VISUAL BASIC

NAZARETH COLLEGE OF ARTS AND SCIENCE

Department of Computer Science

VISUAL PROGRAMMING

Visual Basic concepts and Programming

By
Dr. M. Thyagaraj
Assistant Professor

VISUAL BASIC

Unit - I

1. Customizing a Form
2. Writing Simple Programs
3. Toolbox
4. Creating Controls
   1. Name Property
   2. Text Boxes and Labels
   3. Image Controls

   1.4.3 Command Button
5. Message Boxes
6. Grid - Editing Tools - Access Keys
7. Variables
8. Data Types
9. Strings, Numbers and Dates.

1.Introduction

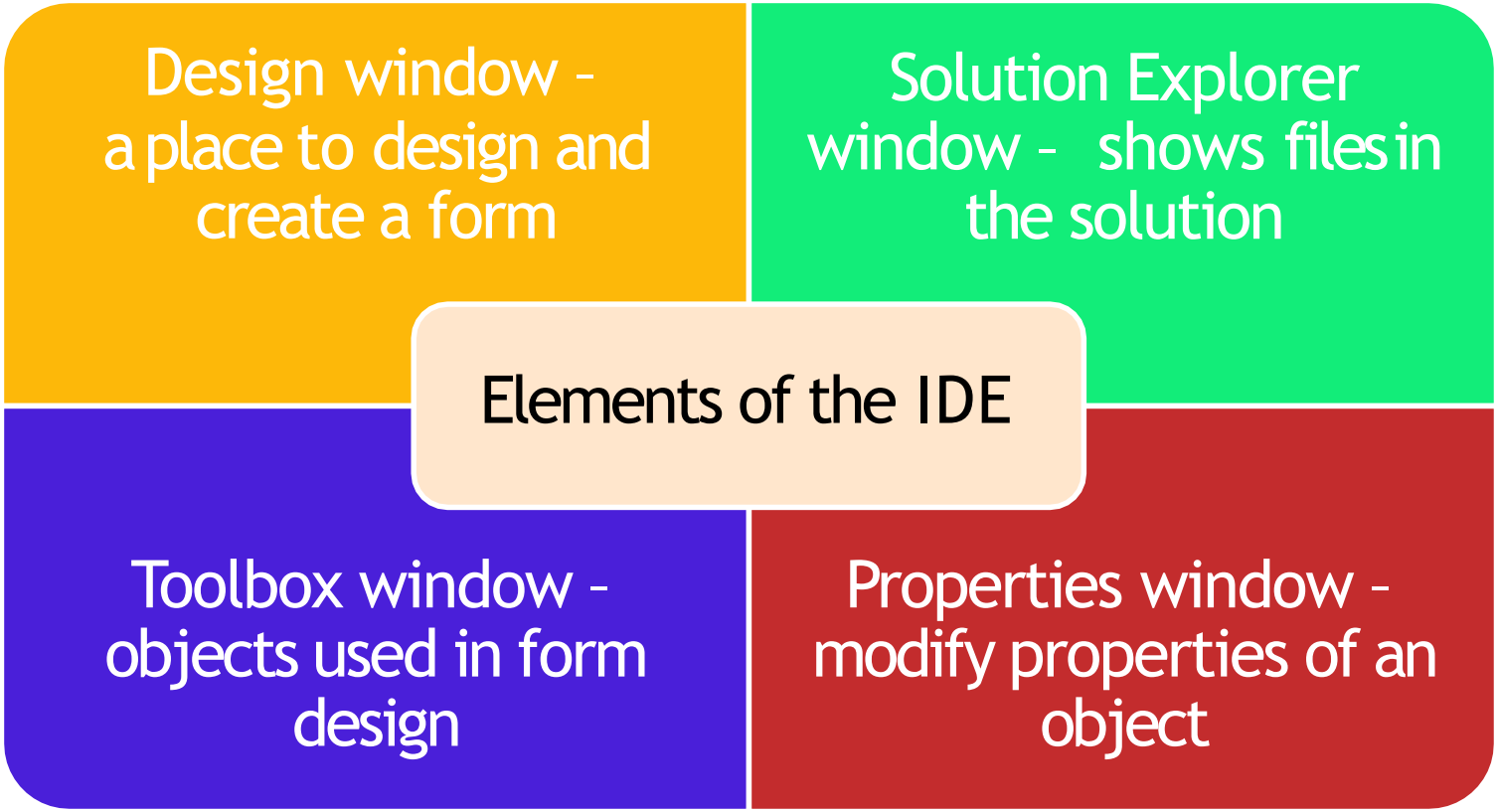
- Visual Basic is a third generation event driven programming language first released my Microsoft in 1991
- It evolved from the earlier DOS version called BASIC (Beginners' All-purpose Symbolic Instruction Code)
- The latest version of Visual Basic is 6.0
- Visual Basic is a user friendly programming language
- It enables anyone to develop GUI window application easily.

VISUAL BASIC

FEATURES OF VB

- Integrated Development Environment
- Efficient and fast GUI application
- Common Object Model (activeX, COM, DCOM)
- It also supports open ODBC
- supports the client/server architecture

Environment



- Dynamic Help window – a handy reference tool
- Toolbar – contains icons for frequently used functions
- Tooltips – a short description of button's purpose

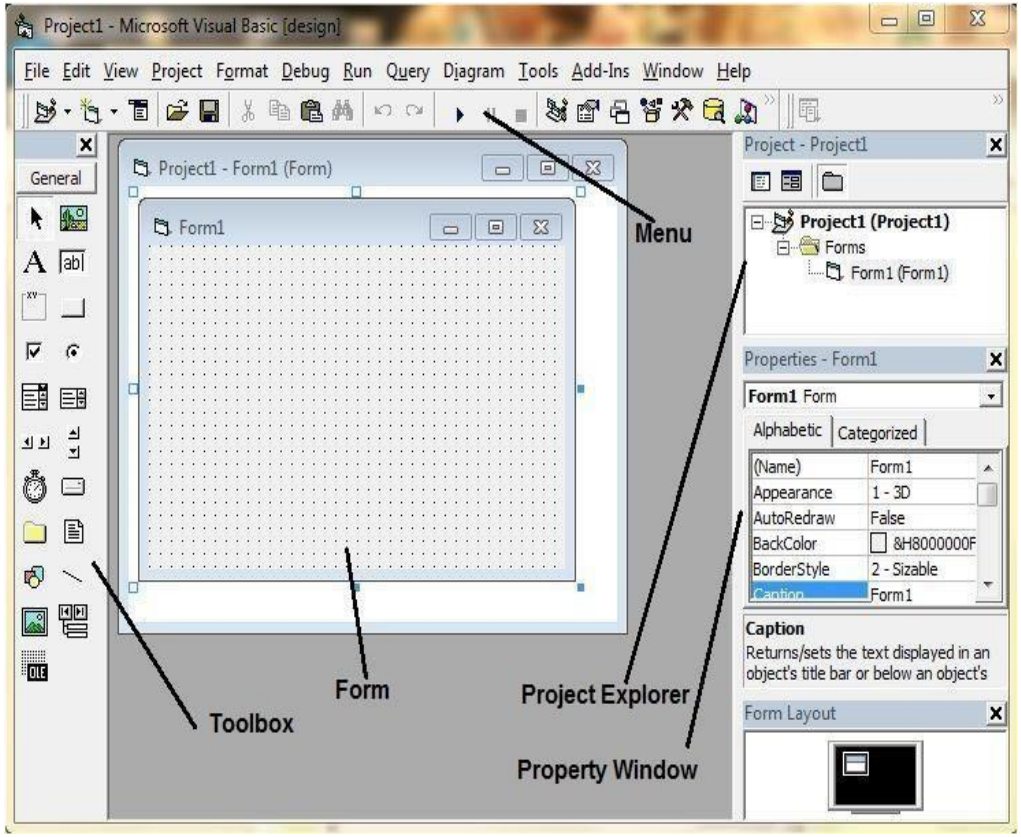
VISUAL BASIC

1.1Customizing a Form

- A form is one of the main building in a visual basic.
- You can draw and view controls on a form.
- Forms are containers for controls.
- User interacts with the controls on a form to obtain the desired result.

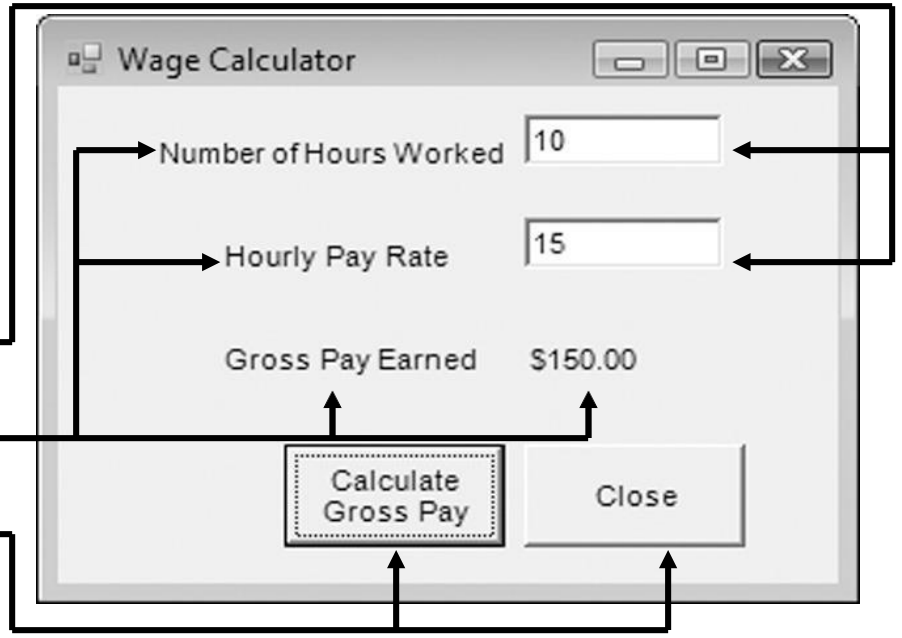
PROPERTIES WINDOW

- The properties window lists the design time properties for selected objects and their current settings.
- A property is a characteristic of an object such as size, caption or color etc.
- View-> properties window and shortcut key is f4



1.2 Example of an Object

- Form elements are objects called *controls*
- This form has:
  - Two *TextBox* controls
  - Four *Label* controls
  - Two *Button* controls
- The value displayed by a control is held in the *text property* of the control
- Left button text property is *Calculate Gross Pay*
- Buttons have methods attached to *click events*

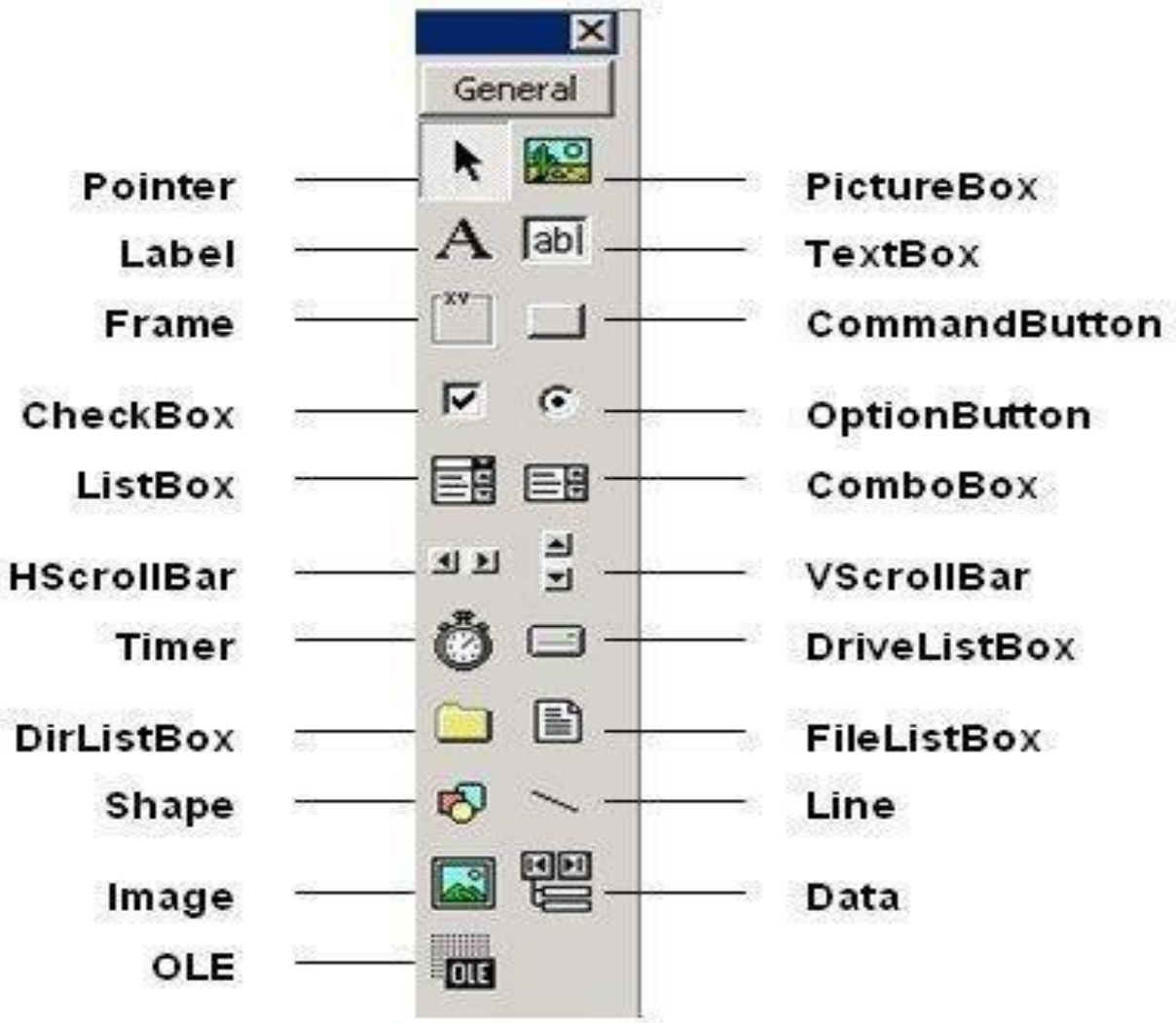


Event Driven Programming: Events

- The GUI environment is *event-driven*
- An event is an action that takes place within a program
  - Clicking a button (a Click event)
  - Keying in a TextBox (a TextChanged event)
- Visual Basic controls are capable of detecting many events
- A program can respond to an event if the programmer writes an *event procedure*

VISUAL BASIC

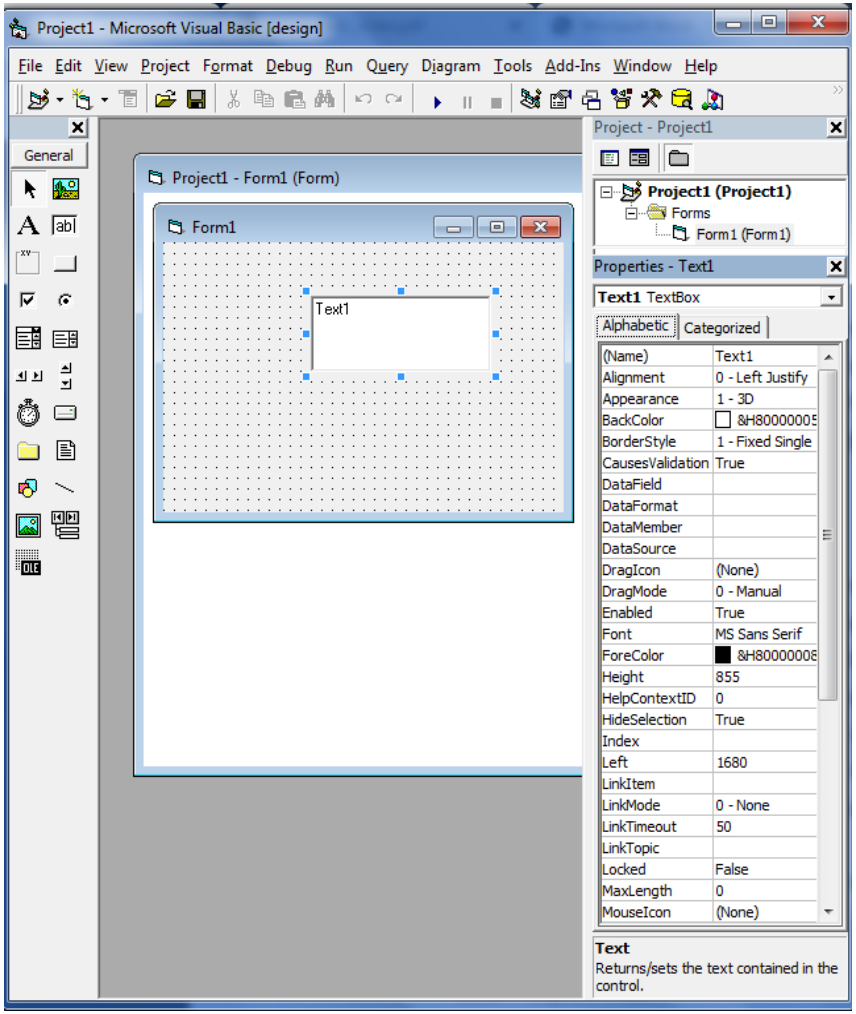
1.3 Toolbox



1.4 Handling some of the common controls

VISUAL BASIC

a) **The text box** is the standard control that is used to
   - ✓ Receive input from the user
   - ✓ To display the output.
   - ✓ It can handle string (text) and numeric data
   - ✓ But not images or pictures.



- Strings in a text box can be **converted** to numeric data by using the function Val(text).

- **<u>Impartant Properties are</u>**:

  Name, text, passwordchar, datafield, multiline

b) **The Label** It is used to provide instructions and guides to the users, it can also be used to display outputs, it can display text and numeric data
One of its most important properties is Caption

c**) The Command Button** it is used to execute commands. It displays an illusion that the button is pressed when the user clicks on it. The most common event associated with the command button is the Click event, and the syntax for the procedure is

```
Private Sub Command1_Click ()
    Statements
End Sub
```

d) **The Picture Box** The Picture Box is one of the controls that used to handle graphics. You can also load the picture at runtime using the LoadPicture method.

```
Picture1.Picture=LoadPicture ("C:\VB program\Images\grape.gif“)
```

e) The Image Box It functions almost identically to the picture box. However, there is one major difference, the image in an Image Box is stretchable, which means it can be resized.

An **access key** is an underlined character in the text of a menu, menu item, or the label of a control such as a button. With an **access key**, the user can "click" a button by pressing the Alt **key** in combination with the predefined **access key**

1.5 Message Box

- A *message box* is a dialog box with a user message in a pop-up window
- The following can be specified
  - Message - text to display within the box
  - Caption - title for the top bar of the box
  - Buttons - indicates which buttons to display
  - Icon - indicates icon to display
  - DefaultButton - indicates which button corresponds to the Return Key
  - All arguments but the Message are optional
  - Use of an argument requires those before it

Example Message Box

```
MessageBox.Show("Do you wish to continue?", _
      "Please Confirm", _
      MessageBoxButtons.YesNo, _
      MessageBoxIcon.Question)
```

1.6 Grid

- In **graphic** design, a **grid** is a series of vertical and horizontal lines that are used to subdivide a page or form vertically and horizontally into margins, columns, inter-column spaces for placing the controls

Editing Tools

- Syntax highlighting - includes code folding.
- Column/Block mode editing.
- Find/Replace
- Project support.

1.7 Why Have Variables?

- A *variable* is a storage location in the computer's memory, used for holding information while the program is running
- The information that is stored in a variable may change, hence the name "variable"

Declaring Variables

- A *variable declaration* is a statement that creates a variable in memory
- The syntax is

  `Dim VariableName As DataType`
  - **`Dim`** (short for Dimension) is a keyword
  - **`VariableName`** is the programmer designated name
  - **`As`** is a keyword
  - **`DataType`** is one of many possible keywords for the type of value the variable will contain
- Example: **`Dim intLength as Integer`**

VISUAL BASIC

1.8 Data Types

Numeric Data Types

Type	Storage	Range of Values
Byte	1 byte	0 to 255
Integer	2 bytes	-32,768 to 32,767
Long	4 bytes	-2,147,483,648 to 2,147,483,648
Single	4 bytes	-3.402823E+38 to -1.401298E-45 for negative values 1.401298E-45 to 3.402823E+38 for positive values.
Double	8 bytes	-1.79769313486232e+308 to -4.94065645841247E-324 for negative values 4.94065645841247E-324 to 1.79769313486232e+308 for positive values.
Currency	8 bytes	-922,337,203,685,477.5808 to 922,337,203,685,477.5807
Decimal	12 bytes	+/- 79,228,162,514,264,337,593,543,950,335 if no decimal is use +/- 7.9228162514264337593543950335 (28 decimal places).

Nonnumeric Data Types

Data Type	Storage	Range
String(fixed length)	Length of string	1 to 65,400 characters
String(variable length)	Length + 10 bytes	0 to 2 billion characters
Date	8 bytes	January 1, 100 to December 31, 9999
Boolean	2 bytes	True or False
Object	4 bytes	Any embedded object
Variant(numeric)	16 bytes	Any value as large as Double
Variant(text)	Length+22 bytes	Same as variable-length string

1.9 Working with the String Data Type

- A *string literal* is enclosed in quotation marks
- The following code assigns the name Nazareth College to the variable *strName*

```
Dim strName as string
strName = " Nazareth College "
```

- An empty string literal can be coded as:
  - Two consecutive quotation marks

```
strName = ""
```

  - Or by the special identifier String.Empty

```
strName = String.Empty
```

VISUAL BASIC

Working with the Date Data Type

- Date data type variables can hold the date *and* time
- A date literal is enclosed within # symbols

```
startDate = #10/20/2005 6:30:00 AM#
```
or
```
startDate = #12/10/2005#
```
or
```
startTime = #21:15:02#
```

- Or can use a function to convert a string to a date

```
startDate = System.Convert.ToDateTime("12/3/2002")
```

- System.Convert.ToDateTime function is used to store a date from a text box in a date variable

```
userDate = System.Convert.ToDateTime(txtDate.text)
```