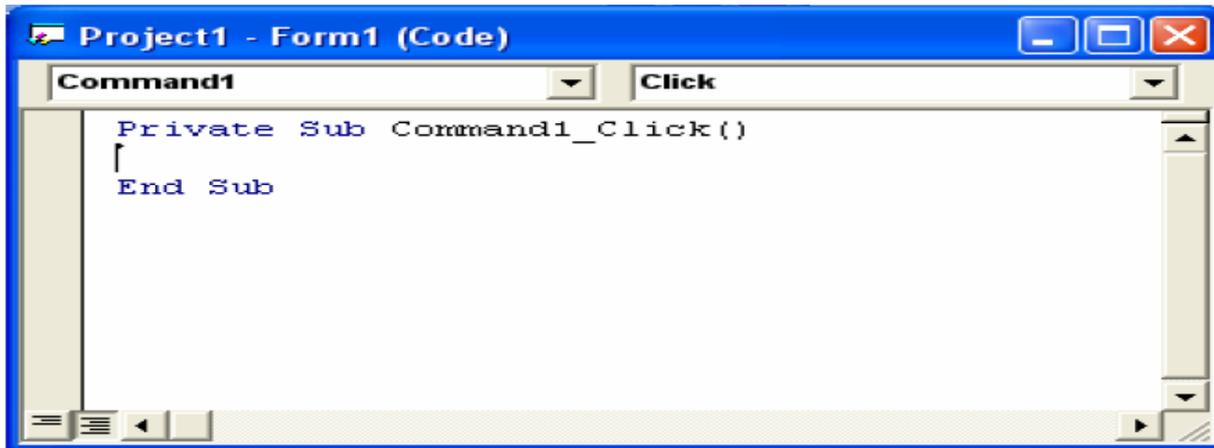**Writing Code**

To program a specific key, the programming window must appear, as this window contains commands or codes to program a specific key. The quick way to show the program window is to press the key twice in a row, and the window appears as follows: -

In this window, we notice two list boxes, one of which contains all the objects on the form screen, and the other contains all the events that occur on the button, which is here. Click; that is when this key is pressed.
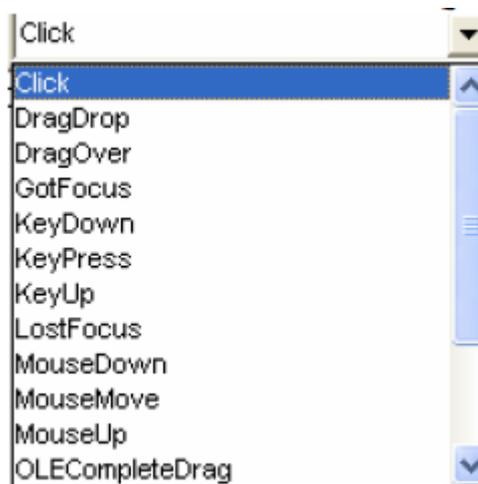
We notice that there are two lines in the programming window, namely- :

First line: - Private Sub Command1_Click()

This line means that a particular event will occur on the command button only, and we notice the presence of the word command1, i.e. the name of the button in the Name property, and then the presence of Click, i.e. when it is pressed.

The second line: End Sub, which means the end of the event.

What is the event?

 The event is all the movements the user makes with the mouse, such as clicking or double-clicking, and the Visual Basic program responds to them. Without their presence, we would not have been able to specify for the program when the command we want to execute should be executed.

```
Project1 - Form1 (Code)                          _ □ ×
Form                          ▼   Load                     ▼
    Private Sub Form Load()  ─────────────── الحدث
    ┌──────────────────┐
    │                  │
    └──────────────────┘
    End Sub        ────── الاوامر        الاداة
```

recipe	the event
It occurs after the window is loaded, i.e. after the load event has occurred	active
When single clicking on the mouse	Click
When you double–click on the mouse	dbclick
When taking focus on a specific tool	gotfocus
When losing focus on a particular tool	lostfocus
When the button goes down	keydown
When you press any button on the keyboard	keypress
When loading the main window	load
When the mouse button goes down	mousedown
When the mouse moves on a specific tool	mousemove
When the mouse button rises to the top	mouseup
when redrawing the window	paint

when resizing the window	resize
When unloading the window	unload

Timer:

The "time tool is an invisible tool" is intended to be a tool for timing certain events that the user does not want to know the time, and the timer tool can be inserted anywhere in the form.

Example 1: In this program, we use an essential software technique, the "counter" timer.

First: Create the objects that have the shape and non-properties as follows:

object	property	preparation
Form1	Caption	digital clock software
	Right to lift	True
Label1	Caption	" "
Timer1	Interval	1000

The program window will appear as shown in the figure:

Second: Writing the code for the program:

Write the code for the object (Timer1):

Private Sub Timer1_Timer()

Label1.Caption = Time
End Sub

When you run the program, it will look like this:

A simplified explanation of the previous code:

The counter is completely similar to the counters that we use in our practical life, but the counter in Visual Basic is sophisticated and advanced as it can be set through properties, and this counter is used to control some program objects; for example, we made the counter hold the label tool (Label1) so that it appears The time, the counter changes the time every 0.001 of a second. The time changes automatically after 0.001 of a second has passed, so we have created a digital clock.

Example 2:

We will make a simple program that contains keys to show the date and time, a welcome message, a clear key, and another to exit the program.

First: create the objects on the table.

الكائن	الخاصية	الإعداد
Form1	Caption	Welcome programme
	Right to lift	True
Text1	Caption	" "
Command1	Caption	Show the current time
Command2	Caption	Show the current date
Command3	Caption	Show your welcome
Command4	Caption	clean
Command5	Caption	Exit

The program window will appear as shown in the figure
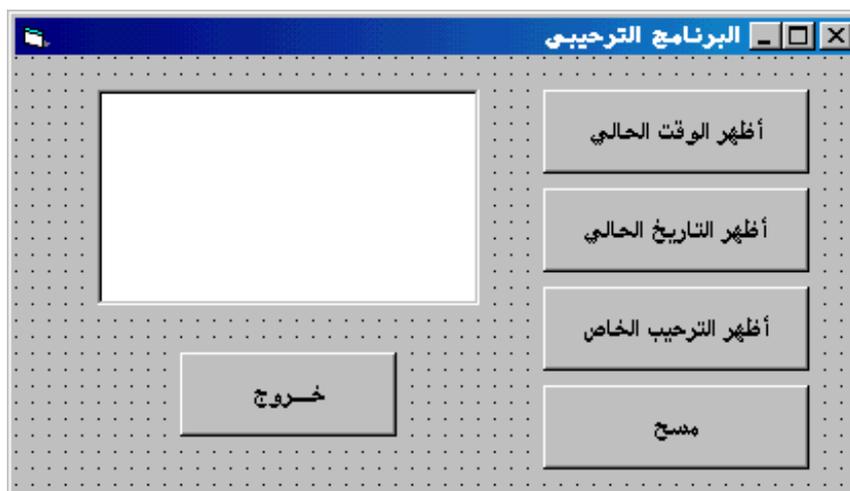
Second: Writing the code for the program:

Note: Writing the code for the program means writing commands and using them in programming the program.

Writing the code for the Show current time button (Command1):

Text1. Text = Time

Write the code to display the current date (Command2):

Text1.Text = Date

Write the code to show the welcome text (Command3):

Text1.Text = "Welcome to the era of Visual Basic."

Write the code to clear the contents of the text box (Command4):

Text1. Text =""

Write the code to exit the program (Command5):

End

A simplified explanation of the previous code:

In writing the code for the button to show the current time, we assign the value of time and knowledge in Visual Basic as (Time) to the text box. We perform a similar operation in the date code, assigning the date value to the text box. In the code for the welcome text, we assign a literal value to the text box, And when we wanted to empty the text box, we commanded Visual Basic to set an empty value in this box, that is, to make it empty.

Note: When we wrote the code to exit the program, we wrote the word (End), and it closed the program with all its contents.

List boxes for drives, directories, and files

Drive listings, which allow you to see valid drives on your computer. Directory listings will enable you to move between folders on a specific drive, and File listings allow you to pick a specific file in a folder.

In the next exercise, you will use these three objects to build a Photo Browser that locates and displays files containing artwork on your computer.

Example 3: Photo Browser

Photo Browser uses three file system objects, an image object, and several lines of code to create a photo browser. After completing the program, you can use it in your daily work to view the images stored on your computer. Or on a floppy disk or a temporary storage unit.
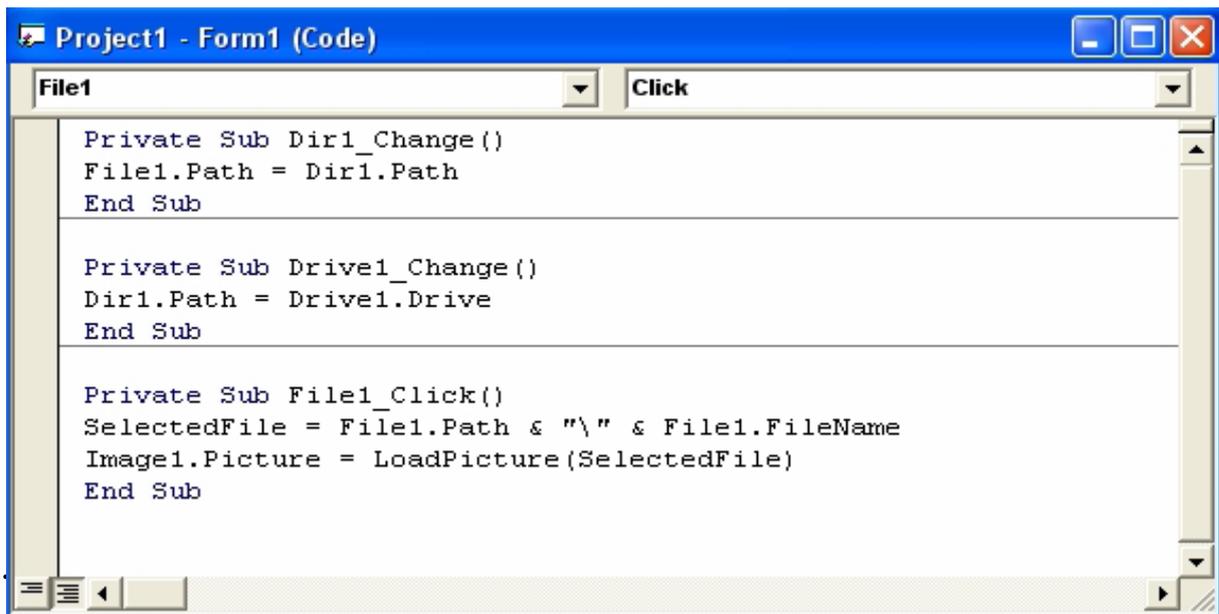
Building the program Photo Browser

1. Create a program and design the following interface:
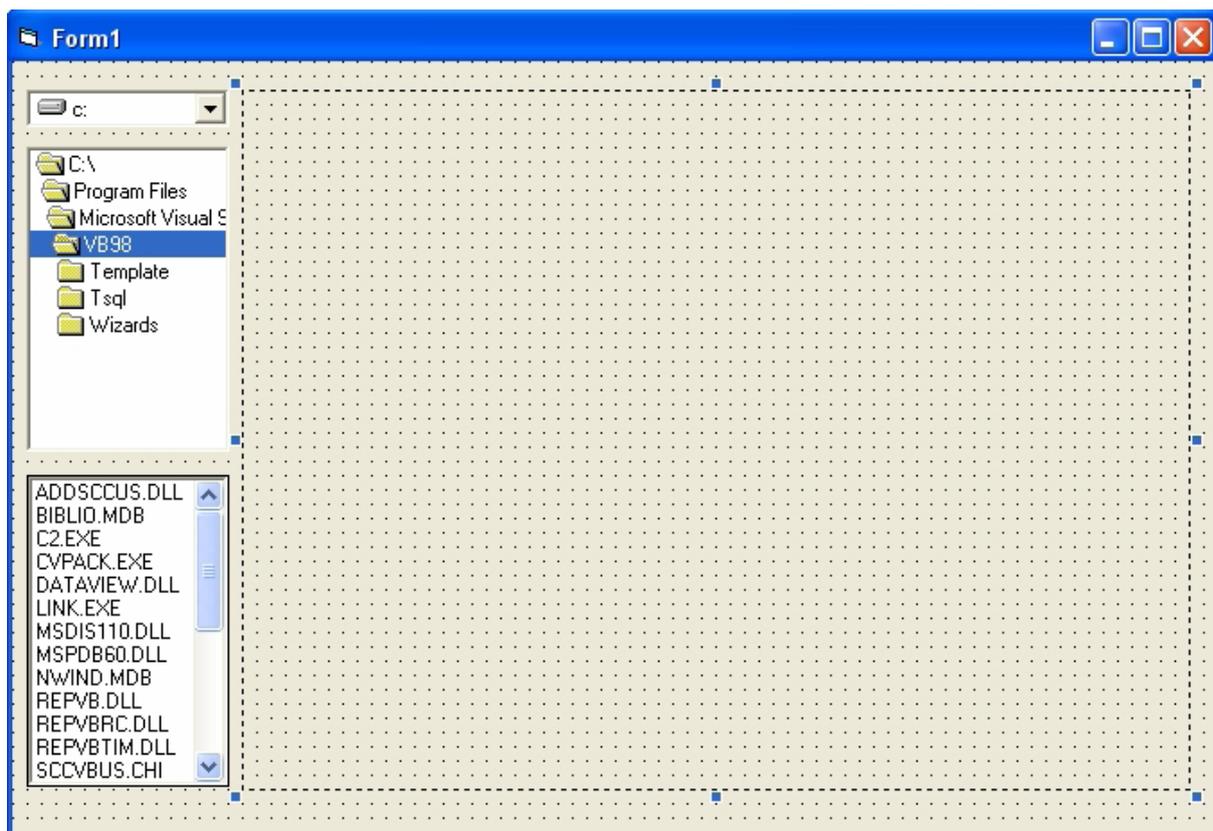
2. Set the properties of the model and its objects according to the following table:

object	property	preparation
File1	Pattern	*.bmp; *.jpg; *.gif *.ico;
Image1	Stretch	True

3. Write the programming as follows:

```
Project1 - Form1 (Code)

File1                          Click

    Private Sub Dir1_Change()
    File1.Path = Dir1.Path
    End Sub

    Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
    End Sub

    Private Sub File1_Click()
    SelectedFile = File1.Path & "\" & File1.FileName
    Image1.Picture = LoadPicture(SelectedFile)
    End Sub
```

4. When the program is executed, the following occurs:
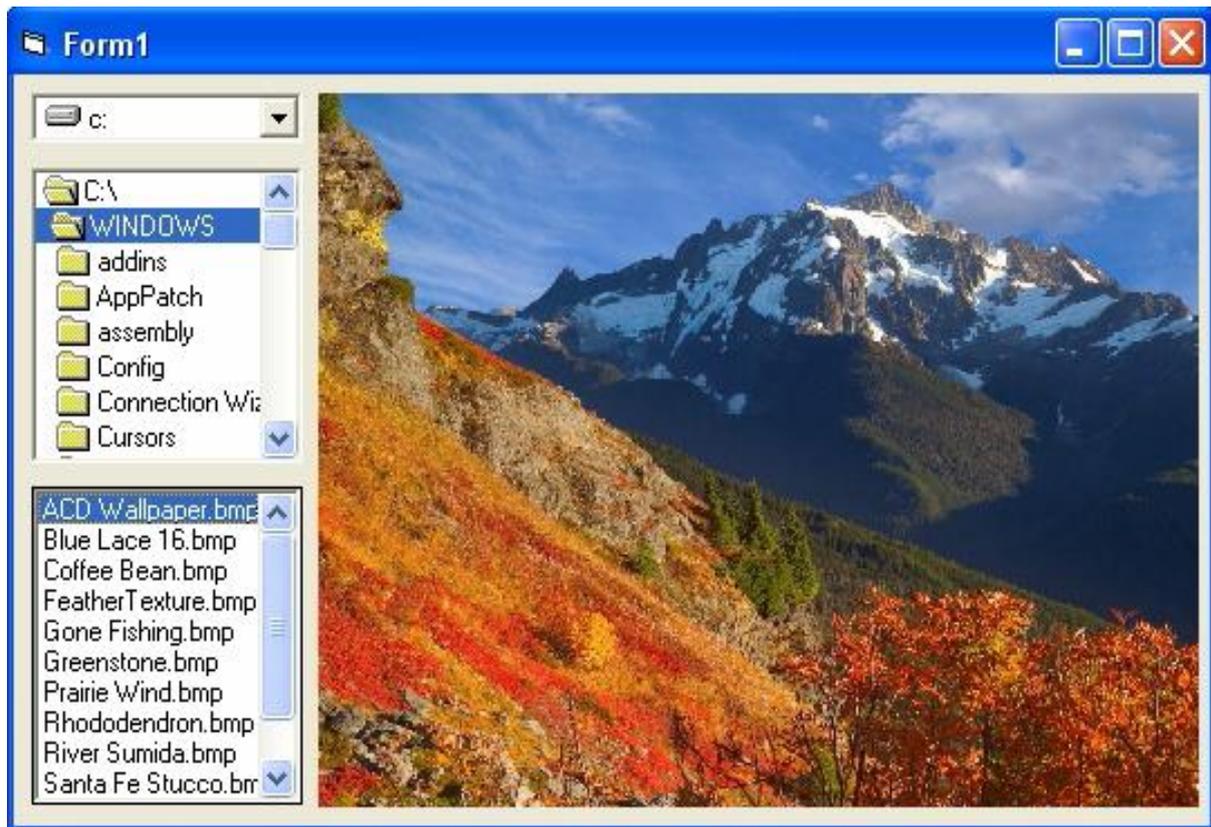
Practical example:

Design a window that contains a (Command Button) element that, when clicked, changes the value of the Caption property of the form to (yes)٠

Practical example 2: Design a window containing two elements (Command Button). When clicking on any of them prints the value of its (Name) property on the form (change the Name property of the second element to be CMD2)