



3



مذكرة فريق الأصدقاء

PREP 3

مذكرتي
ICT

First Term

أ / ياسمين شعيب

2025



Index

CHAPTER 1: Problem Solving

Problem solving

- Problem solving stages
- Flow chart
- Simple flow chart
- The use of Branching (Decision) in Flowcharts.
- The use of Loop in Flowcharts.

Chapter II: Introduction to Visual Basic .Net

- | | |
|--|--|
| <ul style="list-style-type: none"> • The language of Visual Basic.net | <ul style="list-style-type: none"> • Programing and computer memory |
| <ul style="list-style-type: none"> • The language of Visual Basic.net and Framework.net | |
| <ul style="list-style-type: none"> • Visual Basic.net and IDE | |

Chapter III: Controls

- | | |
|---|--|
| <ul style="list-style-type: none"> • Form | <ul style="list-style-type: none"> • Button |
| <ul style="list-style-type: none"> • Label | <ul style="list-style-type: none"> • Text box |
| <ul style="list-style-type: none"> • List box | <ul style="list-style-type: none"> • Combo box |
| <ul style="list-style-type: none"> • Group box | <ul style="list-style-type: none"> • Radio button |
| <ul style="list-style-type: none"> • Check box | |

Chapter IV: Code Window

- | | |
|--|--|
| <ul style="list-style-type: none"> • Code Window. | <ul style="list-style-type: none"> • Event Handler. |
| <ul style="list-style-type: none"> • Setting the (Properties) programmatically. | |

Chapter one " Problem Solving "

□ Problem definition:

Problem is a situation that requires a solution or an objective you want to achieve through following consecutive steps sequentially.

□ Problem solving:

Problem Solving is the steps, activities, and processes to be done to reach an output or objective.

□ Problem solving stages:

(1) Problem Definition

Problem definition implies the identification of required outputs, available inputs and, arithmetic and logical operations to be executed.

(2) Algorithm Preparation

After identifying and analyzing the problem, outputs and inputs; a plan in the form of a series of successive steps is made which is called an (Algorithm),



Algorithm: – is one of the methods used to solve a problem through logically arranged procedures (Flowchart).

(3) Program Design

Having drawn a “Flowchart” to solve the problem, using a computer; we have to translate this flowchart into one of the programming language

(4) Program Testing

During writing a program we unintentionally make some mistakes e.g. writing a minus sign (-) instead of (+). We cannot detect errors unless we begin entering data to the program with previously known results; and compare the results of the current program to those of the well-known results and so you can discover the errors and correct them.

(5) Documentation

This step includes writing all steps taken for solving the problem that include: given Input, output, plan for solving the problem, drawn flowchart, programming language used for coding, instructions, date of last modification of the program and, people who contribute to the program development process. This is done to have the program documented to go back for feedback and correction. This documentation is beneficial when more than one person participates in writing or modifying the program.

□ Flowchart

It is a diagram that uses standard graphical symbols to illustrate the sequence of steps required for solving a problem or specific question.

□ Some advantages of Flowcharts:

- Facilitating the reading and understanding of the problem and illustrating to the programmer what must be done.
- Useful to explain the program to others
- helping in documenting the program in better manner, especially if the program is complicated

□ The most commonly used symbols as shown in the table:

Significance	Symbol
Terminal	
(input & output)	
(process)	
(Decision)	
(Flow lines)	

First: Simple Flowchart

Exercise (1):

Draw a flowchart for a program that will calculate the sum of two numbers entered by the user and display the result.

First: Define the problem

Output: The sum of two numbers

Input: The first number is "A" and the second number is "B"

Processing (Solution): $C = A + B$ where the result is C

Second: Algorithm Third: flowchart

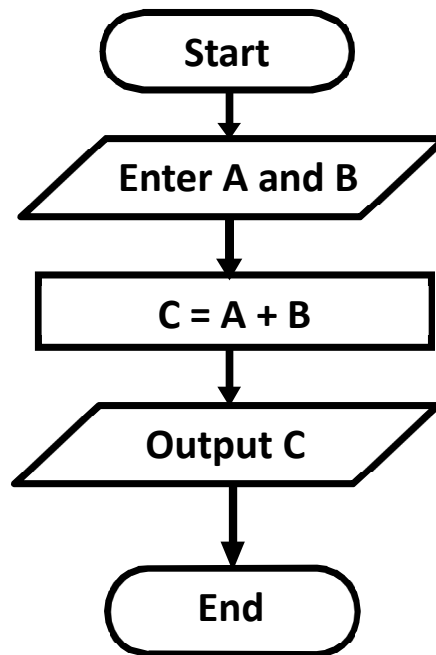
1- Start

2- Enter A and B





3- $C = A + B$

4- Output C

5- End



To construct a flowchart, we should consider the following:

1. The flowchart should start with the Start symbol and end with the End symbol. 
2. A,B,C are variable names. The variable refers to a memory storage that holds a value.
3. Equation: $C=A+B$, indicates the sum of the value of A, to the value of B, and stores the result in C.
4. Entering values in A and B is done by using the term “Enter”, inside a parallelogram, like “Read” or “Input”. 
5. The sum equation is written inside the rectangle,  as it represents an arithmetic operation.
6. The output is expressed with a parallelogram  using the term “Output”, we can also use another term like “Print” or “output”.
7. Note that flow line ↓ shows the order of an Algorithm.

Exercise (2):

Draw a flowchart to solve a first degree equation $Y=3X+2$

First: Define the problem

Output: The value of Y

Input: X

Processing (Solution): Compute the value of “Y” from the equation $Y=3x+2$

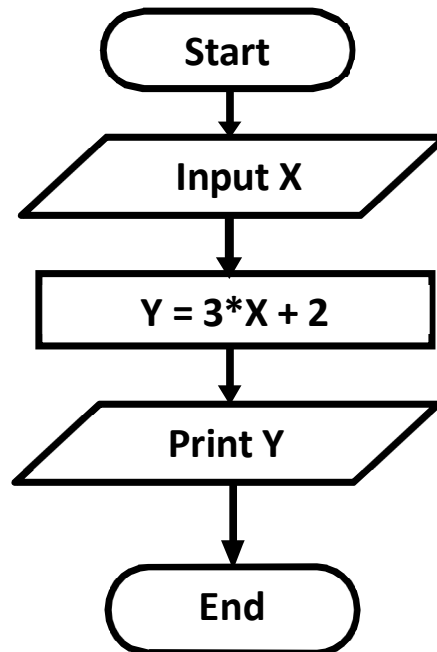
NOTE

1. The left hand side (LHS) of any equation should contain only one variable; the value of this variable will be the (output) or the solution of the equation.
2. The right hand side (RHS) of the equation may contain abstracted values or arithmetic expressions that have one or more variables (inputs).

Second: Algorithm

- 1- Start
- 2- Input X
- 3- $Y = 3 * X + 2$
- 4- Print Y
- 5- End

Third: Flowchart



Exercise (3):

Write down the algorithm, and draw a flowchart to compute the area and the perimeter of a rectangle whose length (L) and width (W) are known, bearing in mind that the equation of the area is: $\text{Area} = L * W$ and that of the perimeter is: $\text{Perimeter} = 2 * (L + W)$.

مذكرتي
Mozkrty.com

First: Define the problem

Output: Area and Perimeter

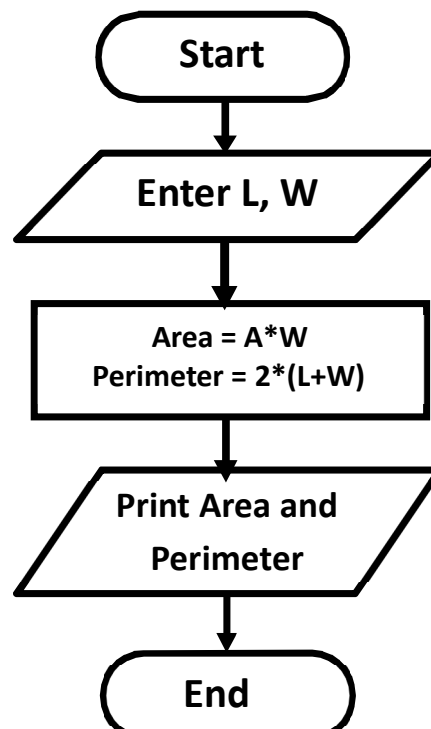
Input: L, W

Solution: $\text{Area} = L * W$, $\text{Perimeter} = 2 * (L + W)$

Second: Algorithm

- 1- Start
- 2- Enter L, W
- 3- $\text{Area} = A * W$
 $\text{Perimeter} = 2 * (L + W)$
- 4- Print Area and Perimeter
- 5- End

Third: Flowchart



Exercise (4):

Write down the algorithm, and draw a flowchart to compute the area of a circle whose radius R is known, bearing in mind that the equation of the area of circle is:

$$\text{Area} = 3.14 * R * R.$$

First: Define the problem

Output: Area

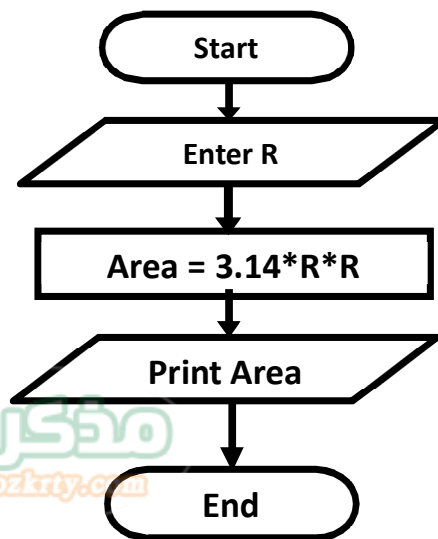
Input: R

Solution: $\text{Area} = 3.14 * R * R.$

Second: Algorithm

- 1- Start
- 2- Enter R
- 3- $\text{Area} = 3.14 * R * R$
- 4- Print Area
- 5- End

Third: Flowchart



Exercise (5):

Write down the algorithm, and draw a flowchart to calculate the number of years, bearing in mind that the number of months is known.

First: Define the problem

Output: Years

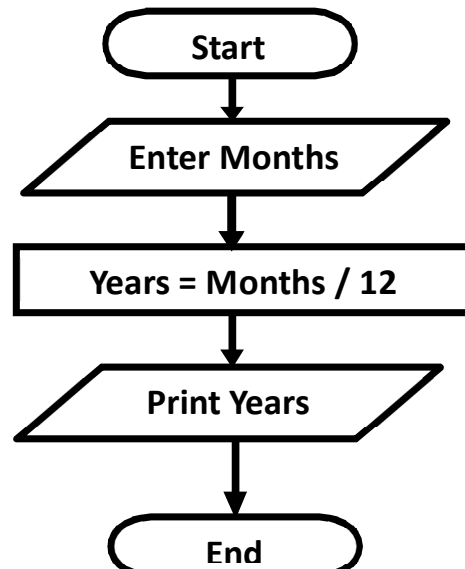
Input: Months

Second: Algorithm

- 1- Start
- 2- Enter Months
- 3- $\text{Years} = \text{Months} / 12$
- 4- Print Years
- 5- End

Solution: $\text{Years} = \text{Months} / 12$

Third: flowchart



Second: The use of Branching (Decision) in flowcharts

There are many problems that contain a question requires a Yes or No, or requires branching to other processes determined by flowchart

Exercise (1):

Draw a flowchart to print the word "successful" In the case of the degree input is greater than or equal to 50?

First: Define the problem

Output: print the word Successful

Input: the score X

Solution: If the value of X is greater than or equal 50; the word "Successful" will be printed

Second: Algorithm Third: flowchart

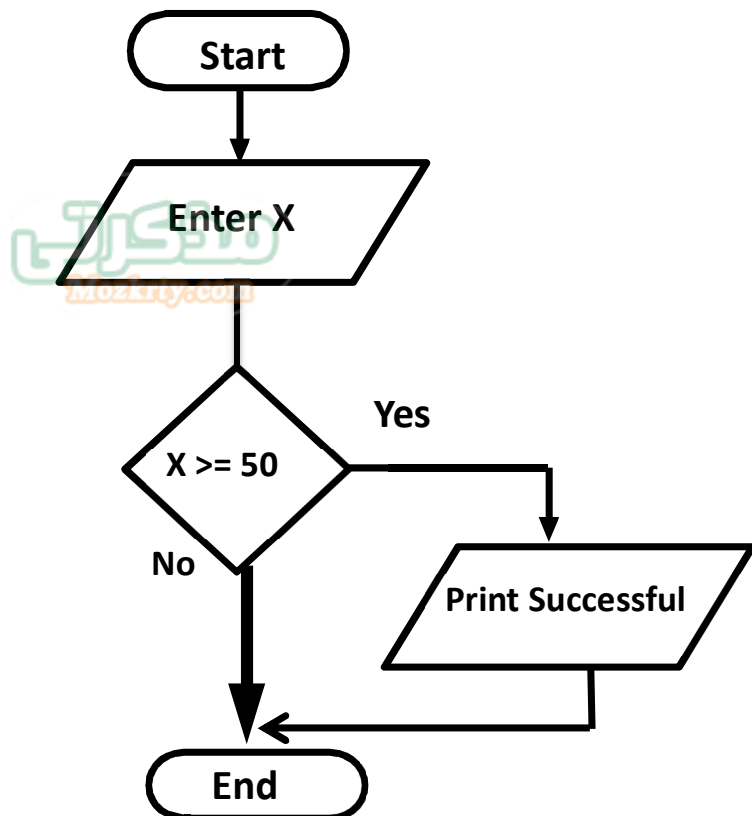
1- Start

2- Enter X

3- If $X \geq 50$ Then

4-1 Print Successful

5- End



Exercise (2):

Draw a flowchart for a program that will calculate the division of two numbers. if the divisor equals (zero), the message displays "undefined".

First: Define the problem

Output: Print the result of dividing two number "R" or print the word "undefined"

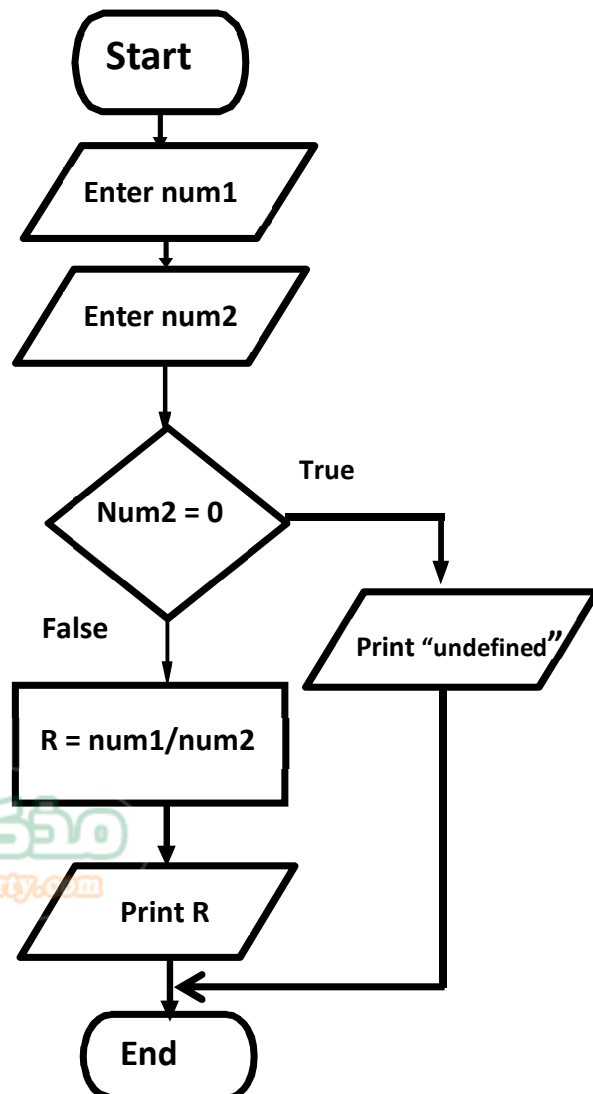
Input: Numerator is "num1", denominator is "num2".

Solution: if $num2=0$ then print "undefined", otherwise print the result of the division "R".

Second: Algorithm

- 1- Start
- 2- Enter the Numerator num1
- 3- If num2 = 0 Then
- 3- Print "undefined"
- 4- Go to step 7
- 5- $R = \text{num1} / \text{num2}$
- 6- Print R
- 7- End

Third: flowchart



Exercise (3):

Draw a flowchart for a program that obtains a number from the user. Determine the number type (even or odd) and print the result?

First: Define the problem

Output: print the number type (even or odd).

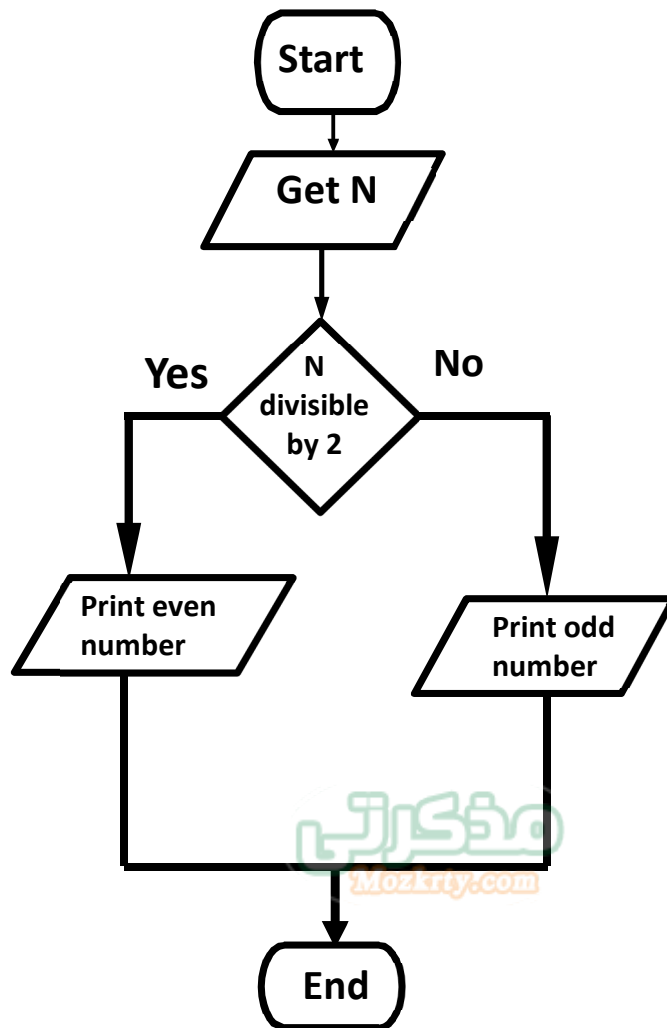
Input: the number "N".

Solution: The even number is determined if the entered number is divisible by 2 without remainder, otherwise it will be odd

Second: Algorithm

- 1- Start
- 2- Enter N
- 3- If N is divisible by 2 without remainder then
- 3-1 Print "even number"
- 4- Else
- 4-1 Print "odd number"
- 5- End

Third: flowchart



Exercise (4):

Draw a flowchart to Get a temperature degree, and print out the following result "greater than zero" – "less than zero" – "equal zero"

First: Define the problem

Output: Print out "greater than zero" – "less than zero" or "equal zero".

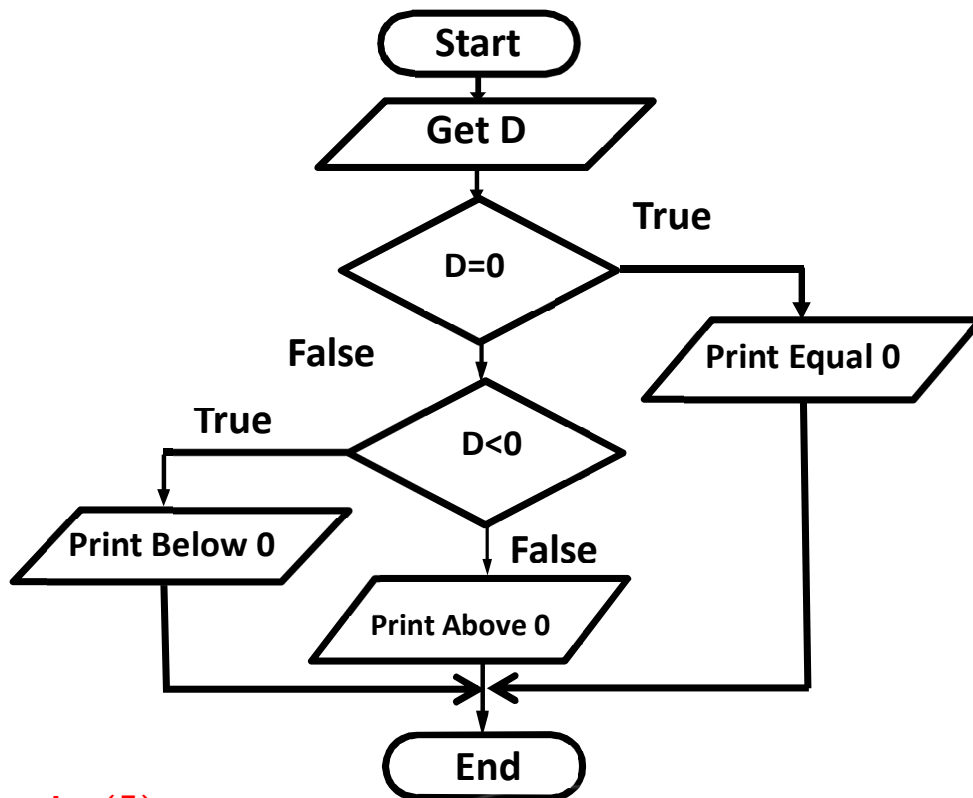
Input: Temperature degree Celsius "D".

Solution: The temperature degree entered will be compared to zero

Second: Algorithm

- 1- Start
- 2- Enter D
- 3- If $D = 0$ Then
 - 3-1 Print Equal 0
- 4- Else
 - 4-1 If $D < 0$ Then
 - 4-1-1 Print Below 0
 - 4-2 Else
 - 4-2-1 Print Above 0
- 5- End

Third: flowchart



Exercise (5):

Write down the Algorithm, and draw a flowchart to enter two unequal numbers, then print "the largest is?" and, "the smallest is.....?"

First: Define the problem

Output: print out "the largest number is ...?" – "the smaller number is?"

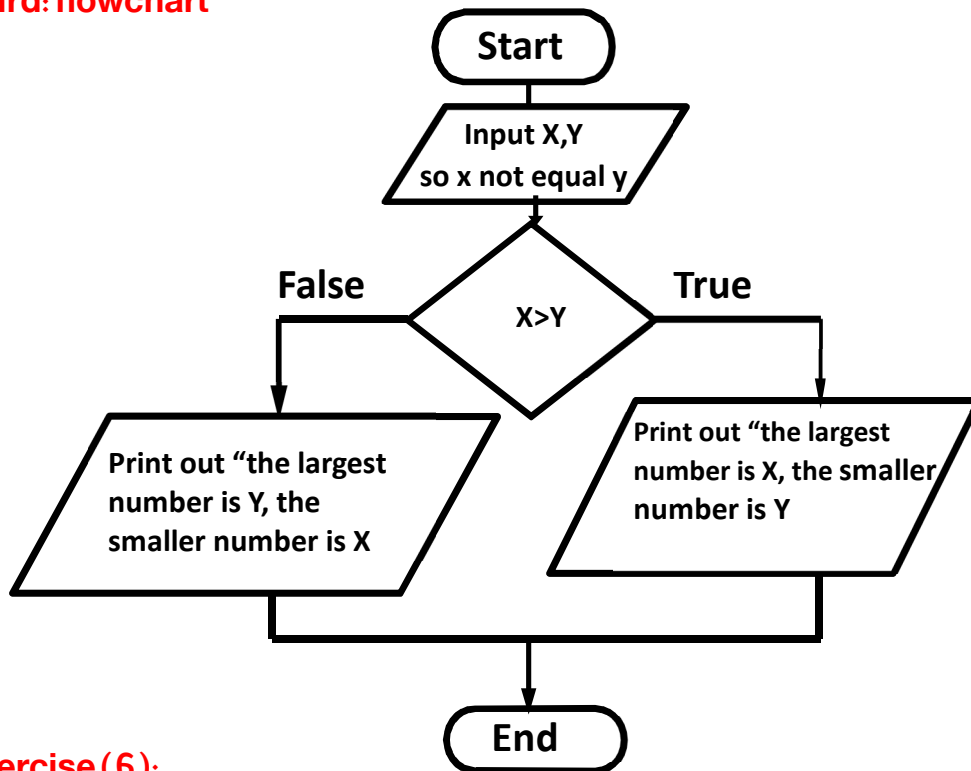
Input: X, Y so x not equally

Solution: the X will be compared to Y

Second: Algorithm

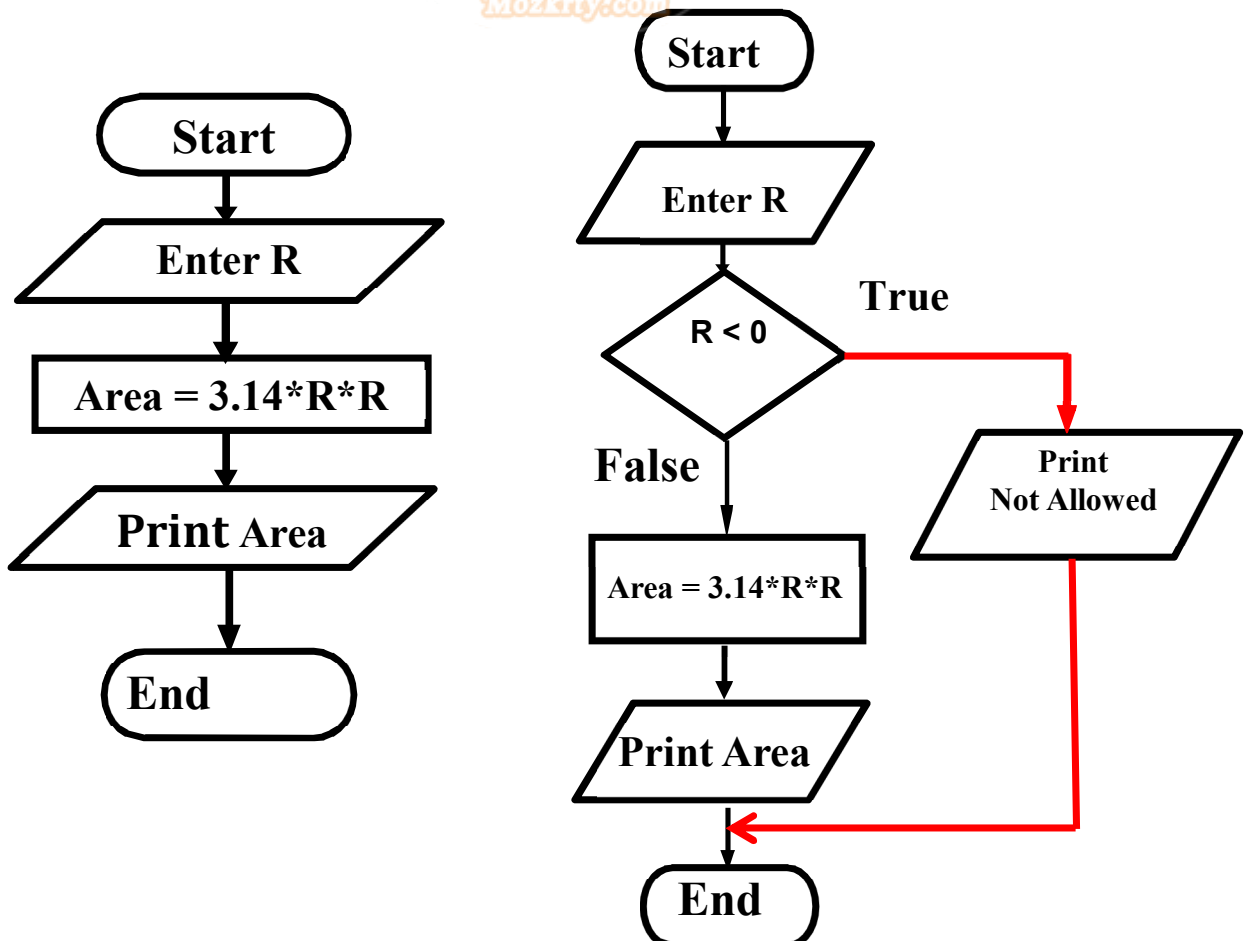
- 1- Start
- 2- Input X, Y so x not equaly
- 3- If $X > Y$ Then
 - 3-1 Print out "the largest number is X, the smaller number is Y"
- 4- Else
 - 4-1 Print out "the largest number is X, the smaller number is Y"
- 5- End

Third: flowchart



Exercise (6):

The following flowchart is used to calculate the Area of a circle with radius "R". Redraw the flowchart so that it displays the message "not allowed" and exits from the program (when the value of "R" is negative)



Third : The use of iterative loops in flowcharts:

Exercise (1):

Print out the numbers from 1 to 3

First: Define the problem

Output: print numbers from 1 to 3

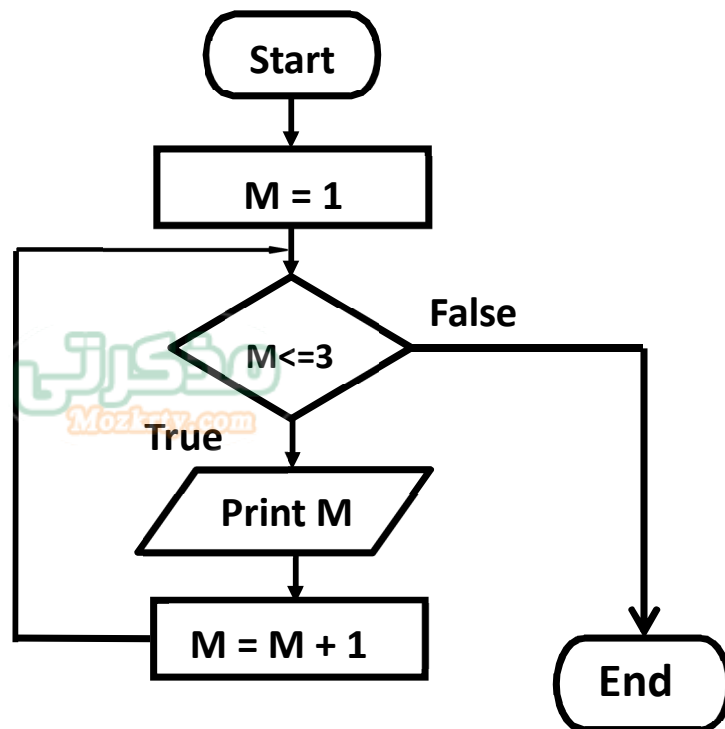
Input: number M

Solution: print number M and increment it by 1 then continue printing until the value of M become greater than 3

Second: Algorithm

- 1- Start
- 2- M=1
- 3- If $M \leq 3$ Then
 - 3-1 Print M
 - 3-2 $M = M + 1$
 - 3-3 Go To step (3)
- 4- End

Third: flowchart



In the previous flowchart complete:

- How many times was the content of the loop executed ...3....
- What will be the value of M after the end of the loop.....4.....

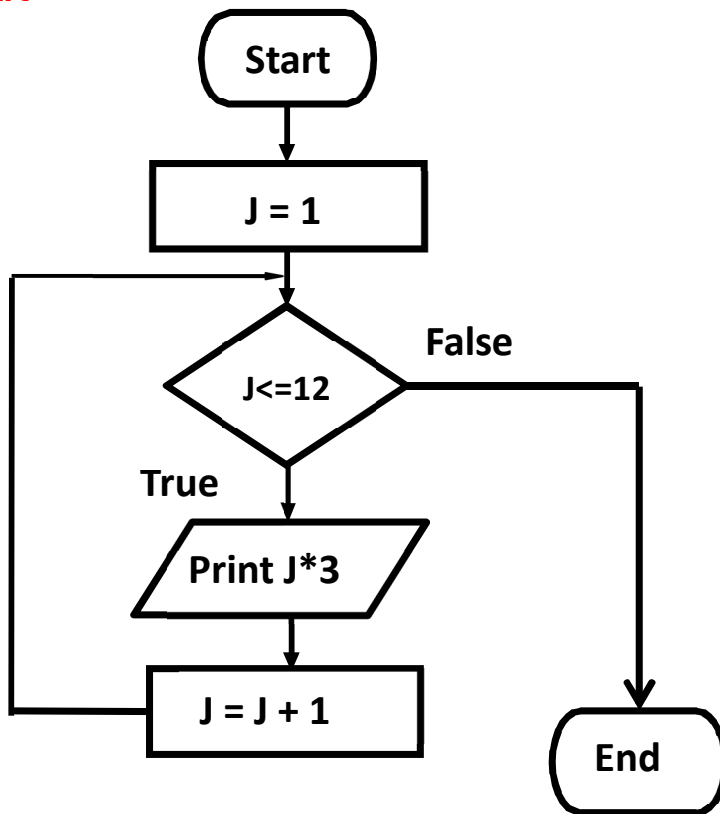
Exercise (2):

Modify the flowchart of the previous exercise to print the multiplication table of No. 3

Second: Algorithm

- 1- Start
- 2- J=1
- 3- If $J \leq 3$ Then
 - 3-1 Print $J * 3$
 - 3-2 $J = J + 1$
 - 3-3 Go To step (3)
- 4- End

Third: flowchart



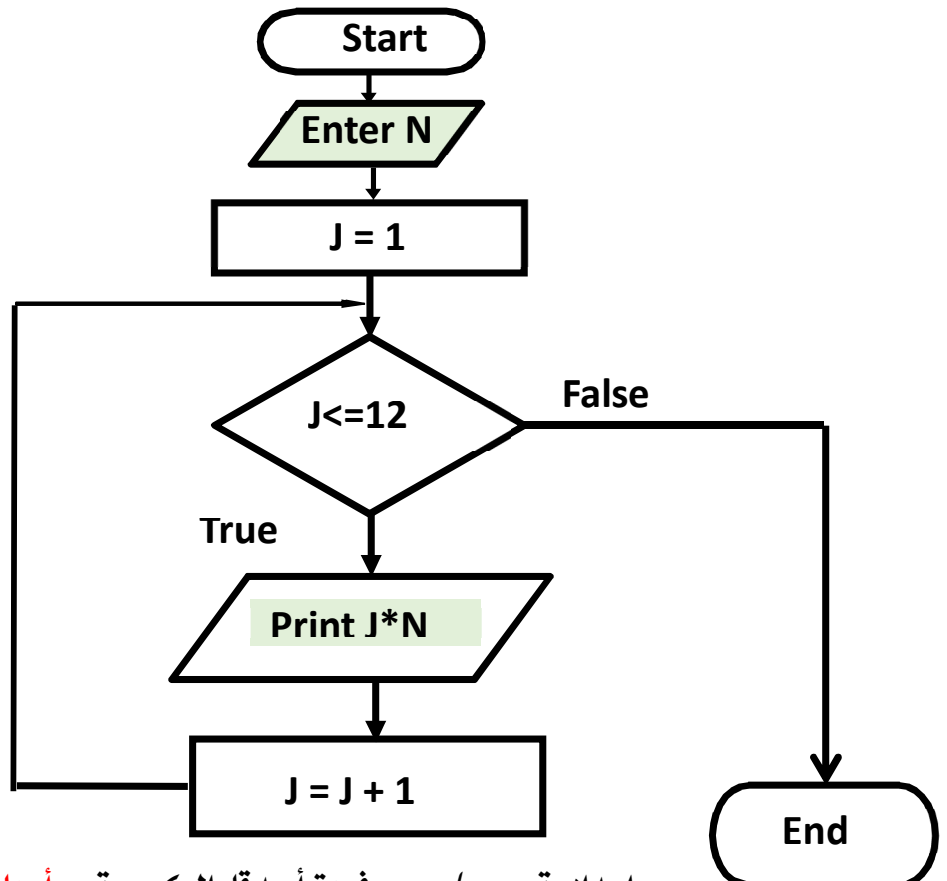
Exercise (3):

Print the multiplication table of any number?

Second: Algorithm

- 1- Start
- 2- Enter N
- 3- J=1
- 4- If J<=3 Then
 - 4-1 Print J*N
 - 4-2 J=J+1
 - 4-3 Go To step (4)
- 5- End

Third: flowchart



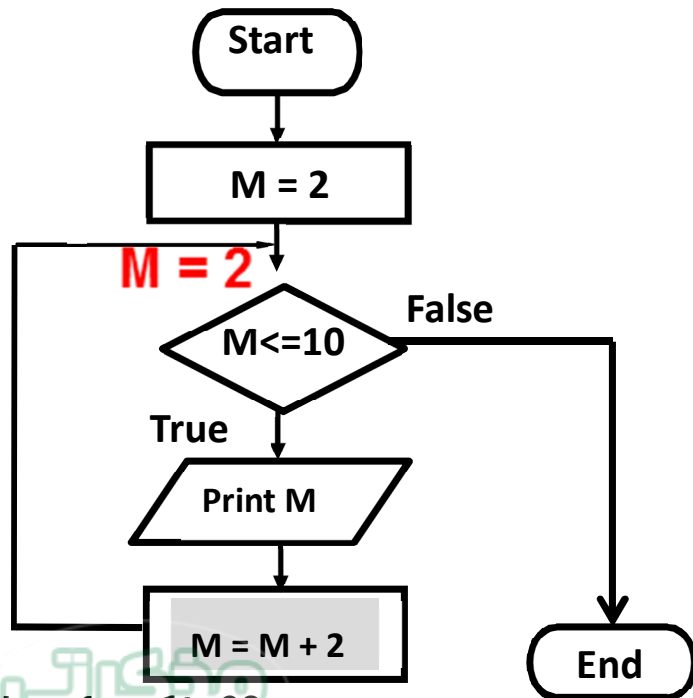
Exercise (4):

Write down the Algorithm and draw a flowchart to print out even numbers from 2 to 10

Second: Algorithm

- 1- Start
- 2- $M=2$
- 3- If $M \leq 10$ Then
 - 3-1 Print M
 - 3-2 $M=M+2$
 - 3-3 Go To step (3)
- 4- End

Third: flowchart



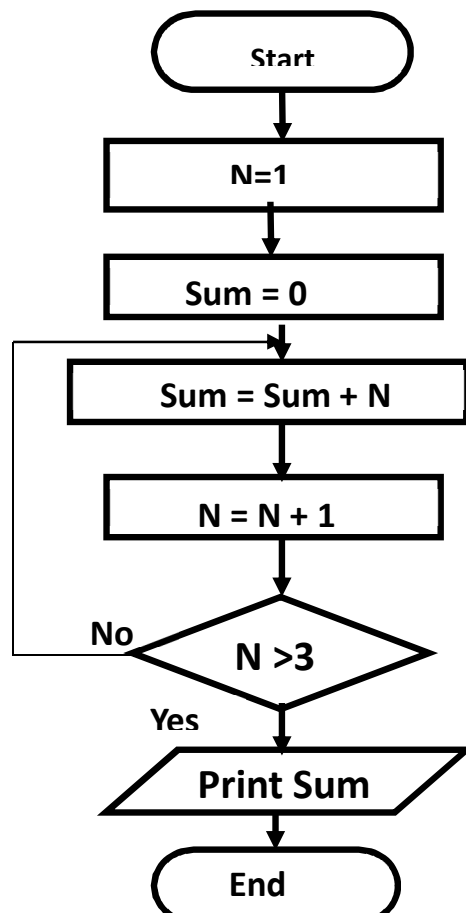
Exercise (5):

Print out the sum of integer numbers from 1 to 3?

Second: Algorithm

- 1- Start
- 2- $N = 1$
- 3- $Sum = 0$
- 4- $Sum = Sum + N$
- 5- $N = N + 1$
- 6- If $N > 3$ Then
 - 6-1 Print Sum
- 7- Else
 - 7-1 Go to step 4
- 8- End

Third: flowchart



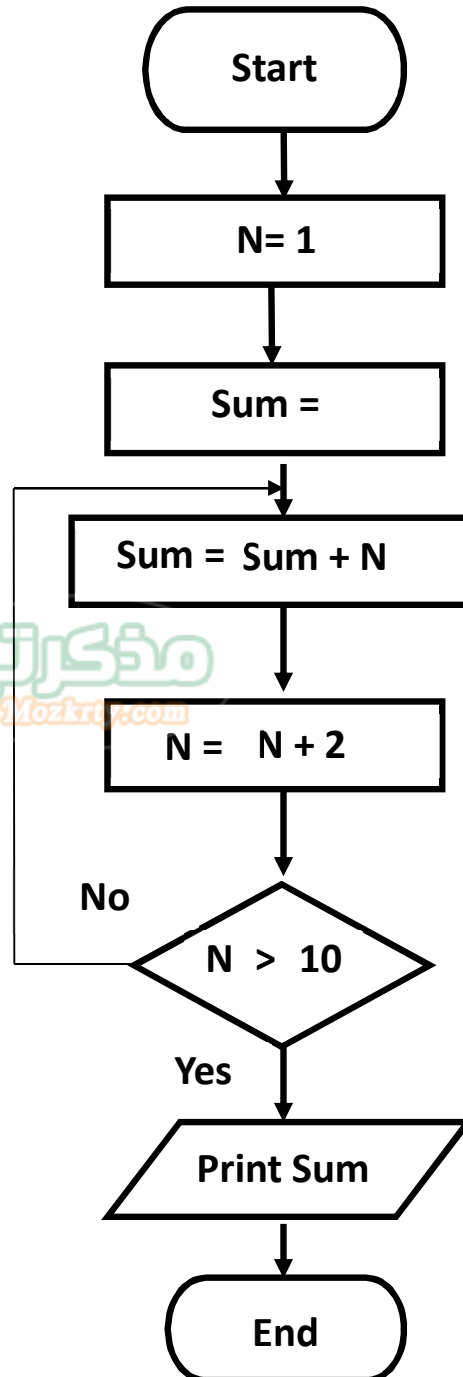
Exercise (6):

Redraw the flowchart of the previous exercise in your notebook after modifying it to print out the sum of odd numbers from 1 to 10

Second: Algorithm

- 1- Start
- 2- $N = 1$
- 3- $Sum = 0$
- 4- $Sum = Sum + N$
- 5- $N = N + 2$
- 6- If $N > 10$ Then
 - 6-1 Print Sum
- 7- Else
 - 7-1 Go to step 4
- 8- End

Third: flowchart



Questions

A. Choose the appropriate answer to complete each phrase of the following:

1. Steps, activities and procedures to be done to reach an objective or an output – can be called:
 - a. problem definition
 - b. problem
 - c. problem solving
2. On drawing flowcharts we use:
 - a. standard symbols and lines
 - b. all geometric figures
 - c. one geometric figure
3. A set of procedures arranged logically for solving a specific problem – can be called:
 - a. problem
 - b. algorithm
 - c. program testing
4. Making sure that the program is free of errors – can be called:
 - a. program testing
 - b. program documentation
 - c. algorithm
5. Writing down all the steps taken to solve a problem errors – can be called:
 - a. program documentation
 - b. program testing
 - c. flowcharts
6. A schematic representation that depends on drawing some standard symbols to clarify the order of procedures to solve a problem can be called:
 - a. Problem
 - b. algorithm
 - c. flowchart

Chapter Two: Introduction to Visual Basic.net

□ The language of visual Basic .net:

It is one of the high level programming languages and designed to be easy to learn as its commands and instructions use English language vocabulary and it can be used in many applications such as:

- 1- Windows applications
- 2- Web applications

□ Programming and computer memory:

Commands and instructions which are written in Visual Basic.net enable you to create objects in computer memory and every object has:

- 1- **Properties** such as (size-color- font) of the text written on the program interface.
- 2- **Events** such as click on a command button.
- 3- **Procedures**, each one contains commands and instructions which are carried out when calling this procedure.

So, the Visual Basic.net is considered:

- ✓ **Object oriented** as its programs work through objects in computer Memory.
- ✓ **Event Driven** as commands and instructions are carried out as soon as certain event occurs.

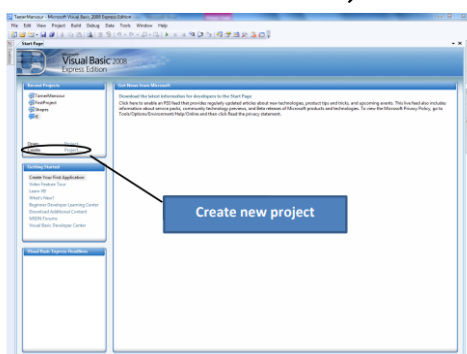
The language of visual basic .net and framework.Net:

The Framework.Net provides the following:

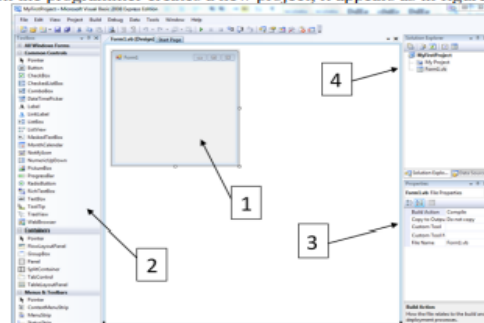
- * **Libraries** through which we create the objects.
- * **Runtime** environment (called Runtime) in computer memory where Applications produced by the language of Visual Basic.net language work in.
- * **Compilers** which compile commands and instructions written in Programming language into machine code which the Computer deals with.

Visual Basic .Net IDE:

The programmer of Visual Basic.net needs Integrated Development Environment (IDE) which provides tools and merits to the programmer that help him create applications (windows – mobile – web.....). Visual Studio represents IDE.

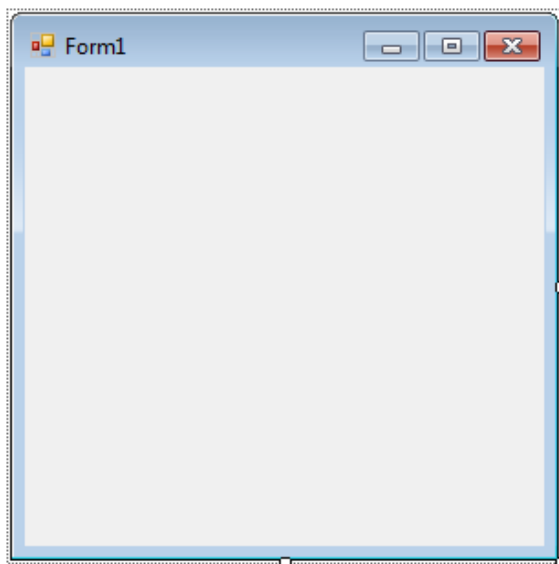


When the programmer creates a new project, it appears as in figure (2-2)



1- Form:

The form is the interface which the user deals with through different controls such as Button, Textbox, label.....etc.



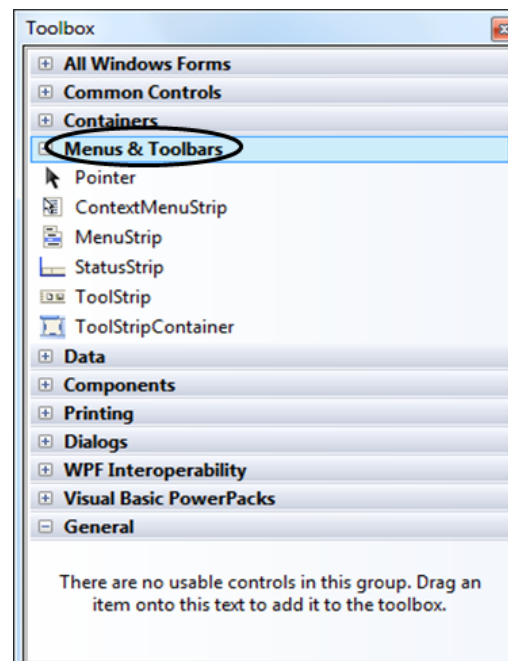
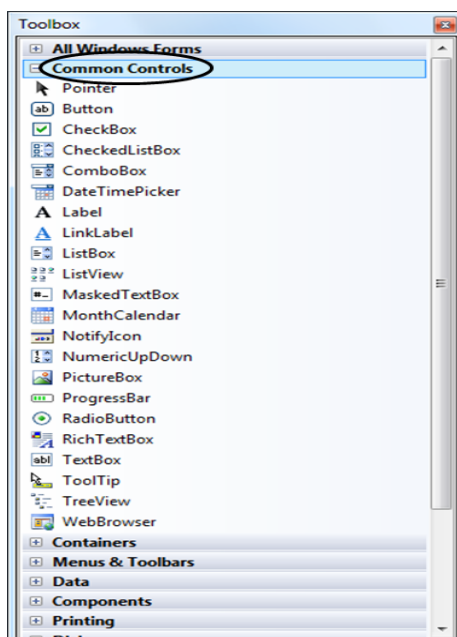
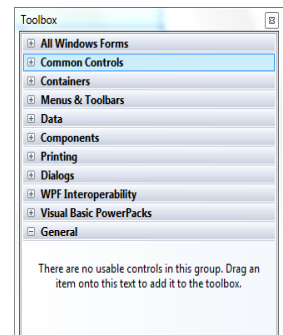
Form window before putting controls Form window after putting controls

2- Toolbox Window:

It contains tools of controls which can be put on the Form and can be shown in categories in the following Figure

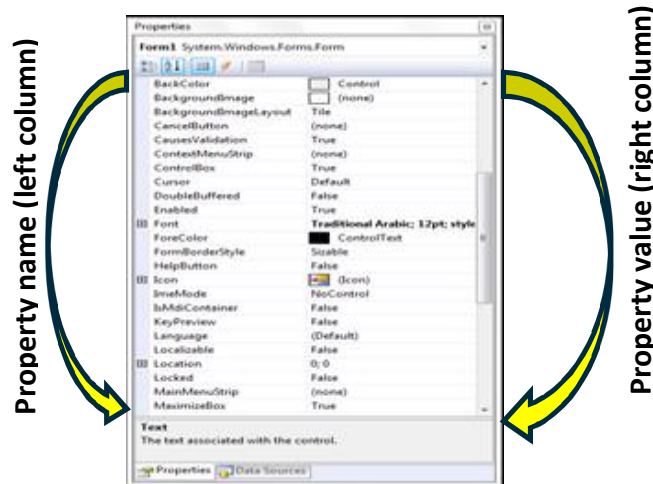
Some of these categories include:

1. Common Controls
2. Menus & Toolbars



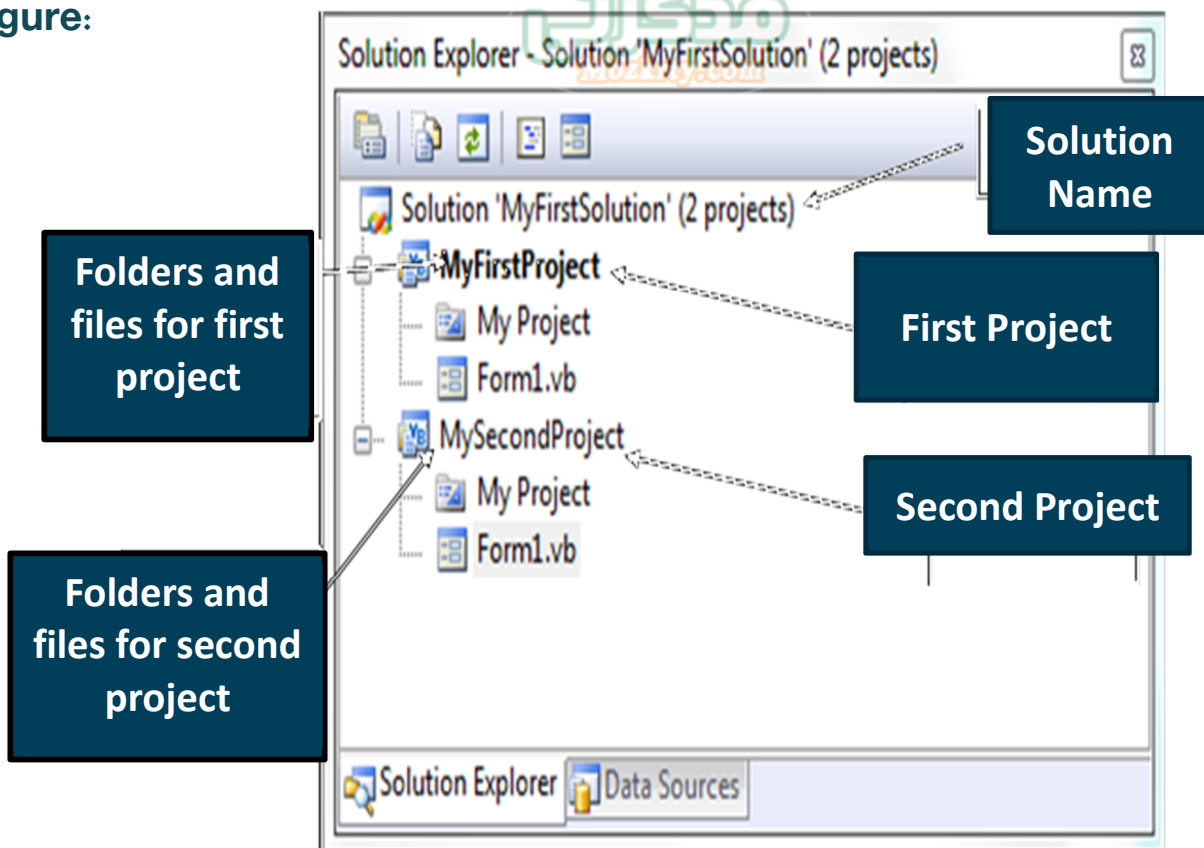
3- Properties Window

Each tool of the above Common Controls has a group of properties which can be adjusted through "Properties Window" as shown in this Figure:



4- Solution Explorer

There is a list of folders and files of the projects in this part as shown in this Figure:



Questions

Choose the appropriate answer to complete each phrase of the following:

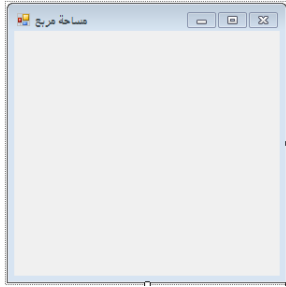
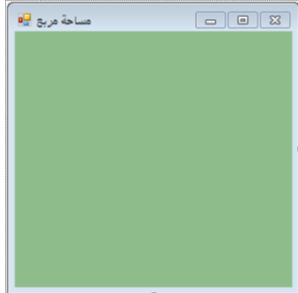
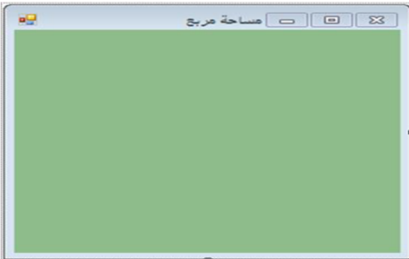
1. Object Oriented programming language depends on:
 - a- Using Windows applications
 - b- Using Web applications
 - c- Objects in computer memory
2. You can produce Windows applications or Web applications by using:
 - a- Objects in computer memory
 - b- VB.net language
 - c- Properties and Events
3. Characteristics which describe the object such as size, name and color are called:
 - a- Properties
 - b- procedure
 - c- Events
4. Click on Button is:
 - a- property
 - b- procedure
 - c- Event
5. Commands and instructions which we want to carry out are called:
 - a- properties
 - b- producers
 - c- Events
6. The Properties term refers to:
 - a- Features that describe the object
 - b- Events that can occur to the object
 - c- Commands and instructions that are carried out
7. The Events term refers to:
 - a- Properties that describe the object.
 - b- Events that can occur to the object.
 - c- Commands and instructions that are carried out.
8. The Procedures term refers to:
 - a- Properties that describe the object.
 - b- Events that can occur to the object.
 - c- Commands and instructions that are carried out.
9. Libraries, Compilers and Environment of runtime of applications are the most important components of:
 - a- Object Oriented.
 - b- Event Driven.
 - c- Framework.net.
10. IDE is called:
 - a- Visual Basic.net
 - b- Visual Studio.
 - c- Framwork.net

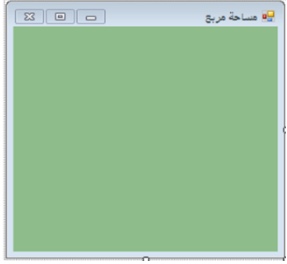
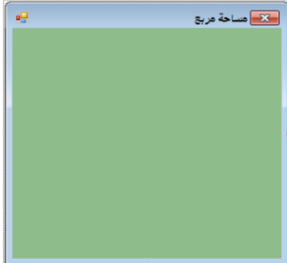

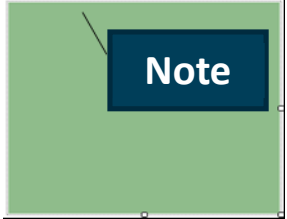
Chapter Three: Controls

□□ Form

The Form has many properties which share in defining the form of program screen we want to create.

Here are the effects of setting the previous properties on the Form:

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Name	frmSquare	In design mode	The value of Name property doesn't appear on the Form because it is used as Form Name in code window.
Text	مساحة المربع	In design mode and runtime mode	
Color	DarkSeaGreen	In design mode and runtime mode	
RightToLeft	Yes	In design mode and runtime mode	

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
RightToLeftLayout	True	In design mode and runtime mode	
MaximizeBox MinimizeBox	False False	In design mode and runtime mode	
ControlBox	False	In design mode and runtime mode	
FormBorderStyle	None	In design mode and runtime mode	
WindowState	Maximized	In runtime mode	The Form is Full screen

NOTES:

1. There are common properties among different Controls such as (**Name-Text – Forecolor – Backcolor – Right To Left.....etc.**)
2. There are some properties which their effect doesn't appear on Controls until you set some other properties , for example the **Right to Left Layout** property doesn't work unless the value **Right to Left** equals Yes
3. There are properties of the form, if they are set, they are applied to Controls which are placed on this Form such as font and **Forecolor** properties
4. The default value of the property (**Text**) and the property (**Name**) is the same and it is (Form1)
5. When you set some properties, the effect of setting appears directly on the Form in design mode of the program
6. There are some properties which their effect doesn't appear on the Form or Controls until you run the program
7. The other Controls may have the same properties mentioned above, so they won't be repeated again when we studying these other controls once more.



Here are some properties of the Form:

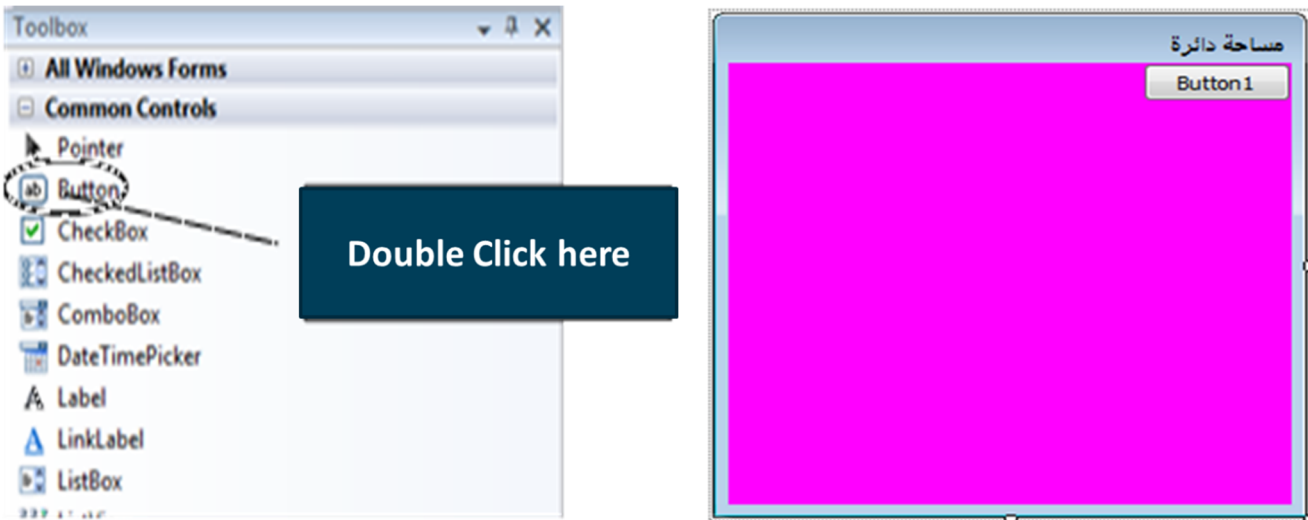
N	Property Name	Function
1	Name	Name of Form used in Code Window
2	Text	The appeared Text on the title bar of the Window
3	BackColor	The background color of the Form.
4	Right to Left	The direction of Controls on the form Window From Right to Left.
5	Right to Left layout	The layout of Controls on the Form from right to left
6	MinimizeBox	It controls the appearance or disappearance of MinimizeBox of Form Window
7	MaximizeBox	It controls the appearance or disappearance of MaximizeBox of Form Window
8	ControlBox	It controls the appearance or disappearance of ControlBox of Form Window
9	FormBorderStyle	The Border style of Form Window
10	WindowState	It defines the Window State of the Form (Maximizing, Minimizing or normal)

□□ Button

It is one of Controls which can be placed on the Form. When you click it, it does a certain task.

Placing the Button on the form:

By double clicking on the Button control on Toolbox, the Button appears on the Form in Design mode as it is shown in the following Figures:



Some distinctive properties of Command Button:

N	Property Name	Function
1	Location	The location of placing Button on the Form.
2	Size	Defining the height and width of Button on the Form.
3	Text	The appeared Text on the Button
4	BackColor	Choosing the backColor of the Button.
5	Font	Defining (shape ,size and style) of the Text font appeared on the Button.
6	ForeColor	Choosing the ForeColor to the appeared Text on the Button

NOTE: - that when you click on the command button in the design mode:

1. Eight squares appear on the borders of the Button
2. You can change the location of Button by using the process of drag and drop after moving the mouse pointer until it turns into a shape with four heads
3. You can change the size of the Button by using the process of drag and drop using the pointer of the mouse through the eight squares



Some distinctive properties of Command Button:

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Location	0,0	In design mode and runtime mode	
Location	98,108	In design mode and runtime mode	
Size	75,23	In design mode and runtime mode	
Size	121,62	In design mode and runtime mode	
BackColor Forecolor Text	Yellow Blue احسب	In design mode and runtime mode	

□□Label

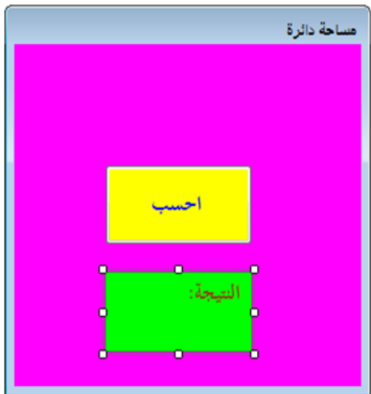
It is a tool used in showing a Text on the Form Window which can't be changed during program Runtime.



Some distinctive properties of the Label:

N	Property Name	Function
1	AutoSize	The Size of the Label is defined automatically according to the written Text if the Value of property equals true.
2	BorderStyle	Choosing the Border Style of the Label

NOTE: - You can change the size of Label manually by using the process of drag and drop when the Value of the property Auto Size is false through the eight handles in design mode only and its effect appears in design mode and runtime mode.

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Name	lblResult	In design mode	
Text ForeColor BackColor Font AutoSize BorderStyle	<u>النتيجة:</u> Choosing an appropriate color Choosing an appropriate color Choosing the Size, Style and Type of the appropriate Font False FixedSingle	In design mode and runtime mode	

□□Textbox

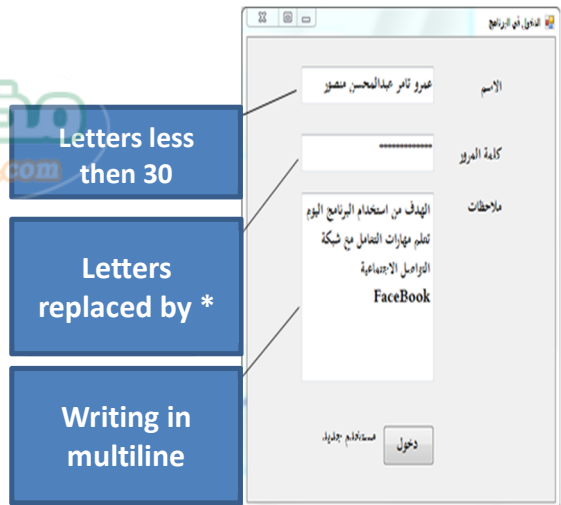
It is a tool used to insert (input) data from the user during program run time.



Some distinctive properties of the TextBox:

N	Property Name	Function
1	Maxlength	It defines the maximum number of letters which can be inserted in the TextBox
2	PasswordChar	It defines a symbol used instead of written text in case we have a password.
3	Multiline	Allows multiple lines within the text box control tool.

Here are the effects of setting some properties on TextBox:

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Maxlength PasswordChar Multiline	30 * True	In design runtime mode In design mode and runtime mode after set a value to the text property	

☐☐ ListBox

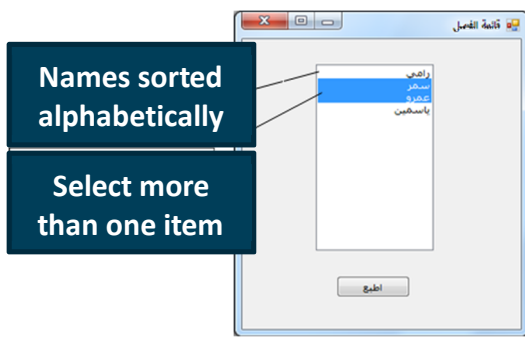
It shows a list of items.



Some distinctive properties of ListBox:

N	Property Name	Function
1	Items	A group of items shown in the ListBox
2	Sorted	It defines whether the elements in the list are sorted or not.
3	selectionMode	It defines whether it is possible to choose one item or more shown in the ListBox.

Here is the effect of setting some properties on the List Box:

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Items	عمرو رامي ياسمين سمر	In design and runtime mode	
SelectionMode	Multiextended	In runtime mode	
Sorted	True	In design and runtime mode	

□□ ComboBox

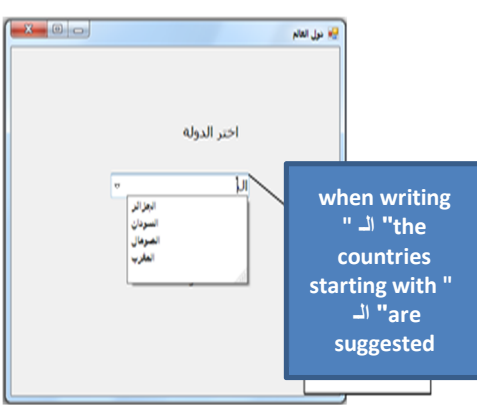
A ComboBox control displays a drop-down list from which one item can be selected.



Some distinctive properties of the ComboBox:

N	Property Name	Function
1	Items	A group of items which are shown in comboBox.
2	AutoCompleteSource	It is a source of suggested items to select from.
3	AutoCompleteMode	It defines the method of list completing process.


Here is the effect of setting some properties on the ComboBox:

Property Name	Value	Appearance mode of property effect	Form Window after setting the property
Items	مصر السودان الصومال الجزائر المغرب (Each country name in single line)	In runtime mode	
AutoCompleteMode	Suggest	In runtime mode	
AutoCompleteSource	ListItems	In runtime mode	

□□ Group Box

Is used to group other controls of same function together on the Form window.

Here is the effect of setting some properties on the GroupBox:

Property Name	Value	Form Window after setting the property
Text	النوع	
Forecolor	Choose the Right color	
RightToLeft	yes	
		In design mode and runtime mode

□□ Radio Button

The program user selects one alternative only.



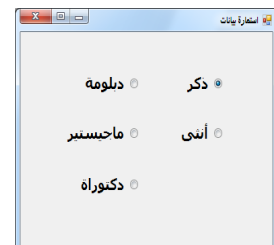
Some distinctive properties of the RadioButton:

N	Property Name	Function
1	Checked	It shows whether RadioButton has been chosen or not.
2	Text	It is the Text shown on RadioButton

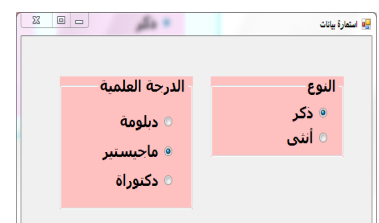
→ In case of placing a group of Radio Buttons and set the text property to each one of them in design mode as it is shown in the following Figure:



→ During program runtime, you can select one Radio Button only as it is shown in Figure:



→ You can use Group Boxes such that each group has a title and the user is allowed to choose one alternative from each group as it is shown in the following Figure:



→ One Radio Button is placed inside the Group Box by dragging the appropriate Radio Button inside the Group Box.

☐☐ Check Box

It is used for placing some alternatives to enable the user to select one Check Box or more as shown in Figure:



→ During program runtime, you can select more than one Check Box.

Questions

Choose the correct answer to complete each statement:-

- 1- The function of "Right to Left" property of the Form is:
 - a- Define the direction of Control tools from Right to Left.
 - b- Define whether the layout of ControlTools on the Form is from Right to Left.
 - c- Define the state of the window in a state of maximazing or minimaizing.
- 2- The Object TextBox is marked by one property :
 - a- AutoSize
 - b- Name
 - c- PasswordChart
- 3- The property responsible for the shape, size, and font effect of the text appearing on the command button is:
 - a- backcolor.
 - b- forecolor.
 - c- font
- 4- You can change the position of the command button on the Form window through the following operations except:
 - A- Drag and drop using the mouse.
 - B- Set the size property.
 - C- Set the Location property



- 5- You can change the position of the command button on the Form window by:
- A- Set the Location property.
 - B- Set the size property.
 - C- The eight squares around the command button.
- 6- On inserting any ControlTool by pressing D-Click from the ToolBox on the Form , the appropriate place to be shown is :
- a- Coordinate (0,0)
 - b- The middle of the Form
 - c- The position of ControlTool is different according to the size of the Form
- 7- The size of the Label widget is automatically determined on the form window if the property is:
- a. AutoSize = False
 - b. BorderStyle= FixedSingle
 - c. AutoSize=True
- 8- The size of Label is defined manually on the Form if the property is:
- a- AutoSize = False
 - b- BorderStyle= FixedSingle
 - c- AutoSize=True.
- 9- the following properties are all for the TextBox control except the property
- a- autosize
 - b- multiline
 - c- maxlenght
- 10- The right value which can be used to set the PasswordChart of the TextBox is :
- a- Pw
 - b- True
 - c- *



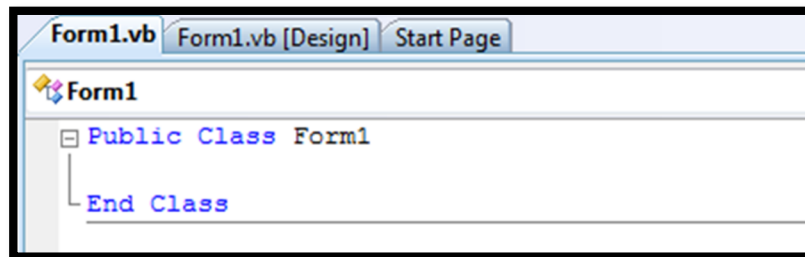
Code Window

Visual Basic.NET language provides a window through which we can write instructions and codes of the program called (code window)

To open the (Code Window) of (Form1) perform the following:

1. Make sure that the window Form is active
2. From the keyboard press (F7)

The Code window is displayed as shown in the following figure:



Code Window

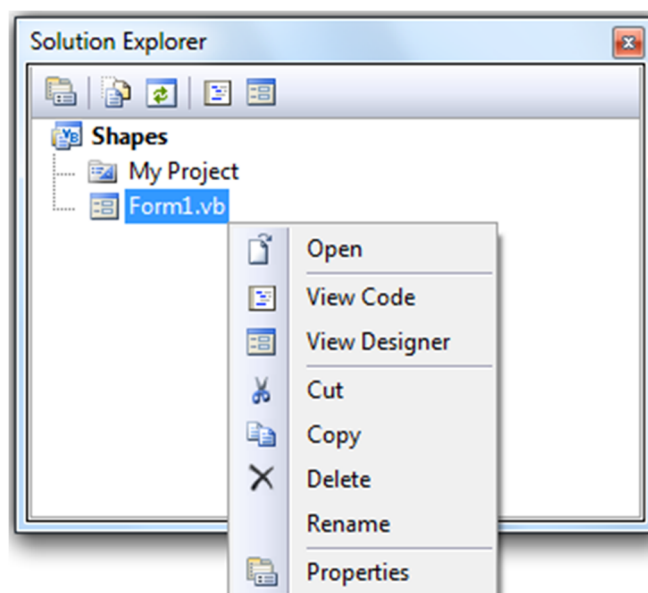
- (1) Name of the file where codes are saved
- (2) Name of the file where the Form window interface is saved
- (3) The declaration of Class; its name is (Form1)
- (4) Space between two lines; to type codes for the Class (Form1)
- (5) The end of the class (form 1)

Event Handler

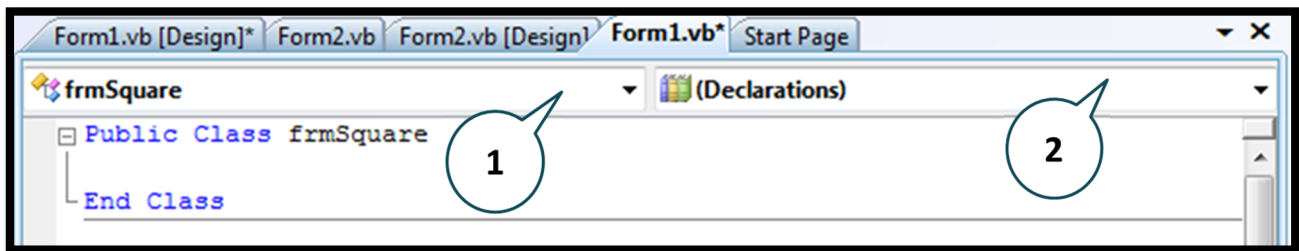
It's a procedure which contains a code that is carried out when a corresponding event occurs.

To create Event Handler do the following steps:

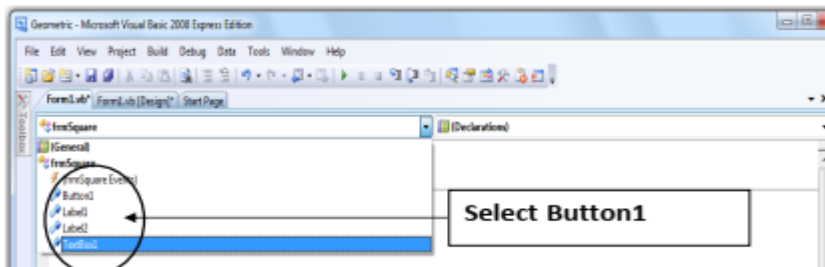
1. In the (Solution Explorer) window, right click the file (Form1.vb) and, select (View Code) from the context menu as shown in this figure:



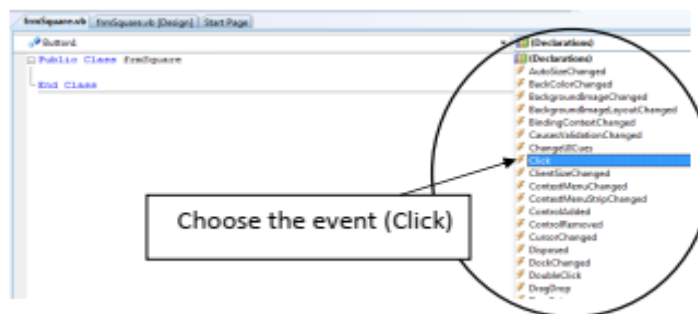
When choosing (View Code) the following figure appears:



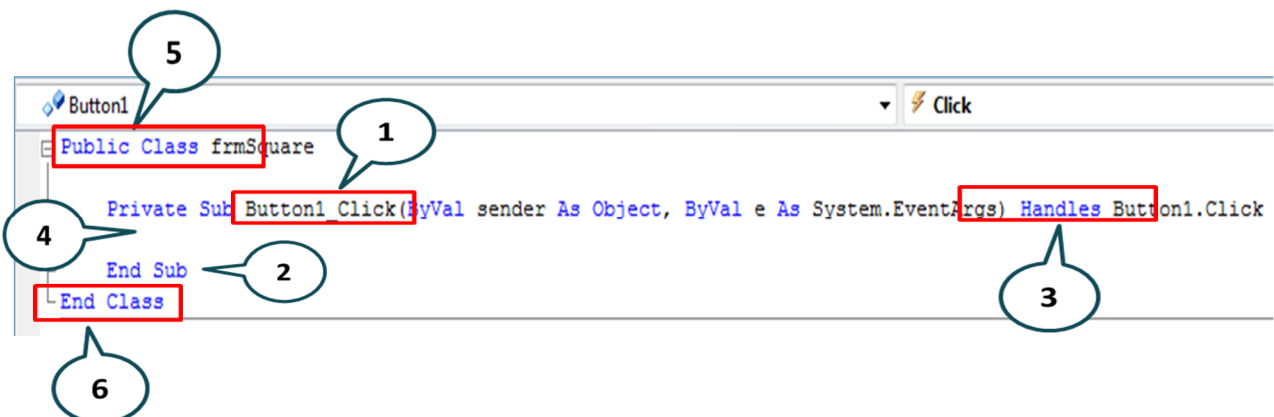
- (1) A drop-down menu of (Class Names) that displays the names of controls on the form.
- (2) A drop-down menu of (Method Names) or events; associated with the item selected from the (Class Names) menu.



2. Open (Class Name) menu and note that the default names of the controls are listed as shown in this figure.
3. When you select (Button1) from the Class menu drop down list, open (Method name) menu it displays the events associated with (Button1).
4. Choose the required event from the drop-down menu (Event handler).



Event Handler



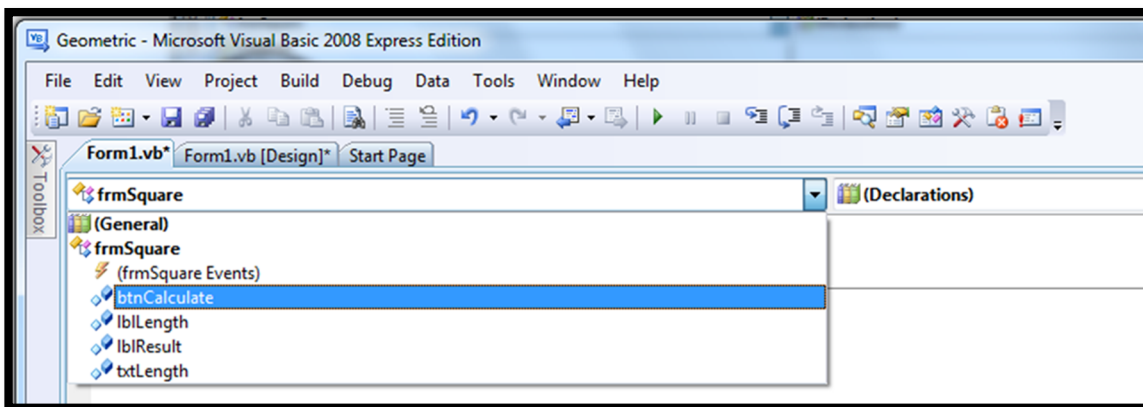
- (1) The procedure name composed of (object name, event name).
- (2) End of procedure line.
- (3))What causes the call of the procedure (event occurrence).
- (4) Between the two lines shown; the code that will be executed on calling the procedure is written after the occurrence of the (Event).
- (5) The declaration of the class line (frmSquare).
- (6) The end of (class) line.

Adjust the property (Name) for the following controls as shown in table:

No	Control	Value of the property(Name)
1	Button1	btnCalculate
2	Label1	lblLength
3	Label2	lblResult
4	TextBox1	TextBox1

➔ After setting the properties, select (**Code**) from (**view**) menu.

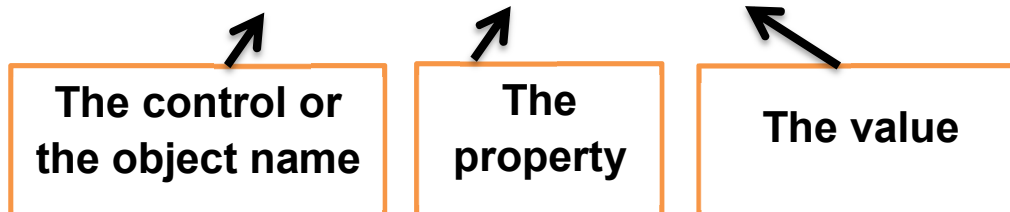
➔ Open the (**Class**) menu and note that the displayed names of the controls have been changed as shown in the following figure:



Setting the (Properties) programmatically

In chapter 3, we have already set the properties through properties window. You can adjust the properties using the following syntax:

CONTROLNAME. PROPERTY = VALUE



Setting the (Properties) programmatically

Example:

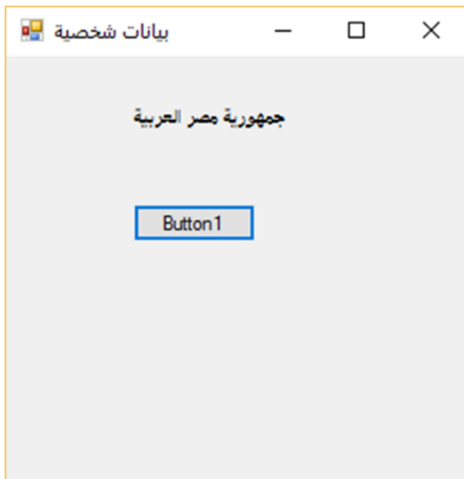
Write the following code in the appropriate event handler (Button1_Click) for button1:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
    Label1.Text = " جمهورية مصر العربية "
```

```
End Sub
```

When the event occurs, formwindow appears, as shown in a figure:

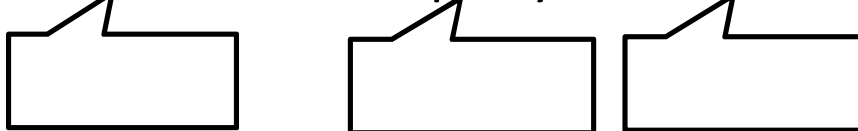


مذكرتي
Mozkrty.com

Question

1. Explain the components of the general syntax to adjust the properties of controls programmatically:

ControlName.Property = Value



2. Explain the following codes through your pervious study for the general syntax to adjust the properties of control programmatically:

(A) Button2.Text = "END"

.....

(B) Label1.AutoSize = True

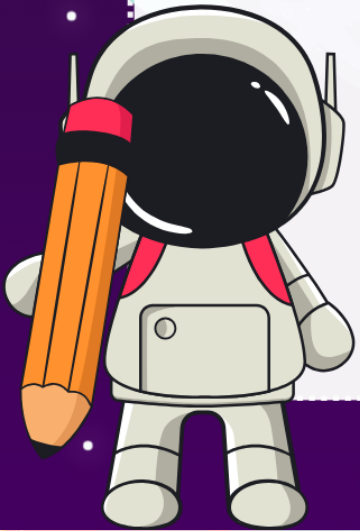
.....



Model

مذكرتي
Mozkrty.com

Answer





LESSON 1

Choose the appropriate answer to complete each phrase of the following:

- Steps, activities and procedures to be done to reach an objective or an output – can be called:
a. problem definition
b. problem
c. problem solving
- On drawing flowcharts we use:
a. standard symbols and lines
b. all geometric figures
c. one geometric figure
- A set of procedures arranged logically for solving a specific problem – can be called:
b. algorithm
c. program testing
- Making sure that the program is free of errors – can be called:
a. program testing
b. program documentation
c. algorithm
- Writing down all the steps taken to solve a problem errors – can be called:
a. program documentation
b. program testing
c. flowcharts
- A schematic representation that depends on drawing some standard symbols to clarify the order of procedures to solve a problem can be called:
a. Problem
b. algorithm
c. flowchart

LESSON 2

Choose the appropriate answer to complete each phrase of the following:

- Object Oriented programming language depends on:
d- Using Windows applications
e- Using Web applications
f- Objects in computer memory
- You can produce Windows applications or Web applications by using:
d- Objects in computer memory
e- VB.net language
f- Properties and Events
- Characteristics which describe the object such as size, name and color are called:
d- Properties
e- procedure
f- Events
- Click on Button is:
d- property



- e- procedure
 - f- **Event**
5. Commands and instructions which we want to carry out are called:
- d- properties
 - e- **producers**
 - f- Events
6. The Properties term refers to:
- d- **Features that describe the object**
 - e- Events that can occur to the object
 - f- Commands and instructions that are carried out
7. The Events term refers to:
- d- Properties that describe the object.
 - e- **Events that can occur to the object.**
 - f- Commands and instructions that are carried out.
8. The Procedures term refers to:
- d- Properties that describe the object.
 - e- Events that can occur to the object.
 - f- **Commands and instructions that are carried out.**
9. Libraries, Compilers and Environment of runtime of applications are the most important components of:
- d- Object Oriented.
 - e- Event Driven.
 - f- **Framework.net.**
10. IDE is called:
- d- Visual Basic.net
 - e- **Visual Studio.**
 - f- Framwork.net



LESSON 3

Choose the correct answer to complete each statement:-

- 1- The function of "Right to Left" property of the Form is:
- a- **Define the direction of Control tools from Right to Left.**
 - b- Define whether the layout of ControlTools on the Form is from Right to Left.
 - c- Define the state of the window in a state of maximaizing or minimaizing.
- 2- The Object TextBox is marked by one property :
- d- AutoSize
 - e- Name
 - f- **PasswordChart**
- 3- The property responsible for the shape, size, and font effect of the text appearing on the command button is:
- a- bgcolor.
 - b- forecolor.
 - c- **font**
- 4- You can change the position of the command button on the Form window through the following operations except:
- A- Drag and drop using the mouse.
 - B- **Set the size property.**

- C- Set the Location property
- 5- You can change the position of the command button on the Form window by:
- A- Set the Location property.
 - B- Set the size property.
 - C- The eight squares around the command button.
- 6- On inserting any ControlTool by pressing D-Click from the ToolBox on the Form , the appropriate place to be shown is :
- d- Coordinate (0,0)
 - e- The middle of the Form
 - f- The position of ControlTool is different according to the size of the Form
- 7- The size of the Label widget is automatically determined on the form window if the property is:
- d. AutoSize = False
 - e. BorderStyle= FixedSingle
 - f. AutoSize=True
- 8- The size of Label is defined manually on the Form if the property is:
- d- AutoSize = False
 - e- BorderStyle= FixedSingle
 - f- AutoSize=True.
- 9- the following properties are all for the TextBox control except the property
- d- autosize
 - e- multiline
 - f- maxlenght
- 10- The right value which can be used to set the PasswordChart of the TextBox is :
- d- Pw
 - e- True
 - f- *

LESSON 4

1. Explain the components of the general syntax to adjust the properties of controls programmatically:



2. Explain the following codes through your pervious study for the general syntax to adjust the properties of control programmatically:

(A) Button2.Text = "END"

..... Show the word End on the face of the button tool.....

(B) Label1.AutoSize = True

Zoom in and out of the Title widget according to the size of the text displayed in it