew.Show
main.Hide
Unload main
addnew.Command1.Default = True
End Sub

```

```

Private Sub visitor_Click()
visitorform.Show
main.Hide
End Sub

```

FILE

The file includes five parts, which are New Record, Update, Delete, Search and Exit.

NEW RECORD

In new record part, the user enters the information of student. There are four buttons on the form. Their names are Enter Data, Main Menu, Save and Cancel. If the user wants to write student information user presses Enter Data Button then

Student information is written to textboxes. Then if user wants to save this record, he or she presses Save button then program gives a message which is "This record will save are you sure". If "Yes" is pressed, record is written to database, if "No" pressed, all textboxes are cleaned or if Cancel button is pressed, record is not saved. If any of textbox is empty, a message will be given. Shown in figure 2

Number	921213	Phone 1	03923548522	Phone 2	05487985401
Name	AYDIN	Record Date	06/01/2000	Address	
Surname	TAŞTAN	Room Number	10		
Department	Computer	Situation	Inside		
Birth day	10/06/1972	Father Name	MUSTAFA		
Birthplace	ANKARA	Mother Name	AYŞE		

Buttons: Enter Data, Main menu, Save, Cancel

Figure 2

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Combo1_KeyPress(KeyAscii As Integer)
```

```
Dim NextIndex As Integer
```

```
If KeyAscii = vbKeyReturn Then
```

```
NextIndex = Index + 1
```

```
If NextIndex = 3 Then
```

```
Else
```

```
Text1(8).SetFocus
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub Combo2_KeyPress(KeyAscii As Integer)
```

```
Dim NextIndex As Integer
```

```
If KeyAscii = vbKeyReturn Then
```

```
NextIndex = Index + 1
```

```
If NextIndex = 3 Then
```

```
Else
```

```
MaskedTextBox1.SetFocus
```

```
End If
```

```
End If
```

```
End Sub
```



```

Private Sub Command1_Click()
Data1.DatabaseName = "d:\YURT\yurtbase.mdb"
Text1(0).SetFocus
Data1.Refresh

MaskedTextBox1.Visible = True
Text1(3).Visible = False
MaskedTextBox2.Visible = True
Text1(6).Visible = False

Combo1.Visible = True
Combo2.Visible = True

Text2.Visible = False
Text3.Visible = False

Data1.Recordset.addnew
For i = 0 To 11
Text1(i).Text = ""
Text1(i).Locked = False
Next
Combo1.Locked = False
Combo2.Locked = False
Command1.Enabled = False
Command4.Enabled = False
Command2.Enabled = True
Command3.Enabled = True
Text1(0).Text = "."
Text1(0).Text = ""
Text1(1).Text = "."
Text1(1).Text = ""
Text1(2).Text = "."
Text1(2).Text = ""
Text1(7).Text = "."

```

```

Text1(7).Text = ""
Combo1.Text = "Inside"
Combo2.Text = "Architecture"
End Sub

Private Sub Command2_Click()
If Text1(0).Text = "" Then
    MsgBox "You must enter student number."
    Text1(0).SetFocus
ElseIf Text1(1).Text = "" Then
    MsgBox "You must enter student name."
    Text1(1).SetFocus
ElseIf Text1(2).Text = "" Then
    MsgBox "You must enter student surname ."
    Text1(2).SetFocus
ElseIf Text1(7).Text = "" Then
    MsgBox "You must enter room number."
    Text1(7).SetFocus
ElseIf Combo1.Text = "" Then
    MsgBox "You must enter student situation."
    Combo1.SetFocus
ElseIf Text1(7).Text > 45 Then
    MsgBox ("Sorry, available room is 45")
    Text1(7).SelStart = 0
    Text1(7).SelLength = Len(Text1(7).Text)
    Text1(7).SetFocus
Else

Text1(3).Text = MaskedTextBox1.Text
Text1(6).Text = MaskedTextBox2.Text

Text2.Text = Combo2.Text
Text3.Text = Combo1.Text
If Text3.Text = "Inside" Then

```

```

        Check1.Value = Checked
    ElseIf Text3.Text = "" Then
        Check1.Value = Unchecked
    End If

    cevap = MsgBox("This record will save are you sure?", 4)
    If cevap = 6 Then
        MaskedTextBox1.Visible = False
        Text1(3).Visible = True
        MaskedTextBox2.Visible = False
        Text1(6).Visible = True

        Combo1.Visible = False
        Combo2.Visible = False
        Text2.Visible = True
        Text3.Visible = True

        Data1.Recordset.update
        Data1.Recordset.MoveLast

        MaskedTextBox1.Text = "__/__/____"
        MaskedTextBox2.Text = "__/__/____"
    Else
        Data1.Recordset.CancelUpdate
        MaskedTextBox1.Visible = False
        Text1(3).Visible = True
        MaskedTextBox2.Visible = False
        Text1(6).Visible = True

        Combo1.Visible = False
        Combo2.Visible = False
        Text2.Visible = True
        Text3.Visible = True
        MaskedTextBox1.Text = "__/__/____"
    End If

```



```
MaskedTextBox2.Text = " __/__/__"
```

```
End If
```

```
For i = 0 To 11
```

```
Text1(i).Locked = True
```

```
Next
```

```
Combo1.Locked = True
```

```
Combo2.Locked = True
```

```
Command1.Enabled = True
```

```
Command4.Enabled = True
```

```
Command2.Enabled = False
```

```
Command3.Enabled = False
```

```
Command1.Default = True
```

```
End If
```

```
End Sub
```

```
Private Sub Command3_Click()
```

```
Data1.Recordset.CancelUpdate
```

```
MaskedTextBox1.Text = " __/__/__"
```

```
MaskedTextBox2.Text = " __/__/__"
```

```
MaskedTextBox1.Visible = False
```

```
Text1(3).Visible = True
```

```
MaskedTextBox2.Visible = False
```

```
Text1(6).Visible = True
```

```
Combo1.Visible = False
```

```
Combo2.Visible = False
```

```
Text2.Visible = True
```

```
Text3.Visible = True
```

```
Command2.Enabled = False
```

```
Command3.Enabled = False
```

```
Command1.Enabled = True
```

```
Command4.Enabled = True
```

```
For i = 0 To 11
Text1(i).Locked = True
Next
Combo1.Locked = True
Combo2.Locked = True
End Sub
```

```
Private Sub Command4_Click()
Unload addnew
addnew.Hide
Data1.Refresh
Load main
main.Show
End Sub
```

```
Private Sub Command5_Click()
floor1.Command1.Visible = False
floor1.Command2.Visible = False
floor1.Command3.Visible = False
floor1.Show
End Sub
```

```
Private Sub Form_Load()
Data1.DatabaseName = ""
Data1.Refresh
Combo1.AddItem "Inside"
Combo1.AddItem "Outside"
Combo2.AddItem "Business"
Combo2.AddItem "Economy"
Combo2.AddItem "International"
Combo2.AddItem "CIS"
Combo2.AddItem "Banking and Finance"
Combo2.AddItem "Electrical-Electronic"
Combo2.AddItem "Computer"
```

```
Combo2.AddItem "Civil"  
Combo2.AddItem "Machanical"  
Combo2.AddItem "Architecture"  
Combo2.AddItem "Marine"  
Combo2.AddItem "Law"  
MaskedTextBox1.Visible = False  
MaskedTextBox2.Visible = False  
Combo1.Visible = False  
Combo2.Visible = False
```

```
For i = 0 To 11  
Text1(i).Locked = True  
Combo1.Locked = True  
Combo2.Locked = True  
Next  
Command2.Enabled = False  
Command3.Enabled = False  
Command1.Enabled = True  
Command4.Enabled = True  
Text1(3).MaxLength = 10  
Text1(6).MaxLength = 10  
End Sub
```

```
Private Sub MaskedTextBox1_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 Then  
Text1(4).SetFocus  
End If  
End Sub
```

```
Private Sub MaskedTextBox2_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 Then  
Text1(7).SetFocus  
End If  
End Sub
```



```

Private Sub Text1_KeyPress(Index As Integer, KeyAscii As Integer)
Dim NextIndex As Integer
If Command2.Enabled = True Then
    If KeyAscii = vbKeyReturn Then
        NextIndex = Index + 1
        If NextIndex = 3 Then
            Combo2.SetFocus
        ElseIf NextIndex = 4 Then
            MaskedTextBox1.SetFocus
        ElseIf NextIndex = 6 Then
            MaskedTextBox2.SetFocus
        ElseIf NextIndex = 8 Then
            Combo1.SetFocus
        ElseIf NextIndex = 12 Then
            Exit Sub
        Else
            Text1(NextIndex).SetFocus
        End If
    End If
End If

If (Index = 0) Or (Index = 5) Or (Index = 7) Or (Index = 10) Then
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then
        Exit Sub
    Else
        KeyAscii = 0
    End If
End If
End Sub

```

UPDATE

Update is used for updating the student information or making deletions of mistakes in student informations.

If student information is going to change, firstly that student's number is written to textbox and Enter or Search Buttons is pressed for looking at student informations in textboxes. If database doesn't consist of this student number, a message is given which is "This student number is not found". If it consists of this student number, a message is given which is "Are you sure to change this record", If the user presses to "Yes" button, student informations is changed. If the user presses to "No" button student information will not change. If the user wants to save this changings, than presses to Save button. The Modify button, which is taken up over form that is used to give permission to information changings. However there are five buttons in the form: First, Previous, Next, Last and Main menu. Shown in figure 3

The screenshot shows a Windows-style application window titled "Update". It contains a grid of text boxes and dropdown menus for entering student data. The data entered includes: Number: 971387, Room Number: 1, Name: ERHAN, Situation: Inside, Surname: OZTURK, Father Name: HUSEYIN, Department: Computer, Mother Name: AYTEN, Birth day: 10/02/1990, Phone 2: 05425845003, Birthplace: AKSARAY, Address: KALE CADDE ABAYLAR AP. NO/1 KAT/1 AKSARAY, Phone 1: 03822136099, and Record Date: 10/10/2001. To the right of the data fields is a section titled "Update operation" which includes a "Student number" input field, a "Search" button, and a vertical stack of buttons: "Modify", "Save", "Cancel", "First", "Previous", "Next", and "Last". At the bottom right of this section is a "Main menu" button.

Figure 3

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
```

```
Dim a, b
```

```
If Data1.Recordset.EOF Then
```

```
    MsgBox ("There is no record")
```

```
Else
```

```
If Text13.Text = "" Then
```

```

MsgBox "Enter student number"
Text13.SetFocus
Else
notgo = Data1.Recordset.Bookmark
stdnumber$ = Text13.Text
bul = "stdnumber='" & stdnumber$ & "'"
Data1.Recordset.FindFirst bul
a = Val(Text13.Text)
b = Val(Text1(0).Text)
If a <> b Then
MsgBox "This student number is not found"
Text13.SelStart = 0
Text13.SelLength = Len(Text13.Text)
Text13.SetFocus
Data1.Recordset.Bookmark = notgo
Else:
Text13.SelStart = 0
Text13.SelLength = Len(Text13.Text)
Text13.SetFocus
cevap = MsgBox("Are you sure to change this record", vbYesNo)
If cevap = 6 Then
Data1.Recordset.Edit
Command1.Enabled = False
Command4.Enabled = False
Command5.Enabled = False
Command2.Enabled = True
Command3.Enabled = True
For i = 0 To 11
Text1(i).Enabled = True
Combo1.Enabled = True
Combo2.Enabled = True
Text1(i).FontBold = False
Text1(i).FontItalic = False
Combo1.FontBold = False

```



```

Combo1.FontItalic = False
Combo2.FontBold = False
Combo2.FontItalic = False
Next
Text1(0).SetFocus
Else: Text13.SelStart = 0
    Text13.SelLength = Len(Text13.Text)
    Text13.SetFocus
End If
End If
End If
End If
End Sub

```

```

Private Sub Command2_Click()
If Text1(0).Text = "" Then
    MsgBox "Enter student number"
    Text1(0).SetFocus
ElseIf Text1(1).Text = "" Then
    MsgBox "Enter student nane"
    Text1(1).SetFocus
ElseIf Text1(2).Text = "" Then
    MsgBox "Enter student surname"
    Text1(2).SetFocus
ElseIf Text1(3).Text = "" Then
    MsgBox "Enter student birth day"
    Text1(3).SetFocus
ElseIf Text1(4).Text = "" Then
    MsgBox "Enter student birthplace"
    Text1(4).SetFocus
ElseIf Text1(5).Text = "" Then
    MsgBox "Enter student phone"
    Text1(5).SetFocus

```

```

ElseIf Text1(6).Text = "" Then
    MsgBox "Enter student record date"
    Text1(6).SetFocus
ElseIf Text1(7).Text = "" Then
    MsgBox "Enter student room number"
    Text1(7).SetFocus
ElseIf Text1(8).Text = "" Then
    MsgBox "Enter student father name"
    Text1(8).SetFocus
ElseIf Text1(9).Text = "" Then
    MsgBox "Enter student mother name"
    Text1(9).SetFocus
ElseIf Text1(10).Text = "" Then
    MsgBox "Enter student phone"
    Text1(10).SetFocus
ElseIf Text1(11).Text = "" Then
    MsgBox "Enter student address"
    Text1(11).SetFocus
ElseIf Combo1.Text = "" Then
    MsgBox "Enter student situation"
    Combo1.SetFocus
ElseIf Combo2.Text = "" Then
    MsgBox "Enter Student department"
    Combo2.SetFocus
Else
    cevap = MsgBox("This record will be saved?", 4)

    If cevap = 6 Then
        Data1.Recordset.update
        Frame3.Enabled = True
        Command1.Enabled = True
        Command4.Enabled = True
        Command5.Enabled = True
        Command2.Enabled = False
    
```

```

Command3.Enabled = False
For i = 0 To 11
    Text1(i).FontBold = True
    Text1(i).FontItalic = True
    Combo1.FontBold = True
    Combo1.FontItalic = True
    Combo2.FontBold = True
    Combo2.FontItalic = True
    Text1(i).Enabled = False
    Combo1.Enabled = False
    Combo2.Enabled = False
Next
Text13.SelStart = 0
    Text13.SelLength = Len(Text13.Text)
    Text13.SetFocus
ElseIf cevap = 7 Then
    Data1.Recordset.CancelUpdate
    Frame3.Enabled = True
    Command1.Enabled = True
    Command4.Enabled = True
    Command5.Enabled = True
    Command2.Enabled = False
    Command3.Enabled = False
    For i = 0 To 11
        Text1(i).FontBold = True
        Text1(i).FontItalic = True
        Combo1.FontBold = True
        Combo1.FontItalic = True
        Combo2.FontBold = True
        Combo2.FontItalic = True
        Text1(i).Enabled = False
        Combo1.Enabled = False
        Combo2.Enabled = False
    Next

```



```
Text13.SelStart = 0
Text13.SelLength = Len(Text13.Text)
Text13.SetFocus
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub Command3_Click()
```

```
Data1.Recordset.CancelUpdate
```

```
Frame3.Enabled = True
```

```
Command1.Enabled = True
```

```
Command4.Enabled = True
```

```
Command5.Enabled = True
```

```
Command2.Enabled = False
```

```
Command3.Enabled = False
```

```
For i = 0 To 11
```

```
Text1(i).Enabled = False
```

```
Combo1.Enabled = False
```

```
Combo2.Enabled = False
```

```
Text1(i).FontBold = True
```

```
Text1(i).FontItalic = True
```

```
Combo1.FontBold = True
```

```
Combo1.FontItalic = True
```

```
Combo2.FontBold = True
```

```
Combo2.FontItalic = True
```

```
Next
```

```
Text13.SelStart = 0
```

```
Text13.SelLength = Len(Text13.Text)
```

```
Text13.SetFocus
```

```
End Sub
```

```
Private Sub Command4_Click()
```

```
Unload updateform
```

```
updateform.Hide
```

```
Load main
```

```
main.Show
```

```
End Sub
```

```
Private Sub Command5_Click()
```

```
If Data1.Recordset.EOF Then
```

```
MsgBox ("There is no any record")
```

```
Else
```

```
Data1.Recordset.Edit
```

```
Frame3.Enabled = False
```

```
Command1.Enabled = False
```

```
Command4.Enabled = False
```

```
Command5.Enabled = False
```

```
Command2.Enabled = True
```

```
Command3.Enabled = True
```

```
For i = 0 To 11
```

```
Text1(i).Enabled = True
```

```
Combo1.Enabled = True
```

```
Combo2.Enabled = True
```

```
Text1(i).FontBold = False
```

```
Text1(i).FontItalic = False
```

```
Combo1.FontBold = False
```

```
Combo1.FontItalic = False
```

```
Combo2.FontBold = False
```

```
Combo2.FontItalic = False
```

```
Next
```

```
Text1(0).SetFocus
```

```
End If
```

```
End Sub
```



```
Private Sub Command6_Click()
```

```
If Data1.Recordset.BOF Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
Data1.Recordset.MoveLast
```

```
End If
```

```
End Sub
```

```
Private Sub Command7_Click()
```

```
If Data1.Recordset.RecordCount = 0 Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
If Data1.Recordset.EOF Then
```

```
    MsgBox ("End of file encountered")
```

```
    Data1.Recordset.MoveLast
```

```
Else
```

```
Data1.Recordset.MoveNext
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub Command8_Click()
```

```
If Data1.Recordset.RecordCount = 0 Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
If (Data1.Recordset.BOF) Then
```

```
    MsgBox ("Beginning of file encountered")
```

```
    Data1.Recordset.MoveFirst
```

```
Else
```

```
Data1.Recordset.MovePrevious
```

```
End If
```

```
End If
```

```
End Sub
```



```

Private Sub Command9_Click()
If Data1.Recordset.EOF Then
    MsgBox ("There is no any record")
Else
Data1.Recordset.MoveFirst
End If
End Sub
Private Sub Form_Activate()
Data1.Refresh
End Sub

```

```

Private Sub Form_Load()
Data1.Refresh
Combo1.AddItem "Inside"
Combo1.AddItem "Outside"
Combo2.AddItem "Business"
Combo2.AddItem "Economy"
Combo2.AddItem "International"
Combo2.AddItem "CIS"
Combo2.AddItem "Banking and Finance"
Combo2.AddItem "Electrical-Electronic"
Combo2.AddItem "Computer"
Combo2.AddItem "Civil"
Combo2.AddItem "Mechanical"
Combo2.AddItem "Architecture"
Combo2.AddItem "Marine"
Combo2.AddItem "Law"
For i = 0 To 11
Text1(i).FontBold = True
Text1(i).FontItalic = True
Text1(i).Enabled = False
Combo1.Enabled = False
Combo2.Enabled = False
Next

```

```
Combo1.FontBold = True
Combo2.FontBold = True
Combo1.FontItalic = True
Combo2.FontItalic = True
Command2.Enabled = False
Command3.Enabled = False
Command1.Enabled = True
Command4.Enabled = True
Command5.Enabled = True
End Sub
```

```
Private Sub Combo1_KeyPress(KeyAscii As Integer)
Dim NextIndex As Integer
If KeyAscii = vbKeyReturn Then
    NextIndex = Index + 1
    If NextIndex = 3 Then
        Else
            Text1(8).SetFocus
        End If
    End If
End Sub
```

```
Private Sub Combo2_KeyPress(KeyAscii As Integer)
Dim NextIndex As Integer
If KeyAscii = vbKeyReturn Then
    NextIndex = Index + 1
    If NextIndex = 3 Then
        Else
            Text1(3).SetFocus
        End If
    End If
End Sub
```

```

Private Sub Text1_KeyPress(Index As Integer, KeyAscii As Integer)
Dim NextIndex As Integer
If Command2.Enabled = True Then
    If KeyAscii = vbKeyReturn Then
        NextIndex = Index + 1
        If NextIndex = 3 Then
            Combo2.SetFocus
        ElseIf NextIndex = 8 Then
            Combo1.SetFocus
        ElseIf NextIndex = 12 Then
            Exit Sub
        Else
            Text1(NextIndex).SetFocus
        End If
    End If
End If

If (Index = 0) Or (Index = 5) Or (Index = 7) Or (Index = 10) Then
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then
        Exit Sub
    Else
        KeyAscii = 0
    End If
End If

If (Index = 3) Or (Index = 6) Then
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii = Asc("/")) Or
    KeyAscii = vbKeyBack Then
        Exit Sub
    Else
        KeyAscii = 0
    End If
End If
End Sub

```



```

Private Sub Text13_KeyPress(KeyAscii As Integer)
If KeyAscii = vbKeyReturn Then
Command1.Value = True
End If
If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then
Exit Sub
Else
KeyAscii = 0
End If
End Sub

```

DELETE

The student record is removed from database in this part. The user then that presses Search Button to search the database writes student number. If it is found all student informations are shown in the textboxes then Delete Button is pressed, it is asked to user "Are you sure to delete this record" If he presses to "Yes" information of this student is removed from database. If he or she presses "No", information is not removed. Shown in figure 4

The screenshot shows a 'Delete' dialog box with the following fields and controls:

Number	971387	Room Number	1
Name	ERHAN	Father Name	HUSEYIN
Surname	OZTURK	Mother Name	AYTEN
Department	Computer	Address	KALE CADDE ABAYLAR AP. NO/71 KAT/1 AKSARAY
Birth Date	10/02/1980		
Birthplace	AKSARAY		
Phone:1	03622136099	Phone:2	05425846003
Record Date	10/10/2001		

Deletion operation section:

- Student Number: []
- Buttons: First, Previous, Next, Last
- Buttons: Search, Delete
- Button: Main menu

Figure 4

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()  
Dim a, b  
If Data1.Recordset.EOF Then  
    MsgBox ("There is no any record")  
Else  
If Text13.Text = "" Then  
    MsgBox "Enter student number for searching!"  
    Text13.SetFocus  
Else  
notgo = Data1.Recordset.Bookmark  
stdnumber$ = Text13.Text  
bul = "stdnumber =" & stdnumber$ & ""  
Data1.Recordset.FindFirst bul  
a = Val(Text13.Text)  
b = Val(Text1.Text)  
If a < b Then  
    MsgBox "This student number is not found"  
    Text13.SelStart = 0  
    Text13.SelLength = Len(Text13.Text)  
    Text13.SetFocus  
    Data1.Recordset.Bookmark = notgo  
Else:  
    Text13.SelStart = 0  
    Text13.SelLength = Len(Text13.Text)  
    Text13.SetFocus  
  
End If  
End If  
End If  
End Sub
```

```

Private Sub Command2_Click()
If Data1.Recordset.BOF Then
    MsgBox ("There is no record to delete")
Else
    seçiminiz = MsgBox("Are you sure to delete this record ?", 36, "Warning")
If seçiminiz = 6 Then
    If Data1.Recordset.EOF Then
        MsgBox "There is no record to delete"
    Else
        Data1.Recordset.delete
        Data1.Recordset.MovePrevious
        Text13.Text = ""
        Text13.SetFocus
        Data1.Refresh
    End If
End If
End If
End Sub

```

```

Private Sub Command3_Click()
If Data1.Recordset.RecordCount = 0 Then
    MsgBox ("There is no any record")
Else
    If Data1.Recordset.EOF Then
        MsgBox ("End of file encountered")
        Data1.Recordset.MoveLast
    Else
        Data1.Recordset.MoveNext
    End If
End If
End Sub

```



```
Private Sub Command4_Click()
```

```
Unload deleteform
```

```
deleteform.Hide
```

```
Data1.Refresh
```

```
Load main
```

```
main.Show
```

```
End Sub
```

```
Private Sub Command5_Click()
```

```
If Data1.Recordset.RecordCount = 0 Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
If (Data1.Recordset.BOF) Then
```

```
    MsgBox ("Beginning of file encountered")
```

```
    Data1.Recordset.MoveFirst
```

```
Else
```

```
Data1.Recordset.MovePrevious
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub Command6_Click()
```

```
If Data1.Recordset.BOF Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
Data1.Recordset.MoveLast
```

```
End If
```

```
End Sub
```

```
Private Sub Command7_Click()
```

```
If Data1.Recordset.BOF Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

Data1.Recordset.MoveFirst

End If

End Sub

Private Sub Form_Activate()

Data1.Refresh

End Sub

Private Sub Form_Load()

Data1.Refresh

Text1.Locked = True

Text2.Locked = True

Text3.Locked = True

Text4.Locked = True

Text5.Locked = True

Text6.Locked = True

Text7.Locked = True

Text8.Locked = True

Text9.Locked = True

Text10.Locked = True

Text11.Locked = True

Text12.Locked = True

Text14.Locked = True

End Sub

Private Sub Text13_KeyPress(KeyAscii As Integer)

If KeyAscii = vbKeyReturn Then

Command1.Value = True

End If

If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then

Exit Sub

Else

KeyAscii = 0

End If

End Sub

LIST AND SEARCH

This form consists of grid that lists all of the student informations, textboxes those showing student informations, which is choosen from grid. Searching is done by five different ways: according to number, name, surmane, room number and department. The field, which is searched, is choosed from option buttons and key word is written to textbox. In this way searching is done character by character. If the user chooses option button, which is department, is choosed from combobox to search and list. In this part, SQL is used. Shown in figure 5

Number	Name	Surname	Department	Recorl date	Room no	Situation	Father name	Mother name
971387	ERHAN	OZTURK	Computer	10/10/2001	1	Inside	HUSEYIN	AYTEN
970302	BULENT	DURUKAN	Computer	01/10/2000	7	Inside		
921213	AYDIN	TAŞTAN	Computer	06/01/2000	10	Inside	MUSTAFA	AYSE

Number	<input type="text" value="971387"/>	Birth day	<input type="text" value="10/02/1980"/>	Room Number	<input type="text" value="1"/>	Address: <input type="text" value="KALE CADDE ABAYLAR AP. NO:1 KAT:1 AKSARAY"/>
Name	<input type="text" value="ERHAN"/>	Birthplace	<input type="text" value="AKSARAY"/>	Father Name	<input type="text" value="HUSEYIN"/>	
Surname	<input type="text" value="OZTURK"/>	Phone 1	<input type="text" value="03822136099"/>	Mother Name	<input type="text" value="AYTEN"/>	
Department	<input type="text" value="Computer"/>	Record Date	<input type="text" value="10/10/2001"/>	Phone 2	<input type="text" value="06425846002"/>	

Search by

☒ Number ☐ Name ☐ Surname ☐ Room no ☐ Department

main menu

Figure 5

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Combo1_Change()
```

```
Dim kayitseti As Recordset
```

```
Dim db As Connection
```

```
Set db = New Connection
```

```
db.Open Adodc1.ConnectionString
```

```
Set kayitseti = New Recordset
```

```
kayitseti.Open Text1.Text + Combo1.Text + Text3.Text, db, adOpenStatic,
```

```
adLockOptimistic
```

```
Set Adodc1.Recordset = kayitseti
```

```
Set DataGrid1.DataSource = kayitseti
```

```
End Sub
```

```
Private Sub Combo1_Click()
```

```
Dim kayitseti As Recordset
```

```
Dim db As Connection
```

```
Set db = New Connection
```

```
db.Open Adodc1.ConnectionString
```

```
Set kayitseti = New Recordset
```

```
kayitseti.Open Text1.Text + Combo1.Text + Text3.Text, db, adOpenStatic,
```

```
adLockOptimistic
```

```
Set Adodc1.Recordset = kayitseti
```

```
Set DataGrid1.DataSource = kayitseti
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
Unload Form1
```

```
Form1.Hide
```

```
Load main
```

```
main.Show
```

```
End Sub
```

```

Private Sub Form_Activate()
Adodc1.Refresh
End Sub

Private Sub Form_Load()
Option1.Value = True
Combo1.AddItem "Business"
Combo1.AddItem "Economy"
Combo1.AddItem "International"
Combo1.AddItem "CIS"
Combo1.AddItem "Banking and Finance"
Combo1.AddItem "Electrical-Electronic"
Combo1.AddItem "Computer"
Combo1.AddItem "Civil"
Combo1.AddItem "Machanical"
Combo1.AddItem "Architecture"
Combo1.AddItem "Marine"
Combo1.AddItem "Law"
End Sub

```

```

Private Sub Option1_Click()
Text1.Text = "select
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mo
thername from yurt WHERE stdnumber like '%"
Text3.Text = "%'"
Text2.Text = ""
Combo1.Text = ""
End Sub

```

```

Private Sub Option2_Click()
Text2.SetFocus
Text1.Text = "select
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mo
thername from yurt WHERE stdname like '%"
Text3.Text = "%'"

```

```
Text2.Text = ""
```

```
Combo1.Text = ""
```

```
End Sub
```

```
Private Sub Option3_Click()
```

```
Text2.SetFocus
```

```
Text1.Text = "select
```

```
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE stdsurname like "%"
```

```
Text3.Text = "%"
```

```
Text2.Text = ""
```

```
Combo1.Text = ""
```

```
End Sub
```

```
Private Sub Option4_Click()
```

```
Text2.SetFocus
```

```
Text1.Text = "select
```

```
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE roomno like "%"
```

```
Text3.Text = "%"
```

```
Text2.Text = ""
```

```
Combo1.Text = ""
```

```
End Sub
```

```
Private Sub Option5_Click()
```

```
Text1.Text = "select
```

```
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE department like "%"
```

```
Text3.Text = "%"
```

```
Text2.Text = ""
```

```
Combo1.Text = ""
```

```
End Sub
```



```

Private Sub Text2_Change()
Dim kayitseti As Recordset
Dim db As Connection
Set db = New Connection
db.Open Adodc1.ConnectionString
Set kayitseti = New Recordset
kayitseti.Open Text1.Text + Text2.Text + Text3.Text, db, adOpenStatic,
adLockOptimistic
Set Adodc1.Recordset = kayitseti
Set DataGrid1.DataSource = kayitseti
End Sub

```

CONTROL

CHECK STUDENT AND ROOMS

There are four buttons in this form, which are 1.Floor, 2.Floor, 3.floor and Main menu. 3 floors and 45 rooms show student dormitory. There are 15 rooms in each floor and 2 persons in each room. Shown in figure 6

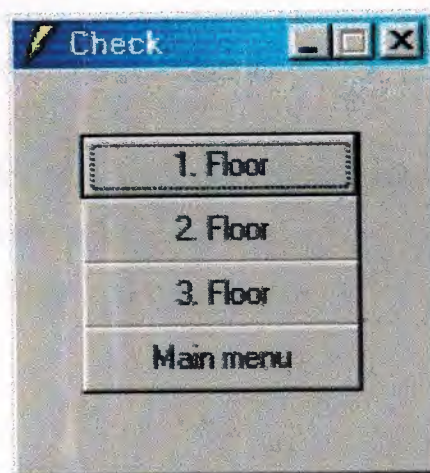


Figure 6

If the user presses a button from Floor Buttons, the user sees easily the person, whose number, name, surname, status staying in that choosed floor, or that sees empty rooms. Before to make a new record, the student's floor and room which is determined easily with this way. Shown in figure 7

floor room control

1. floor			
1. Room		6. Room	
Name: ERHAN	Name:	Name:	Name:
L Name: OZTURK	L Name:	L Name:	L Name:
Number: 971387	Number:	Number:	Number:
Status: Inside	Status:	Status:	Status:
2. Room		7. Room	
Name:	Name:	Name: BÜLENT	Name:
L Name:	L Name:	L Name: DÜRKÜK	L Name:
Number:	Number:	Number: 970302	Number:
Status:	Status:	Status: Inside	Status:
3. Room		8. Room	
Name:	Name:	Name:	Name:
L Name:	L Name:	L Name:	L Name:
Number:	Number:	Number:	Number:
Status:	Status:	Status:	Status:
4. Room		9. Room	
Name:	Name:	Name:	Name:
L Name:	L Name:	L Name:	L Name:
Number:	Number:	Number:	Number:
Status:	Status:	Status:	Status:
5. Room		10. Room	
Name:	Name:	Name: AYDIN	Name:
L Name:	L Name:	L Name: TAŞTAN	L Name:
Number:	Number:	Number: 921213	Number:
Status:	Status:	Status: Inside	Status:
11. Room		12. Room	
Name:	Name:	Name:	Name:
L Name:	L Name:	L Name:	L Name:
Number:	Number:	Number:	Number:
Status:	Status:	Status:	Status:
13. Room		14. Room	
Name:	Name:	Name:	Name:
L Name:	L Name:	L Name:	L Name:
Number:	Number:	Number:	Number:
Status:	Status:	Status:	Status:
15. Room			
Name:	Name:	Name:	Name:
L Name:	L Name:	L Name:	L Name:
Number:	Number:	Number:	Number:
Status:	Status:	Status:	Status:

2. Floor 3. Floor Main menu

Figure 7

I wrote some program codes. Now I will show them to you.

Program Codes for 1. Floor:

```
Private Sub Command1_Click()
```

```
Load floor1
```

```
floor1.Show
```

```
floorcontrol.Hide
```

```
Unload floorcontrol
```

```
If Data1.Recordset.BOF Then
```

```
    MsgBox ("There is no any record")
```

```
Else
```

```
roomno$ = "1"
```

```
bul = "roomno =" & roomno$ & ""
```



```

Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 1 Then
Data1.Recordset.FindFirst bul
floor1.Text1.Text = Data1.Recordset.stdname
floor1.Text2.Text = Data1.Recordset.stdsurname
floor1.Text61.Text = Data1.Recordset.stdnumber
floor1.Text91.Text = Data1.Recordset.situation
Else
floor1.Text1.Text = ""
floor1.Text2.Text = ""
floor1.Text61.Text = ""
floor1.Text91.Text = ""
End If

roomno$ = "1"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 1 Then
Data1.Recordset.FindNext bul
floor1.Text3.Text = Data1.Recordset.stdname
floor1.Text4.Text = Data1.Recordset.stdsurname
floor1.Text62.Text = Data1.Recordset.stdnumber
floor1.Text92.Text = Data1.Recordset.situation
If floor1.Text1.Text = floor1.Text3.Text Then
floor1.Text3.Text = ""
floor1.Text4.Text = ""
floor1.Text62.Text = ""
floor1.Text92.Text = ""
End If
End If

roomno$ = "2"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul

```



```

If Data1.Recordset.roomno = 2 Then
Data1.Recordset.FindFirst bul
floor1.Text13.Text = Data1.Recordset.stdname
floor1.Text14.Text = Data1.Recordset.stdsurname
floor1.Text67.Text = Data1.Recordset.stdnumber
floor1.Text109.Text = Data1.Recordset.situation
Else
floor1.Text13.Text = ""
floor1.Text14.Text = ""
floor1.Text67.Text = ""
floor1.Text109.Text = ""
End If

```

```

roomno$ = "2"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 2 Then
    Data1.Recordset.FindNext bul
    floor1.Text15.Text = Data1.Recordset.stdname
    floor1.Text16.Text = Data1.Recordset.stdsurname
    floor1.Text68.Text = Data1.Recordset.stdnumber
    floor1.Text110.Text = Data1.Recordset.situation
    If floor1.Text13.Text = floor1.Text15.Text Then
        floor1.Text15.Text = ""
        floor1.Text16.Text = ""
        floor1.Text68.Text = ""
        floor1.Text110.Text = ""
    End If
End If

```

```

roomno$ = "3"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 3 Then

```

```

Data1.Recordset.FindFirst bul
floor1.Text25.Text = Data1.Recordset.stdname
floor1.Text26.Text = Data1.Recordset.stdsurname
floor1.Text74.Text = Data1.Recordset.stdnumber
floor1.Text111.Text = Data1.Recordset.situation
Else
floor1.Text25.Text = ""
floor1.Text26.Text = ""
floor1.Text74.Text = ""
floor1.Text111.Text = ""
End If

```

```

roomno$ = "3"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 3 Then
    Data1.Recordset.FindNext bul
    floor1.Text27.Text = Data1.Recordset.stdname
    floor1.Text28.Text = Data1.Recordset.stdsurname
    floor1.Text73.Text = Data1.Recordset.stdnumber
    floor1.Text112.Text = Data1.Recordset.situation
    If floor1.Text25.Text = floor1.Text27.Text Then
        floor1.Text27.Text = ""
        floor1.Text28.Text = ""
        floor1.Text73.Text = ""
        floor1.Text112.Text = ""
    End If
End If

```

```

roomno$ = "4"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 4 Then
    Data1.Recordset.FindFirst bul

```

```

floor1.Text5.Text = Data1.Recordset.stdname
floor1.Text6.Text = Data1.Recordset.stdsurname
floor1.Text80.Text = Data1.Recordset.stdnumber
floor1.Text113.Text = Data1.Recordset.situation
Else
floor1.Text5.Text = ""
floor1.Text6.Text = ""
floor1.Text80.Text = ""
floor1.Text113.Text = ""
End If

```

```

roomno$ = "4"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 4 Then
    Data1.Recordset.FindNext bul
    floor1.Text7.Text = Data1.Recordset.stdname
    floor1.Text8.Text = Data1.Recordset.stdsurname
    floor1.Text79.Text = Data1.Recordset.stdnumber
    floor1.Text114.Text = Data1.Recordset.situation
    If floor1.Text5.Text = floor1.Text7.Text Then
        floor1.Text7.Text = ""
        floor1.Text79.Text = ""
        floor1.Text8.Text = ""
        floor1.Text114.Text = ""
    End If
End If

```

```

roomno$ = "5"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 5 Then
    Data1.Recordset.FindFirst bul
    floor1.Text17.Text = Data1.Recordset.stdname

```


floor1.Text18.Text = Data1.Recordset.stdsurname

floor1.Text86.Text = Data1.Recordset.stdnumber

floor1.Text115.Text = Data1.Recordset.situation

Else

floor1.Text17.Text = ""

floor1.Text18.Text = ""

floor1.Text86.Text = ""

floor1.Text115.Text = ""

End If

roomno\$ = "5"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindNext bul

If Data1.Recordset.roomno = 5 Then

Data1.Recordset.FindNext bul

floor1.Text19.Text = Data1.Recordset.stdname

floor1.Text20.Text = Data1.Recordset.stdsurname

floor1.Text85.Text = Data1.Recordset.stdnumber

floor1.Text116.Text = Data1.Recordset.situation

If floor1.Text17.Text = floor1.Text19.Text Then

floor1.Text19.Text = ""

floor1.Text20.Text = ""

floor1.Text85.Text = ""

floor1.Text116.Text = ""

End If

End If

roomno\$ = "6"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindFirst bul

If Data1.Recordset.roomno = 6 Then

Data1.Recordset.FindFirst bul

floor1.Text29.Text = Data1.Recordset.stdname

floor1.Text30.Text = Data1.Recordset.stdsurname

```

floor1.Text63.Text = Data1.Recordset.stdnumber
floor1.Text93.Text = Data1.Recordset.situation
Else
floor1.Text29.Text = ""
floor1.Text30.Text = ""
floor1.Text63.Text = ""
floor1.Text93.Text = ""
End If

```

```

roomno$ = "6"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 6 Then
    Data1.Recordset.FindNext bul
    floor1.Text31.Text = Data1.Recordset.stdname
    floor1.Text32.Text = Data1.Recordset.stdsurname
    floor1.Text64.Text = Data1.Recordset.stdnumber
    floor1.Text94.Text = Data1.Recordset.situation
    If floor1.Text29.Text = floor1.Text31.Text Then
        floor1.Text31.Text = ""
        floor1.Text32.Text = ""
        floor1.Text64.Text = ""
        floor1.Text94.Text = ""
    End If
End If

```

```

roomno$ = "7"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 7 Then
    Data1.Recordset.FindFirst bul
    floor1.Text9.Text = Data1.Recordset.stdname
    floor1.Text10.Text = Data1.Recordset.stdsurname
    floor1.Text69.Text = Data1.Recordset.stdnumber

```

floor1.Text99.Text = Data1.Recordset.situation

Else

floor1.Text9.Text = ""

floor1.Text10.Text = ""

floor1.Text69.Text = ""

floor1.Text99.Text = ""

End If

roomno\$ = "7"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindNext bul

If Data1.Recordset.roomno = 7 Then

Data1.Recordset.FindNext bul

floor1.Text11.Text = Data1.Recordset.stdname

floor1.Text12.Text = Data1.Recordset.stdsurname

floor1.Text70.Text = Data1.Recordset.stdnumber

floor1.Text100.Text = Data1.Recordset.situation

If floor1.Text9.Text = floor1.Text11.Text Then

floor1.Text11.Text = ""

floor1.Text12.Text = ""

floor1.Text70.Text = ""

floor1.Text100.Text = ""

End If

End If

roomno\$ = "8"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindFirst bul

If Data1.Recordset.roomno = 8 Then

Data1.Recordset.FindFirst bul

floor1.Text21.Text = Data1.Recordset.stdname

floor1.Text22.Text = Data1.Recordset.stdsurname

floor1.Text76.Text = Data1.Recordset.stdnumber

floor1.Text103.Text = Data1.Recordset.situation


```

Else
floor1.Text21.Text = ""
floor1.Text22.Text = ""
floor1.Text76.Text = ""
floor1.Text103.Text = ""
End If

```

```

roomno$ = "8"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 8 Then
    Data1.Recordset.FindNext bul
    floor1.Text23.Text = Data1.Recordset.stdname
    floor1.Text24.Text = Data1.Recordset.stdsurname
    floor1.Text75.Text = Data1.Recordset.stdnumber
    floor1.Text104.Text = Data1.Recordset.situation
    If floor1.Text21.Text = floor1.Text23.Text Then
        floor1.Text23.Text = ""
        floor1.Text24.Text = ""
        floor1.Text75.Text = ""
        floor1.Text104.Text = ""
    End If
End If

```

```

roomno$ = "9"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 9 Then
    Data1.Recordset.FindFirst bul
    floor1.Text33.Text = Data1.Recordset.stdname
    floor1.Text34.Text = Data1.Recordset.stdsurname
    floor1.Text82.Text = Data1.Recordset.stdnumber
    floor1.Text119.Text = Data1.Recordset.situation
Else

```

```
floor1.Text33.Text = ""  
floor1.Text34.Text = ""  
floor1.Text82.Text = ""  
floor1.Text119.Text = ""  
End If
```

```
roomno$ = "9"  
bul = "roomno =" & roomno$ & ""  
Data1.Recordset.FindNext bul  
If Data1.Recordset.roomno = 9 Then  
    Data1.Recordset.FindNext bul  
    floor1.Text35.Text = Data1.Recordset.stdname  
    floor1.Text36.Text = Data1.Recordset.stdsurname  
    floor1.Text81.Text = Data1.Recordset.stdnumber  
    floor1.Text120.Text = Data1.Recordset.situation  
    If floor1.Text33.Text = floor1.Text35.Text Then  
        floor1.Text35.Text = ""  
        floor1.Text36.Text = ""  
        floor1.Text81.Text = ""  
        floor1.Text120.Text = ""  
    End If  
End If
```

```
roomno$ = "10"  
bul = "roomno =" & roomno$ & ""  
Data1.Recordset.FindFirst bul  
If Data1.Recordset.roomno = 10 Then  
    Data1.Recordset.FindFirst bul  
    floor1.Text37.Text = Data1.Recordset.stdname  
    floor1.Text38.Text = Data1.Recordset.stdsurname  
    floor1.Text88.Text = Data1.Recordset.stdnumber  
    floor1.Text117.Text = Data1.Recordset.situation  
Else  
    floor1.Text37.Text = ""
```

```
floor1.Text38.Text = ""  
floor1.Text88.Text = ""  
floor1.Text117.Text = ""
```

```
End If
```

```
roomno$ = "10"
```

```
bul = "roomno =" & roomno$ & ""
```

```
Data1.Recordset.FindNext bul
```

```
If Data1.Recordset.roomno = 10 Then
```

```
    Data1.Recordset.FindNext bul
```

```
    floor1.Text39.Text = Data1.Recordset.stdname
```

```
    floor1.Text40.Text = Data1.Recordset.stdsurname
```

```
    floor1.Text87.Text = Data1.Recordset.stdnumber
```

```
    floor1.Text118.Text = Data1.Recordset.situation
```

```
If floor1.Text37.Text = floor1.Text39.Text Then
```

```
    floor1.Text39.Text = ""
```

```
    floor1.Text40.Text = ""
```

```
    floor1.Text87.Text = ""
```

```
    floor1.Text118.Text = ""
```

```
End If
```

```
End If
```

```
roomno$ = "11"
```

```
bul = "roomno =" & roomno$ & ""
```

```
Data1.Recordset.FindFirst bul
```

```
If Data1.Recordset.roomno = 11 Then
```

```
    Data1.Recordset.FindFirst bul
```

```
    floor1.Text41.Text = Data1.Recordset.stdname
```

```
    floor1.Text42.Text = Data1.Recordset.stdsurname
```

```
    floor1.Text66.Text = Data1.Recordset.stdnumber
```

```
    floor1.Text95.Text = Data1.Recordset.situation
```

```
Else
```

```
    floor1.Text41.Text = ""
```

```
    floor1.Text42.Text = ""
```


floor1.Text66.Text = ""

floor1.Text95.Text = ""

End If

roomno\$ = "11"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindNext bul

If Data1.Recordset.roomno = 11 Then

Data1.Recordset.FindNext bul

floor1.Text43.Text = Data1.Recordset.stdname

floor1.Text44.Text = Data1.Recordset.stdsurname

floor1.Text65.Text = Data1.Recordset.stdnumber

floor1.Text96.Text = Data1.Recordset.situation

If floor1.Text41.Text = floor1.Text43.Text Then

floor1.Text43.Text = ""

floor1.Text44.Text = ""

floor1.Text65.Text = ""

floor1.Text96.Text = ""

End If

End If

roomno\$ = "12"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindFirst bul

If Data1.Recordset.roomno = 12 Then

Data1.Recordset.FindFirst bul

floor1.Text45.Text = Data1.Recordset.stdname

floor1.Text46.Text = Data1.Recordset.stdsurname

floor1.Text72.Text = Data1.Recordset.stdnumber

floor1.Text97.Text = Data1.Recordset.situation

Else

floor1.Text45.Text = ""

floor1.Text46.Text = ""

floor1.Text72.Text = ""

floor1.Text97.Text = ""

End If

roomno\$ = "12"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindNext bul

If Data1.Recordset.roomno = 12 Then

Data1.Recordset.FindNext bul

floor1.Text47.Text = Data1.Recordset.stdname

floor1.Text48.Text = Data1.Recordset.stdsurname

floor1.Text71.Text = Data1.Recordset.stdnumber

floor1.Text98.Text = Data1.Recordset.situation

If floor1.Text45.Text = floor1.Text47.Text Then

floor1.Text47.Text = ""

floor1.Text48.Text = ""

floor1.Text71.Text = ""

floor1.Text98.Text = ""

End If

End If

roomno\$ = "13"

bul = "roomno =" & roomno\$ & ""

Data1.Recordset.FindFirst bul

If Data1.Recordset.roomno = 13 Then

Data1.Recordset.FindFirst bul

floor1.Text49.Text = Data1.Recordset.stdname

floor1.Text50.Text = Data1.Recordset.stdsurname

floor1.Text78.Text = Data1.Recordset.stdnumber

floor1.Text101.Text = Data1.Recordset.situation

Else

floor1.Text49.Text = ""

floor1.Text50.Text = ""

floor1.Text78.Text = ""

floor1.Text101.Text = ""

End If

```

roomno$ = "13"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 13 Then
    Data1.Recordset.FindNext bul
    floor1.Text51.Text = Data1.Recordset.stdname
    floor1.Text52.Text = Data1.Recordset.stdsurname
    floor1.Text77.Text = Data1.Recordset.stdnumber
    floor1.Text102.Text = Data1.Recordset.situation
    If floor1.Text49.Text = floor1.Text51.Text Then
        floor1.Text51.Text = ""
        floor1.Text52.Text = ""
        floor1.Text77.Text = ""
        floor1.Text102.Text = ""
    End If
End If

```

```

roomno$ = "14"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 14 Then
    Data1.Recordset.FindFirst bul
    floor1.Text53.Text = Data1.Recordset.stdname
    floor1.Text54.Text = Data1.Recordset.stdsurname
    floor1.Text84.Text = Data1.Recordset.stdnumber
    floor1.Text105.Text = Data1.Recordset.situation
Else
    floor1.Text53.Text = ""
    floor1.Text54.Text = ""
    floor1.Text84.Text = ""
    floor1.Text105.Text = ""
End If

```



```

roomno$ = "14"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 14 Then
    Data1.Recordset.FindNext bul
    floor1.Text55.Text = Data1.Recordset.stdname
    floor1.Text56.Text = Data1.Recordset.stdsurname
    floor1.Text83.Text = Data1.Recordset.stdnumber
    floor1.Text106.Text = Data1.Recordset.situation
If floor1.Text53.Text = floor1.Text55.Text Then
    floor1.Text55.Text = ""
    floor1.Text56.Text = ""
    floor1.Text83.Text = ""
    floor1.Text106.Text = ""
End If
End If

```

```

roomno$ = "15"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
If Data1.Recordset.roomno = 15 Then
    Data1.Recordset.FindFirst bul
    floor1.Text57.Text = Data1.Recordset.stdname
    floor1.Text58.Text = Data1.Recordset.stdsurname
    floor1.Text89.Text = Data1.Recordset.stdnumber
    floor1.Text107.Text = Data1.Recordset.situation
Else
    floor1.Text57.Text = ""
    floor1.Text58.Text = ""
    floor1.Text89.Text = ""
    floor1.Text107.Text = ""
End If

```

```

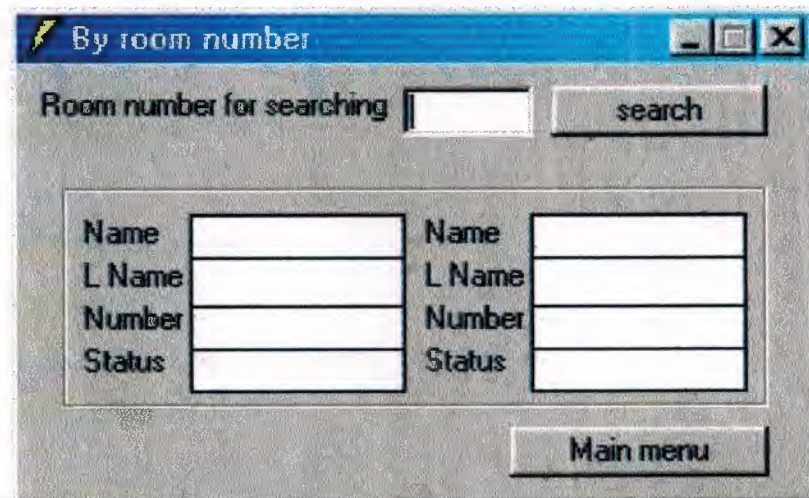
roomno$ = "15"
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
If Data1.Recordset.roomno = 15 Then
    Data1.Recordset.FindNext bul
    floor1.Text59.Text = Data1.Recordset.stdname
    floor1.Text60.Text = Data1.Recordset.stdsurname
    floor1.Text90.Text = Data1.Recordset.stdnumber
    floor1.Text108.Text = Data1.Recordset.situation
    If floor1.Text57.Text = floor1.Text59.Text Then
        floor1.Text59.Text = ""
        floor1.Text60.Text = ""
        floor1.Text90.Text = ""
        floor1.Text108.Text = ""
    End If
End If
End If
End Sub

```

I wrote same program codes for 2. and 3. Floor button.

SEARCH BY ROOM NUMBER

This form is used to learn who stays in which room. The room number is written to textbox and is looked at the rooms, which are empty or full, and owner of the room. The user can see each room status. Shown in figure 8



Room number for searching	
<input type="text"/>	<input type="button" value="search"/>
Name	<input type="text"/>
L Name	<input type="text"/>
Number	<input type="text"/>
Status	<input type="text"/>
<input type="button" value="Main menu"/>	

Figure 8

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
```

```
Load main
```

```
main.Show
```

```
byroomnumber.Hide
```

```
Unload byroomnumber
```

```
End Sub
```

```
Private Sub Command2_Click()
```

```
Dim a, b
```

```
If Text5.Text = "" Then
```

```
    MsgBox "Enter room number"
```

```
    Text5.SetFocus
```

```
Else
```

```
If Text5.Text > 45 Then
```

```
    MsgBox ("Sorry, available room is 45")
```



```

Text5.Text = ""
Text5.SetFocus
Else
roomno$ = Text5.Text
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindFirst bul
a = Val(Text5.Text)
b = Val(Text6.Text)
If a < b Then
MsgBox "Room is empty"
Text5.SelStart = 0
Text5.SelLength = Len(Text5.Text)
Text5.SetFocus
Else:
Text5.SelStart = 0
Text5.SelLength = Len(Text5.Text)
Text5.SetFocus
Frame2.Caption = Text5.Text + "." + " " + "room"
Text1.Text = Data1.Recordset.stdname
Text2.Text = Data1.Recordset.stdsurname
Text61.Text = Data1.Recordset.stdnumber
Text91.Text = Data1.Recordset.situation

roomno$ = Text5.Text
bul = "roomno =" & roomno$ & ""
Data1.Recordset.FindNext bul
Text3.Text = Data1.Recordset.stdname
Text4.Text = Data1.Recordset.stdsurname
Text62.Text = Data1.Recordset.stdnumber
Text92.Text = Data1.Recordset.situation
If Text61.Text = Text62.Text Then
Text3.Text = ""
Text4.Text = ""
Text62.Text = ""

```

```

        Text92.Text = ""
    End If
End If
End If
End If
End Sub

Private Sub Form_Activate()
    Data1.Refresh
End Sub

Private Sub Text5_KeyPress(KeyAscii As Integer)
    If KeyAscii = vbKeyReturn Then
        Command2.Value = True
    End If
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then
        Exit Sub
    Else
        KeyAscii = 0
    End If
End Sub

```

PERMIT

PERMISSION SITUATION

This form is used to give permission to students who will leave. The student number is written to textbox who leaved and Search Button is pressed. If the Checkbox is clicked, the permission is given to student and leaving date is also written after student leaves the dormitory. Cancel Permission Button is used to cancel the permission of student. Shown in figure 9

Figure 9

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
If Data1.Recordset.EOF Then
    MsgBox ("There is no any record")
Else
If Text6.Text = "" Then
MsgBox "Enter student number"
Text6.SetFocus
Else
notgo = Data1.Recordset.Bookmark
stdnumber$ = Text6.Text
bul = "stdnumber=" & stdnumber$ & ""
Data1.Recordset.FindFirst bul
a = Val(Text6.Text)
b = Val(Text1.Text)
```


If a <> b Then

 MsgBox "This student number is not found"

 Text6.SelStart = 0

 Text6.SelLength = Len(Text6.Text)

 Text6.SetFocus

 Data1.Recordset.Bookmark = notgo

Else:

 Text6.SelStart = 0

 Text6.SelLength = Len(Text6.Text)

 Text6.SetFocus

End If

End If

End If

End Sub

Private Sub Command2_Click()

 Load main

 main.Show

 permitform.Hide

 Unload permitform

End Sub

Private Sub Command3_Click()

 Load onleave

 onleave.Show

 Unload permitform

 permitform.Hide

End Sub

Private Sub Command4_Click()

 cevap = MsgBox("Are you sure to cancel permitformting?", 4)

 If cevap = 6 Then

 Text6.Text = ""

 Text6.SetFocus

```

    Text8.Text = ""
    Check1.Value = Unchecked
ElseIf cevap = 7 Then
    Text6.Text = ""
    Text6.SetFocus
End If
End Sub

```

```

Private Sub DBList1_Click()
    stdnumber$ = DBList1.Text
    bul = "stdnumber =" & stdnumber$ & ""
    Data1.Recordset.FindFirst bul
End Sub

```

```

Private Sub Form_Activate()
    Data1.Refresh
End Sub

```

```

Private Sub Form_Load()
    Data1.Refresh
End Sub

```

```

Private Sub Text6_KeyPress(KeyAscii As Integer)
    If KeyAscii = vbKeyReturn Then
        Command1.Value = True
    End If
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then
        Exit Sub
    Else
        KeyAscii = 0
    End If
End Sub

```


I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()  
onleave.Hide  
permitform.Show  
End Sub
```

```
Private Sub Form_Activate()  
datPrimaryRS.Refresh  
End Sub
```

```
Private Sub datPrimaryRS_Error(ByVal ErrorNumber As Long, Description As String,  
ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal  
HelpContext As Long, fCancelDisplay As Boolean)  
MsgBox "Data error event hit err:" & Description  
End Sub
```

```
Private Sub datPrimaryRS_WillChangeRecord(ByVal adReason As  
ADODB.EventReasonEnum, ByVal cRecords As Long, adStatus As  
ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)  
Dim bCancel As Boolean  
Select Case adReason  
Case adRsnAddNew  
Case adRsnClose  
Case adRsnDelete  
Case adRsnFirstChange  
Case adRsnMove  
Case adRsnRequery  
Case adRsnResynch  
Case adRsnUndoAddNew  
Case adRsnUndoDelete  
Case adRsnUndoUpdate  
Case adRsnUpdate  
End Select  
If bCancel Then adStatus = adStatusCancel  
End Sub
```

Private Sub cmdClose_Click()

Unload Me

main.Show

End Sub

PENALTY

OPERATION OF PENALTY

This form is used for determining the students who have discipline punishment. The student number is written to learn to textbox who has discipline punishment and Search Button is pressed.

“Give / Remove penalty “ button is pressed then if checkbox is on clicked, student has penalty then punishment date and reason is written in the textboxes. If penalty will be given, punishment date and reason is written in the textboxes than these informations are recorded by Save button. Punishment can also be canceled by using Cancel button. Shown in figure 11

Penalty

Number: 971387

Name: ERHAN

Surname: ÖZTÜRK

department: Computer

Room no: 1

Student Number: []

Search

Give / Remove Penalty

PENALTY STATUS

Give Penalty: ☐ Penalty Date: []

Reason for the Penalty: []

Save Cancel Exit

List Main menu

Figure 11

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
If Data1.Recordset.BOF Then
    MsgBox ("There is no any record")
Else
If Text6.Text = "" Then
MsgBox "Enter student number"
Text6.SetFocus
Else
notgo = Data1.Recordset.Bookmark
stdnumber$ = Text6.Text
bul = "stdnumber =" & stdnumber$ & ""
Data1.Recordset.FindFirst bul
    a = Val(Text6.Text)
    b = Val(Text1.Text)

    If a <> b Then
        MsgBox "This student number is not found"
        Text6.SelStart = 0
        Text6.SelLength = Len(Text6.Text)
        Text6.SetFocus
        Data1.Recordset.Bookmark = notgo
    Else:
        Text6.SelStart = 0
        Text6.SelLength = Len(Text6.Text)
        Text6.SetFocus
    End If
End If
End If
End Sub
```

```
Private Sub Command2_Click()
```



```
Load main
main.Show
penaltyform.Hide
Unload penaltyform
End Sub
```

```
Private Sub Command3_Click()
Load punishedlist
punishedlist.Show
Unload penaltyform
penaltyform.Hide
End Sub
```

```
Private Sub Command4_Click()
cevap = MsgBox("Are you sure to cancel pealty?", 4)
If cevap = 6 Then
Frame2.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Check1.Enabled = False
Label7.Enabled = False
Label6.Enabled = False
Command4.Enabled = False
Command5.Enabled = False
Command7.Enabled = False
Frame1.Enabled = True
DBList1.Enabled = True
Command6.Enabled = True
Command3.Enabled = True
Command2.Enabled = True
Text6.Text = ""
Text6.SetFocus
Text7.Text = ""
Text8.Text = ""
```

```
Check1.Value = Unchecked
Data1.Recordset.update
ElseIf cevap = 7 Then
Frame2.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Check1.Enabled = False
Label7.Enabled = False
Label6.Enabled = False
```

```
Command4.Enabled = False
Command5.Enabled = False
Command7.Enabled = False
Frame1.Enabled = True
DBList1.Enabled = True
Command6.Enabled = True
Command3.Enabled = True
Command2.Enabled = True
Text6.Text = ""
Text6.SetFocus
Data1.Recordset.CancelUpdate
End If
End Sub
```

```
Private Sub Command5_Click()
cevap = MsgBox("This penalty will save are you sure?", 4)
If cevap = 6 Then
Frame2.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Check1.Enabled = False
Label7.Enabled = False
Label6.Enabled = False
```

```

Command4.Enabled = False
Command5.Enabled = False
Command7.Enabled = False
Frame1.Enabled = True
DBList1.Enabled = True
Command6.Enabled = True
Command3.Enabled = True
Command2.Enabled = True
    Text6.Text = ""
    Text6.SetFocus
    Data1.Recordset.update
ElseIf cevap = 7 Then
    Frame2.Enabled = False
    Label7.Enabled = False
    Label6.Enabled = False
    Text7.Enabled = False
    Text8.Enabled = False
    Check1.Enabled = False
    Command4.Enabled = False
    Command5.Enabled = False
    Command7.Enabled = False

    Frame1.Enabled = True
    DBList1.Enabled = True
    Command6.Enabled = True
    Command3.Enabled = True
    Command2.Enabled = True
        Text6.Text = ""
        Text6.SetFocus
    Data1.Recordset.CancelUpdate
End If
End Sub

Private Sub Command6_Click()

```


If Data1.Recordset.BOF Then

Else

Data1.Recordset.Edit

Frame1.Enabled = False

Frame2.Enabled = True

Label7.Enabled = True

Label6.Enabled = True

Text7.Enabled = True

Text8.Enabled = True

Check1.Enabled = True

Command4.Enabled = True

Command5.Enabled = True

Command7.Enabled = True

DBList1.Enabled = False

Command6.Enabled = False

Command3.Enabled = False

Command2.Enabled = False

End If

End Sub

Private Sub Command7_Click()

Frame2.Enabled = False

Label7.Enabled = False

Label6.Enabled = False

Text7.Enabled = False

Text8.Enabled = False

Check1.Enabled = False

Command4.Enabled = False

Command5.Enabled = False

Command7.Enabled = False

Frame1.Enabled = True

DBList1.Enabled = True

Command6.Enabled = True

Command3.Enabled = True

```

Command2.Enabled = True
    Text6.Text = ""
    Text6.SetFocus
Data1.Recordset.CancelUpdate
End Sub

```

```

Private Sub DBList1_Click()
    stdnumber$ = DBList1.Text
    bul = "stdnumber =" & stdnumber$ & ""
Data1.Recordset.FindFirst bul
    End Sub

```

```

Private Sub Form_Activate()
Data1.Refresh
Frame2.Enabled = False
Label7.Enabled = False
Label6.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Check1.Enabled = False
Command4.Enabled = False
Command5.Enabled = False
Command7.Enabled = False
End Sub

```

```

Private Sub Form_Load()
Data1.Refresh
End Sub

```

```

Private Sub Text6_KeyPress(KeyAscii As Integer)
    If KeyAscii = vbKeyReturn Then
        Command1.Value = True
    End If
    If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9")) Or KeyAscii = vbKeyBack Then

```

```

Exit Sub
Else
    KeyAscii = 0
End If
End Sub
Private Sub Text8_KeyPress(KeyAscii As Integer)
If (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii = Asc("/")) Or
KeyAscii = vbKeyBack Then
    Exit Sub
Else
    KeyAscii = 0
End If
End Sub

```

LIST OF PUNISHMENT

There is the list of students who have punishment in this form. Shown in figure 12

Name	Surname	Number	Department	Room no	Cause of penalty	Penalty date
AYDIN	TAŞTAN	921213	Computer	10	We write penalty reason in this part	11/12/2001

The application window includes a 'Back' button and a 'Main menu' button at the bottom right.

Figure 12

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
```

```
    punishedlist.Hide
```

```
    penaltyform.Show
```

```
End Sub
```

```
Private Sub Form_Activate()
```

```
    datPrimaryRS.Refresh
```

```
End Sub
```

```
Private Sub datPrimaryRS_Error(ByVal ErrorNumber As Long, Description As String,  
    ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal  
    HelpContext As Long, fCancelDisplay As Boolean)
```

```
    MsgBox "Data error event hit err:" & Description
```

```
End Sub
```

```
Private Sub datPrimaryRS_WillChangeRecord(ByVal adReason As  
    ADODB.EventReasonEnum, ByVal cRecords As Long, adStatus As  
    ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)
```

```
    Dim bCancel As Boolean
```

```
    Select Case adReason
```

```
        Case adRsnAddNew
```

```
        Case adRsnClose
```

```
        Case adRsnDelete
```

```
        Case adRsnFirstChange
```

```
        Case adRsnMove
```

```
        Case adRsnRequery
```

```
        Case adRsnResynch
```

```
        Case adRsnUndoAddNew
```

```
        Case adRsnUndoDelete
```

```
        Case adRsnUndoUpdate
```

```
        Case adRsnUpdate
```

```
    End Select
```

```
    If bCancel Then adStatus = adStatusCancel
```

End Sub

Private Sub cmdClose_Click()

Unload Me

main.Show

End Sub

STUDENT CONTROL

This form is used for learning which student is inside or outside. All of the student's lists are shown in the grid, when form is opened firstly. There are three options buttons under the form, which are Inside, Outside and All students. Which position is wanted to control, that button is pressed. Shown in figure 13

	Name	Surname	Number	department	Room no
▶	ERHAN	OZTURK	971387	Computer	1
	BULENT	DURUKAN	970302	Computer	7
	AYDIN	TAŞTAN	921213	Computer	10

☒ Inside ☐ Outside ☐ All students Total Students Number= 3

Main menu

Figure 13

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Form_Activate()  
datPrimaryRS.Refresh  
End Sub
```

```
Private Sub Form_Load()  
Text4.Text = "Total Students Number=" + Str(grdDataGrid.ApproxCount)  
End Sub
```

```
Private Sub Form_Resize()  
On Error Resume Next  
    grdDataGrid.Height = Me.ScaleHeight - datPrimaryRS.Height - 30 -  
picButtons.Height  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)  
Screen.MousePointer = vbDefault  
End Sub
```

```
Private Sub datPrimaryRS_Error(ByVal ErrorNumber As Long, Description As String,  
ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal  
HelpContext As Long, fCancelDisplay As Boolean)  
    MsgBox "Data error event hit err:" & Description  
End Sub
```

```
Private Sub datPrimaryRS_WillChangeRecord(ByVal adReason As  
ADODB.EventReasonEnum, ByVal cRecords As Long, adStatus As  
ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)  
    Dim bCancel As Boolean  
    Select Case adReason  
        Case adRsnAddNew  
        Case adRsnClose
```



```

Case adRsnDelete
Case adRsnFirstChange
Case adRsnMove
Case adRsnRequery
Case adRsnResynch
Case adRsnUndoAddNew
Case adRsnUndoDelete
Case adRsnUndoUpdate
Case adRsnUpdate
End Select
If bCancel Then adStatus = adStatusCancel
End Sub

```

```

Private Sub cmdClose_Click()
    Unload Me
    main.Show
End Sub

```

```

Private Sub Option1_Click()
    Text1.Text = "select
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE situation like '%"
    Text3.Text = "%'"
    Text2.Text = "Inside"
    Text4.Text = "Inside Students Number=" + Str(grdDataGrid.ApproxCount)
End Sub

```

```

Private Sub Option2_Click()
    Text1.Text = "select
stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE situation like '%"
    Text3.Text = "%'"
    Text2.Text = "Outside"
    Text4.Text = "Outside Students Number=" + Str(grdDataGrid.ApproxCount)

```

End Sub

Private Sub Option3_Click()

Text1.Text = "select

stdnumber,stdname,stdsurname,department,recorddate,roomno,situation,fathername,mothername from yurt WHERE stdname like "%"

Text3.Text = "%"

Text2.Text = ""

Text4.Text = "Total Students Number=" + Str(grdDataGrid.ApproxCount)

End Sub

Private Sub Text2_Change()

Dim kayitseti As Recordset

Dim db As Connection

Set db = New Connection

db.Open datPrimaryRS.ConnectionString

Set kayitseti = New Recordset

kayitseti.Open Text1.Text + Text2.Text + Text3.Text, db, adOpenStatic,
adLockOptimistic

Set datPrimaryRS.Recordset = kayitseti

Set grdDataGrid.DataSource = kayitseti

End Sub

VISITOR

This form is used for checking visitors who came to dormitory and is recorded to visitor informations. There is New Record, Delete, Modify, Press visitor card, Main Menu buttons in this form. New record button is used to record visitor informations, Delete button is used to erase visitor informations from database and Modify is used to update or change the visitor informations. All of the visitors are also shown in the grid that is at bottom of the form. There is another button in the form, which is used for printing visitor card that is used by visitors. Shown in figure 14

The screenshot shows a Windows application window titled "Visitors". The window has a menu bar with "File", "Edit", and "Help". Below the menu bar, there are several input fields for a new visitor record:

- Name: HÜSEYİN
- Surname: ÖZTÜRK
- Licence number: 64887645
- Which room is visited: 1
- Who is visited: ERHAN
- Entering time: 18:23

There are three buttons: "New record", "Delete", and "Modify". A "Press visitor card" button is also present. A "Main menu" button is located below the input fields.

Below the input fields, there is a table titled "LISTS OF VISITORS". The table has the following columns: Name, Surname, Licence no, Who is visited, Visited room, and Entering Time.

Name	Surname	Licence no	Who is visited	Visited room	Entering Time
HÜSEYİN	ÖZTÜRK	64887645	ERHAN	1	18:23
AHMET	ÖZTÜRK	65478984	BÜLENT	2	14:45

Figure 14

I wrote some program codes. Now I will show them to you.

Program Codes :

```
Private Sub Command1_Click()
```

```
Load main
```

```
main.Show
```

```
visitorform.Hide
```

```
Unload visitorform
```

```
End Sub
```

```
Private Sub Command10_Click()
```

```
CommonDialog1.Action = 5
```

```
Printer.PaperSize = vbPRPSA4
```

```
Printer.PrintQuality = vbPRPQMedium
```

```
Printer.PaperBin = 7
```

```
Printer.Scale (2, 0)-(210, 296)
```



```

Printer.Font = "arial"
Printer.FontSize = 20

Printer.CurrentX = 10
Printer.CurrentY = 10
Printer.Print "VISITOR CARD"

Printer.FontSize = 12

Printer.CurrentX = 10
Printer.CurrentY = 20
Printer.Print "Name.....:" + Text1(0)
Printer.CurrentX = 10
Printer.CurrentY = 25
Printer.Print "Surname.....:" + Text1(1)
Printer.CurrentX = 10
Printer.CurrentY = 30
Printer.Print "Licence Number.....:" + Text1(2)
Printer.CurrentX = 10
Printer.CurrentY = 35
Printer.Print "Which Room is Visited.....:" + Text1(3)
Printer.CurrentX = 10
Printer.CurrentY = 40
Printer.Print "Who is Visitede.....:" + Text1(4)
Printer.CurrentX = 10
Printer.CurrentY = 45
Printer.Print "Entering Time.....:" + Text1(5); ""

Printer.EndDoc
End Sub

Private Sub Command2_Click()
Text1(0).SetFocus
datPrimaryRS.Refresh

```

```

datPrimaryRS.Recordset.addnew
For i = 0 To 5
    Text1(i).Text = ""
    Text1(i).Locked = False
Next
Command2.Visible = False
Command4.Visible = True
Command3.Visible = True
Command1.Visible = False
Command5.Visible = False
Command6.Visible = False
Command7.Visible = False
Command8.Visible = False
End Sub

```

```

Private Sub Command3_Click()
    If Text1(0) = "" Then
        Text1(0).SetFocus
    ElseIf Text1(1) = "" Then
        Text1(1).SetFocus
    ElseIf Text1(2) = "" Then
        Text1(2).SetFocus
    ElseIf Text1(3) = "" Then
        Text1(3).SetFocus
    ElseIf Text1(4) = "" Then
        Text1(4).SetFocus
    ElseIf Text1(5) = "" Then
        Text1(5).SetFocus
    Else
        cevap = MsgBox("This record will save are you sure?", 4)
        If cevap = 6 Then
            datPrimaryRS.Recordset.Save
        Else
            datPrimaryRS.Recordset.Cancel
        End If
    End If
End Sub

```

```
datPrimaryRS.Refresh
```

```
End If
```

```
For i = 0 To 5
```

```
Text1(i).Locked = True
```

```
Next
```

```
Command2.Visible = True
```

```
Command4.Visible = False
```

```
Command3.Visible = False
```

```
Command1.Visible = True
```

```
Command5.Visible = True
```

```
Command6.Visible = True
```

```
Command7.Visible = False
```

```
Command8.Visible = False
```

```
End If
```

```
End Sub
```

```
Private Sub Command4_Click()
```

```
datPrimaryRS.Recordset.Cancel
```

```
datPrimaryRS.Refresh
```

```
For i = 0 To 5
```

```
Text1(i).Locked = True
```

```
Next
```

```
Command2.Visible = True
```

```
Command4.Visible = False
```

```
Command3.Visible = False
```

```
Command1.Visible = True
```

```
Command5.Visible = True
```

```
Command6.Visible = True
```

```
Command7.Visible = False
```

```
Command8.Visible = False
```

```
End Sub
```

```
Private Sub Command5_Click()
```

```
On Error GoTo DeleteErr
```



```

seçiminiz = MsgBox("Are you sure to delete this record ?", 36, "Warning")
If seçiminiz = 6 Then
    With datPrimaryRS.Recordset
        .delete
        .MoveNext
        If .EOF Then .MoveLast
    End With
    Exit Sub
DeleteErr:
    MsgBox Err.Description
    End If
End Sub

Private Sub Command6_Click()
    If datPrimaryRS.Recordset.BOF Then
        MsgBox ("There is no any record")
    Else
        For i = 0 To 5
            Text1(i).Locked = False
        Next
        Text1(0).SetFocus
        Command2.Visible = False
        Command7.Visible = True
        Command8.Visible = True
        Command1.Visible = False
        Command5.Visible = False
        Command6.Visible = False
        Command3.Visible = False
        Command4.Visible = False
    End If
End Sub

Private Sub Command7_Click()
    cevap = MsgBox("This record will be saved?", 4)

```

```

    If cevap = 6 Then
        datPrimaryRS.Recordset.update
        Command2.Visible = True
        Command7.Visible = False
        Command8.Visible = False
        Command1.Visible = True
        Command5.Visible = True
        Command6.Visible = True
        Command3.Visible = False
        Command4.Visible = False
        For i = 0 To 5
            Text1(i).Locked = True
        Next
    ElseIf cevap = 7 Then
        datPrimaryRS.Recordset.Cancel
        datPrimaryRS.Refresh
        Command2.Visible = True
        Command7.Visible = False
        Command8.Visible = False
        Command1.Visible = True
        Command5.Visible = True
        Command6.Visible = True
        Command3.Visible = False
        Command4.Visible = False
        For i = 0 To 5
            Text1(i).Locked = True
        Next
    End If
End Sub

```

```

Private Sub Command8_Click()
    datPrimaryRS.Recordset.Cancel
    datPrimaryRS.Refresh
    Command2.Visible = True

```

```

Command7.Visible = False
Command8.Visible = False
Command1.Visible = True
Command5.Visible = True
Command6.Visible = True
Command3.Visible = False
Command4.Visible = False
For i = 0 To 5
    Text1(i).Locked = True
Next
End Sub

```

```

Private Sub Form_Load()
    For i = 0 To 5
        Text1(i).Locked = True
    Next
    Command3.Visible = False
    Command4.Visible = False
    Command7.Visible = False
    Command8.Visible = False
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
    Screen.MousePointer = vbDefault
End Sub

```

```

Private Sub datPrimaryRS_Error(ByVal ErrorNumber As Long, Description As String,
    ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal
    HelpContext As Long, fCancelDisplay As Boolean)
    MsgBox "Data error event hit err:" & Description
End Sub

```



```
Private Sub datPrimaryRS_WillChangeRecord(ByVal adReason As  
ADODB.EventReasonEnum, ByVal cRecords As Long, adStatus As  
ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)
```

```
    Dim bCancel As Boolean
```

```
    Select Case adReason
```

```
        Case adRsnAddNew
```

```
        Case adRsnClose
```

```
        Case adRsnDelete
```

```
        Case adRsnFirstChange
```

```
        Case adRsnMove
```

```
        Case adRsnRequery
```

```
        Case adRsnResynch
```

```
        Case adRsnUndoAddNew
```

```
        Case adRsnUndoDelete
```

```
        Case adRsnUndoUpdate
```

```
        Case adRsnUpdate
```

```
    End Select
```

```
    If bCancel Then adStatus = adStatusCancel
```

```
End Sub
```

```
Private Sub Text1_KeyPress(Index As Integer, KeyAscii As Integer)
```

```
    Dim NextIndex As Integer
```

```
    If Command3.Visible = True Then
```

```
        If KeyAscii = vbKeyReturn Then
```

```
            NextIndex = Index + 1
```

```
            If NextIndex = 6 Then
```

```
                Command3.SetFocus
```

```
            Else
```

```
                Text1(NextIndex).SetFocus
```

```
            End If
```

```
        End If
```

```
    End If
```

```
End Sub
```

CONCLUSION :

This project obtains me to increase my knowledge about Visual Basic and finding solutions to some specific problems. Before we learned programming in traditional languages. This kind of programming languages obligates the programmer to write lots of codes and procedures to develop similar program that includes data functions (as searching, browsing, finding an information, deleting, etc.), buttons, different types of text-boxes.

In visual languages these are allowed by program, maintenance and improvement of program is very easy to rearrangement. Object oriented programming gives me opinion about to make best situation for designing program in very sufficient conditions.

References

- [1] Assoc. Prof. Dr. Şenol Bektaş, Prof. Dr. Fakhreddin Mamedov, Assoc. Prof. Dr. Doğan Haktanır, Asst. Prof. Dr. Kadri Bürüncük, Mr. Özgür C. Özerdem, NEU-CEE 2001 SYMPOSIUM, Neu press, TRNC Lefkoşa 23 May 2001
- [2] Memik Yanık, Microsoft Visual Basic 6.0 For Windows, Bate Basım Yayım Dağıtım, İstanbul, May 2000
- [3] İhsan Karagülle & Zeydin Pala, Visual Basic Pro , Bate Basım Yayım Dağıtım, İstanbul 2000
- [4] www.programlama.com
- [5] [www. Vbproj.com](http://www.Vbproj.com)