

Computer & ICT

Third Prep.

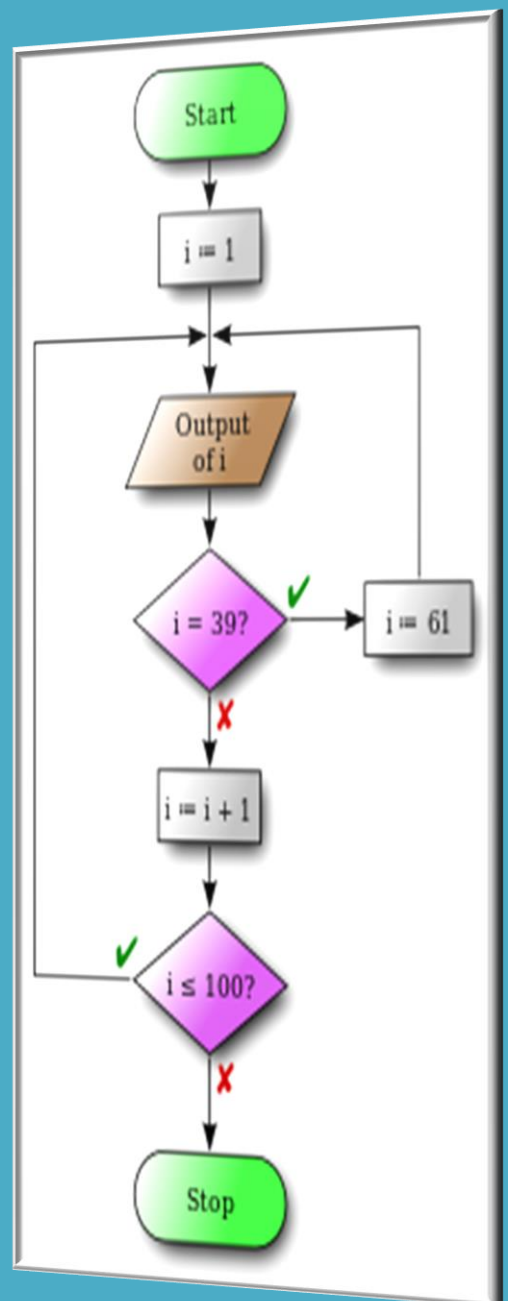


Visual Basic

By Mrs.

Ekhlās F. Mohamed

First Semester

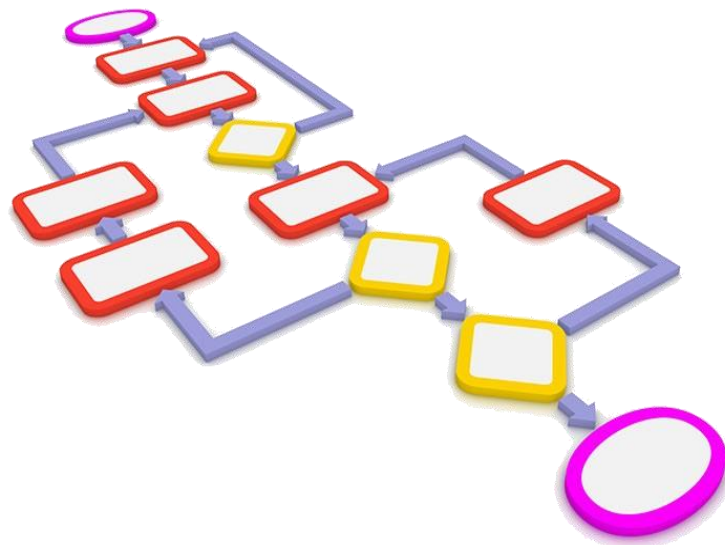


The Contents

Chapter	Subjects	Pages
Chapter One : <u>Problem Solving</u>		
	Problem & Problem Solving	1
	Problem Solving Stages.	1
	Flowchart.	2
	Simple Flowchart.	4
	The use of Branching (Decision) in Flowcharts	10
	The use of Loop in Flowcharts.	15
	Questions.	20
Chapter Two :<u>Introduction to Visual Basic .Net</u>		25
	The language of Visual Basic.net	26
	Programing and computer memory	26
	The language of Visual Basic.net and Framework.net	26
	Visual Basic.net and IDE	27
	Questions.	30
Chapter Three :<u>The Controls</u>		33
	Form.	34
	Button.	39
	Label.	43
	Textbox.	44
	ListBox.	46
	ComboBox.	47
	GroupBox.	49
	RadioButton.	50
	CheckBox.	52
	Questions.	53
Chapter Four :<u>Code Window</u>		56
	Code Window.	57
	Event Handler	57
	Setting the (Properties) programmatically.	60
	Questions.	60

Chapter One

Problem Solving



Preface

In your daily life, we may face many various problems.

The Problem:

Problem is a situation (موقف) that requires a solution or an objective you want to achieve (تحقيق), through following consecutive (متتالية) steps sequentially.

Another words: the problem means an objective or an output that we have to achieve (تحقيق), through a sequence of specific steps and activities

Problem Solving:

It is a set of steps, activities and processes (operations) to be done to reach an output or objective.

For Example: *Baking a cake* is considered a problem.

- *The inputs available:* (eggs, flour, milk ...etc)
- We have to follow *activities or procedures* that should be done *sequentially* to attain the *required objective* and get the cake

Problem Solving Stages:

1st. Problem Definition:

It's identification of required outputs, available inputs and, arithmetic or logical operations to be executed.

2nd. Algorithm Preparation:

Algorithm

It's a group of logically arranged procedures that used to solve the problem

- It is one of the methods used to solve the problem.
- It is a plan in the form of a series of successive steps.
- The algorithm is represented by drawing "Flowcharts".

The Flowchart:

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

The importance (advantages) of Flowchart:

- 1] Facilitating (تسهل) the reading and understanding of the problem and illustrating to the programmer what must be done.
- 2] Shows what tasks should be performed when writing program codes
- 3] Explains the program to others.
- 4] helping in documenting the program in better manner (بطريقة افضل), especially (خاصة) if the program is complicated (معقد)

3rd. Program Design:

After drawing a "Flowchart", to solve the problem, using a computer we translate this flowchart into one of the programming languages.

4th. Program Testing:


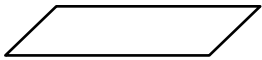
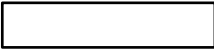
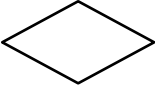

- During writing the program codes, it is possible to make some mistakes; such as; typing (-) symbol instead of (+).
- So we entering data to the program its results is known previously, to compare the results of the current program to those of the well-known results to discover the errors and debug (correct) them.

5th. Program Documentation:

- By typing all steps taken for solving the problem that include: (given Input, output, plan for solving the problem, drawn flowchart, programming language used etc)
- This is done to have the program documented to go back for feedback and correction.
- The documentation is beneficial when more than one person participate in writing or modifying the program.



The most commonly used symbols in drawing the Flowchart:

Significance	Symbol
Terminal (Start , End)	
Input / Output	
Process	
Decision	
Flow Lines	

Simple Flowcharts:

Example (1)

Draw a flowchart to calculate the sum of two numbers entered and display the result.

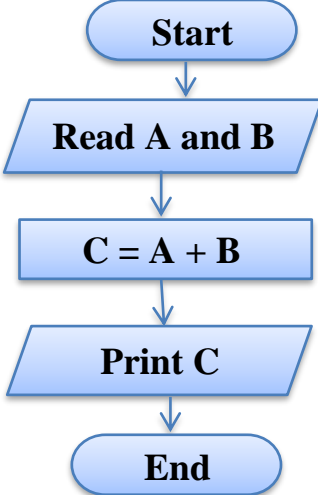
The answer:

First: Define the problem


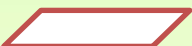
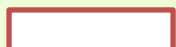
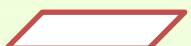

Output: The sum of two numbers

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

Solution: $C = A + B$ where the result is C.

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter the number A and the number B 3. Performing the sum of the two numbers using this equation $C = A + B$, the output is C. 4. Print C 5. End 	 <pre> graph TD Start([Start]) --> Read[/Read A and B/] Read --> Process[C = A + B] Process --> Print[/Print C/] Print --> End([End]) </pre>

In the previous example, note that:

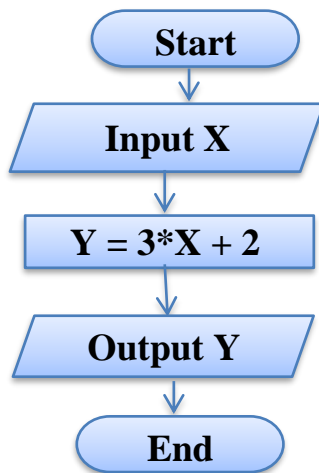
1. The flowchart should start and end with the "Terminal" symbol 
2. A, B, C, are variable names .A **Variable** refers to a memory storage that holds a value.
3. The equation $C = A+B$, indicates the sum of the value of A to the value of B , then store the result in variable C
4. Entering values in the variables A and B is done by using the term "Enter" inside the parallelogram  , you can also another term to get the same meaning like "Read" or "Input". (Input/output shape).
5. Any arithmetic operation (such as: equation of the sum) is written inside the rectangle  (Process shape).
6. The output is expressed with a parallelogram  using the term "Output" , or "Print".
7. We use the flow lines  to show the exact order of Algorithm.

Example (2)Solving the first degree equation $Y = 3X + 2$ **The answer:**First: Define the problem

Output: The value of "Y".

Input: X.

Solution: Compute the value of "Y" from the equation $Y=3x+2$.

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter value of X 3. Calculate $Y = (3 * X + 2)$ 4. Output value of Y 5. End 	 <pre> graph TD Start([Start]) --> Input[/Input X/] Input --> Process[Y = 3 * X + 2] Process --> Output[/Output Y/] Output --> End([End]) </pre>

REMEMBER

1. The left hand side (LHS) of any equation should contain only one variable, the value of this variable will be the (**output**) of the equation.
2. But the right hand side (RHS) of the equation may contain absolute value or expression that contains one or more variables (**inputs**).

Activity (1)

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

The answer:**First: Define the problem**

Output: compute the area and the perimeter of a rectangle,

Input: the length (L) and width (W) of rectangle.

Solution: Area = L*W and Perimeter = 2*(L+W)

Second : Algorithm	Flowchart
	<pre> graph TD Start([Start]) --> Input[/Input/] Input --> Process[Process] Process --> Output[/Output/] Output --> End([End]) </pre>

Activity (2)

Write down the Algorithm, and draw a flowchart to calculate the area of a circle whose radius "R" is known, bearing in mind that the equation of the area is: **Area =3.14*R*R.**

The answer:**First:** -----

Output: -----,

Input: -----

Solution: -----

Second : Algorithm	Flowchart

Activity (3)

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

The answer:**First:** -----

Output: -----,

Input: -----

Solution: -----

Second : -----	-----

ANSWER THESE QUESTIONS

[A] Put (✓) or (✗)

1. The Problem Solving is the objective or the specific output that we want to attain. ()
2. The Problem is the steps, activities, and processes to be done to reach an output or objective. ()
3. Determine of required outputs, available inputs and arithmetic or logical operations is the first of problem solving stages and called (Problem Definition)()
4. A Flowchart is a representation depends on drawing by standard graphical symbols to illustrate steps of the solution (Algorithm). ()
5. Use flowcharts to represent steps to solve the problem more difficult to resolve the problem of the programmer. ()
6. The Variable refers to a memory storage that holds a value. ()
7. The Program Documentation is beneficial when more than one person participate in writing or modifying the program. ()
8. The Stage of experience program on data their result already known is called (Documentation). ()

[B] Draw a flowchart for a program that will compute the average and Product of three numbers (A, B, C) where **average = $(A+B+C)/3$** and **Product = $A*B*C$**

The answer:

First: -----

Output: -----,

Input: -----

Solution: -----

<u>Second</u> : -----	-----

[C] Re-arrange these statements:

- () Program design.
- () Program Documentation.
- () Program Testing.
- () Performing step-by-step instructions (Algorithm).
- () Problem Definition.



The use of Branching (Decision) in Flowchart:

- We use the "**Decision**" shape in a flowchart, if the problem include a Yes / No question, where two possible answers are available "**Yes**" and "**No**".
- The decision point has two paths (branches), each presents one of the possible answers.

Example (3)

Draw a flowchart to print the word "successful" in case the degree input is greater than or equal to 50.

The answer:

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	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter the values of X 3. If $X \geq 50$ then (3-1) Print "successful" 4. End 	<pre> graph TD Start([Start]) --> EnterX[/Enter X/] EnterX --> Decision{X >= 50} Decision -- True --> Print[/Print "Successful"/] Print --> End([End]) Decision -- False --> End </pre>

Notice:

The execution of step (3-1) will be done only when the value of x is greater than or equal to 50 (the result of the condition is "**True**"), then it moves to step (4); otherwise when the value of x is less than 50 (the result of the condition is "**False**"), the execution is transferred to step 4 directly.

Example (4)

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

The answer:

First: Define the problem

Output : Print the result of dividing two numbers "R" or print the word "Undefined".

Input : The dividend (Numerator) is "num1", and the divisor (denominator) is "num2".

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

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter the dividend num1 3. Enter the divisor num2 4. If num2 =0 then <ol style="list-style-type: none"> (4-1) Print "Undefined" (4-2) Go to step (6) 5. Else <ol style="list-style-type: none"> (5-1) $R = \text{num1}/\text{num2}$ (5-2) Print R 6. End 	<pre> graph TD Start([Start]) --> Enter1[/Enter num1/] Enter1 --> Enter2[/Enter num2/] Enter2 --> Decision{num2=0} Decision -- True --> PrintU[/Print "Undefined"/] Decision -- False --> ProcessR[R= num1/num2] ProcessR --> PrintR[/Print R/] PrintU --> End([End]) PrintR --> End </pre>

Notice:

1. Step (4) is a decision (represented by the rhombus shape).
2. It is a comparison that evaluates this question (is num2=0?).
3. If the result is "True", the step (4-1) is executed {the word "undefined" will be printed} according to mathematical rules; then execute the step (4-2) then, will be transferred to step (6) to end the program and prevent the execution of the division.

4. If the result is "False", the step (5) is executed directly {because step (4-1) and (4-2) won't be executed} and execute the step (5-1) then the step (5-2), then, will be transferred to step (6) also, to end the program

Example (5)

Draw a flowchart for a program that obtains (يحصل عليه) a number from the user. Determine the number type (Even or Odd) and print the result.

The answer:

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	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter N 3. If N is divisible by 2 without remainder then (3-1) Print "Even" (3-2) Go to step (5) 4. Else (4-1) Print "Odd " 5. End 	<pre> graph TD Start([Start]) --> EnterN[/Enter N/] EnterN --> Divisible{N Divisible by 2} Divisible -- True --> PrintEven[/Print "Even"/] Divisible -- False --> PrintOdd[/Print "Odd"/] PrintEven --> End([End]) PrintOdd --> End </pre>

Example (6)

Get a temperature degree from the user, and print out the following results "greater than zero" - "less than zero" - "equal zero".

The answer:

First: Define the problem

Output : Print out "greater than zero" - "less than zero" - "equal zero".

Input : temperature degree Celsius "D".

Solution : the temperature degree entered will be compared to zero.

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter D (temperature degree) 3. If D=0 then <ol style="list-style-type: none"> (3-1) Print "Equal zero" (3-2) Go to step (4) 4. Else <ol style="list-style-type: none"> (4-1) if D<0 then <ol style="list-style-type: none"> (4-1-1) Print "Below zero" (4-1-2) Go to step (4) (4-2) Else <ol style="list-style-type: none"> (4-2-1) Print "Above zero" 5. End 	<pre> graph TD Start([Start]) --> EnterD[/Enter D/] EnterD --> D0{D = 0} D0 -- True --> PrintEqual[/Print "Equal 0"/] D0 -- False --> Dlt0{D < 0} Dlt0 -- True --> PrintBelow[/Print "Below 0"/] Dlt0 -- False --> PrintAbove[/Print "Above 0"/] PrintEqual --> End([End]) PrintBelow --> End PrintAbove --> End </pre>

Activity (4)

Write down the Algorithm, and draw a flowchart to enter two numbers, then Print "the largest number is" and, "the smallest number is....".

The answer:

First: Define the problem

Output: print out "the largest number is ..." - "the smaller number is ..."

Input: X, Y.

Solution: -----

Second : Algorithm	Flowchart

Activity (5)

The following flowchart is used to calculate the Area of a circle whose 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).

Flowchart	Modified Flowchart
<pre> graph TD Start([Start]) --> Input[/Input R/] Input --> Process[A = 3.14*R*R] Process --> Output[/Print A/] Output --> End([End]) </pre>	



ANSWER THESE QUESTIONS

Choose the correct answers:

- 1- ----- Symbol is used to connect a shape to another in the flowchart. { *Decision – Flow line – Input/Output* }
- 2- ----- means a store in the computer memory that has a certain title and its value changes during the program running. { *Variable – Assignment Statement – Algorithm* }.
- 3- Which of the following symbols describe the decision-making processes in flowchart :



The use of Loop in Flowchart:

Example (7)

Print out the numbers from 1 to 3.

The answer:

First: Define the problem

Output : Print out the numbers from 1 to 3

Input : nothing

Solution : print out the number M then add 1 and print again until the Value of M becomes 3.

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. $M = 1$ 3. If $M \leq 3$ then <ol style="list-style-type: none"> (3-1) Print M (3-2) $M = M + 1$ (3-3) Go to step (3) 4. End 	<pre> graph TD Start([Start]) --> M1[M = 1] M1 --> Cond{M <= 3} Cond -- True --> Print[/Print M/] Print --> Mplus[M = M + 1] Mplus --> Cond Cond -- False --> End([End]) </pre>

What Answer of:

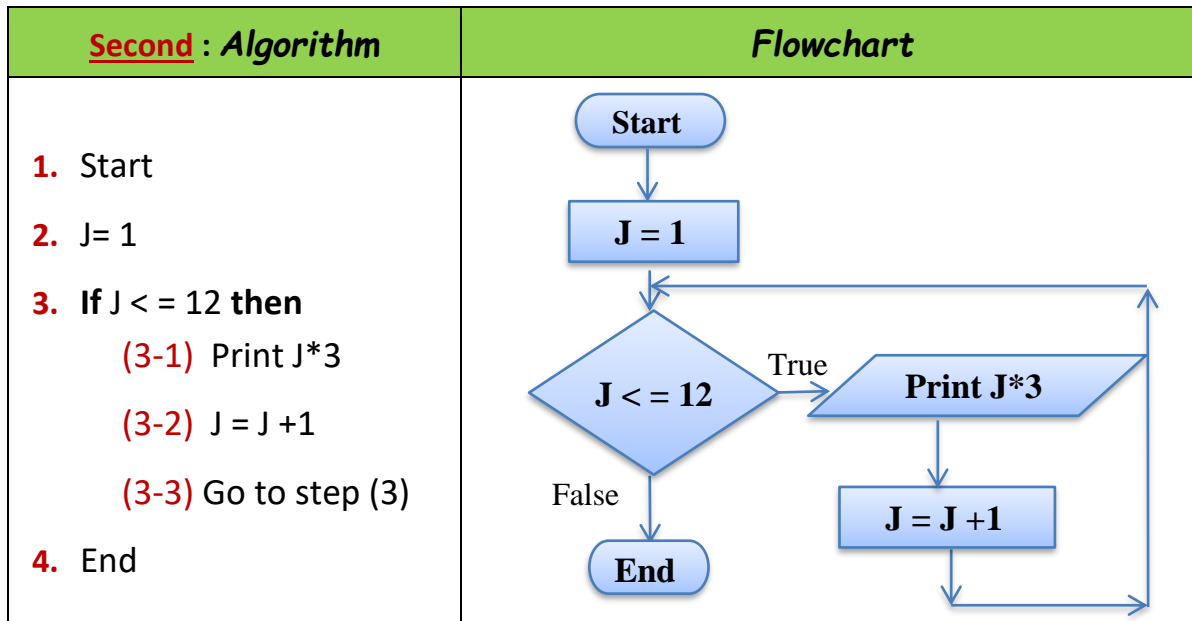
- How many times was the content of the loop executed?
- What will be the value of M at the end of the loop?

Notice: In the previous example:

1. As long as the value of **M** does not exceed (لم تتجاوز) **3**, (the condition result is "True") the program execute step (3-1) that is mean print the value of **M**, then transfer to step (3-2) then (3-3).
2. When the value of **M** equals **4**, (the condition becomes "False"), the program goes to step (4).
3. We conclude that (نستنتج ان) the number of repetition in the loop is pre known (معروف مسبقا اي نعرفه) in this exercise.
4. The variable **M** is a **counter**; as it causes the steps (3-1) and (3-2) and (3-3) to be repeated for a defined number of times (3 times).

Example (8)

Modify the flowchart of the pervious exercise to print the multiplication table of number 3.



Activity (6)

- Track (تتبع مسار) the values of the variable (J) and the printed value on executing every step in the previous exercise (write down in your notebook).
- What is the value of the variable (J), when the result of the condition is “False”, and the loop ends?

Complete the following sentences:

- 1] The value of the variable (J) is
- 2] The loop ends

Value of J	J*N
1	3
2	6
3	9
4	12
-----	-----
-----	-----
-----	-----
-----	-----
-----	-----
-----	-----
-----	-----
-----	-----
12	36
13	

Activity (7)

Make the necessary modifications to the Flowchart of the previous exercise, so that you can enter the required multiplication table constantly (بإستمرار); instead of (بدلاً من) printing the multiplication table of "3".

Second : Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. Enter N 3. J=1 4. If J<=12 then <ol style="list-style-type: none"> (4-1) Print J*N (4-2) J=J+1 (4-3) Go To step(4) 5. End 	

Activity (8)

Write down the Algorithm, and draw a flowchart to print out even numbers from 1 to 10

Second : Algorithm	Flowchart

Example (9) Print out the sum of integer numbers from 1 to 3.

Algorithm	Flowchart
<ol style="list-style-type: none"> 1. Start 2. N=1 3. Sum=0 4. Sum =Sum + N 5. N=N+1 6. If N>3 Then <ol style="list-style-type: none"> (6-1) Print Sum (6-2) go to step (8) 7. Else <ol style="list-style-type: none"> (7-1)Go to step (4) 8. End <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Notice : Variable N is considered a Counter while variable Sum is an accumulating (تجميعي) variable.</p> </div>	<pre> graph TD Start([Start]) --> N1[N = 1] N1 --> Sum0[Sum = 0] Sum0 --> SumAdd[Sum = Sum + N] SumAdd --> NInc[N = N + 1] NInc --> Ngt3{N > 3} Ngt3 -- True --> PrintSum[/Print Sum/] PrintSum --> End([End]) Ngt3 -- False --> SumAdd </pre>

Activity (9)

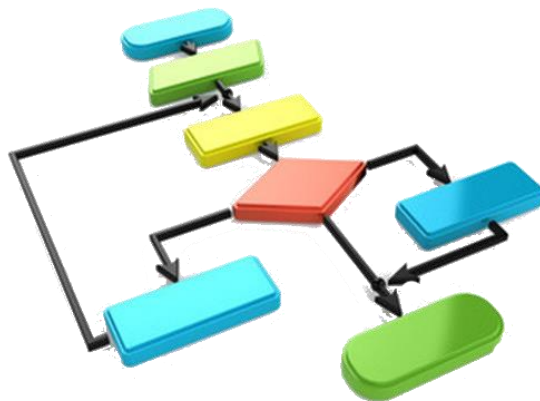
Redraw the Flowchart of the previous exercise in your notebook, after modifying it; to print out the sum of odd numbers from 1 to 10.

Algorithm	Flowchart

Activity (10)

After executing the previous activity, draw the flowchart in your notebook; to print out the sum of even numbers instead of the odd ones.







Algorithm	Flowchart





The Questions

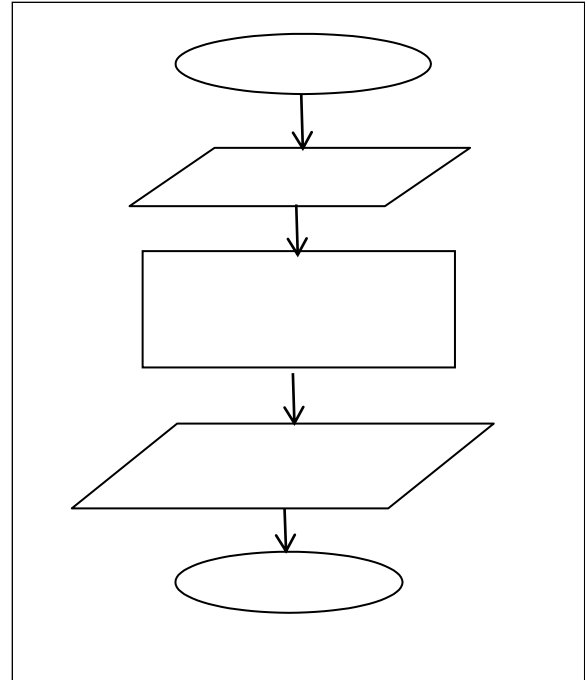
First: Put (✓) in front of the correct sentence and (✗) in front of the wrong one:

1. Flowcharts use standard symbols and lines to represent a problem Algorithm. ()
2. You can use any Geometric shape to represent Algorithms when drawing flowcharts ()
3. Flowcharts can be drawn using software only and can't be drawn on paper. ()
4. The symbol  is used to represent start and end of flowchart ()
5. The rectangle symbol  represents only one processing operation. ()
6. The flow of steps will always be from top to bottom or from left to right. ()
7. The symbol  is used when a question has more than one alternative ()
8. Two paths (lines) should come out from the diamond  symbol ()
9. The line with an arrow (flow line) should be from left to right or from top to bottom ()
10. The (Algorithm) is the first stage of Problem Solving. ()
11. The rectangle symbol  is used to represent the data input process.()
12. The symbol  is used to represent a decision process in flowcharts. ()
13. The problem means that an objective or output is required to reach. ()
14. Preparing a cup of tea is an example of a problem. ()
15. Problem solving is the steps, activities, and processes to be done to reach an output or objective. ()
16. The program documentation is a set of procedures arranged logically for solving a specific problem. ()
17. The program testing is writing down all the steps taken to solve a problem. ()
18. Documenting the program means making sure that the program is free of errors. ()
19. Algorithm is a set of procedures arranged logically for solving a specific problem. ()
20. The program documentation is writing down all the steps taken to solve a problem. ()
21. Testing the program means making sure that the program is free of errors. ()
22. Flowcharts help to facilitate understanding of the problem, analyses and convert it to a program. ()

Second: Inside each symbol of the following Flowcharts write the appropriate instructions.

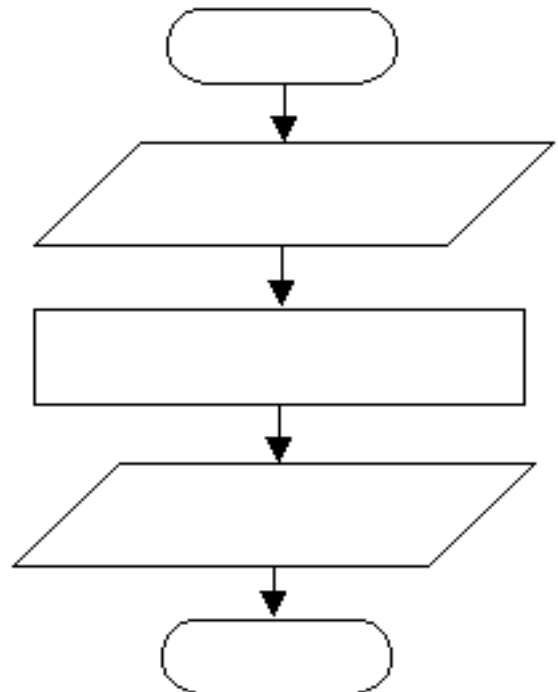
- The Figure shows a flow chart for calculating the area and circumference for a circle ,given: the radius of the circle R

$A = 3.14 * R * R$ $C = 2 * 3.14 * R$
START
END
OUTPUT A OUTPUT C
READ R

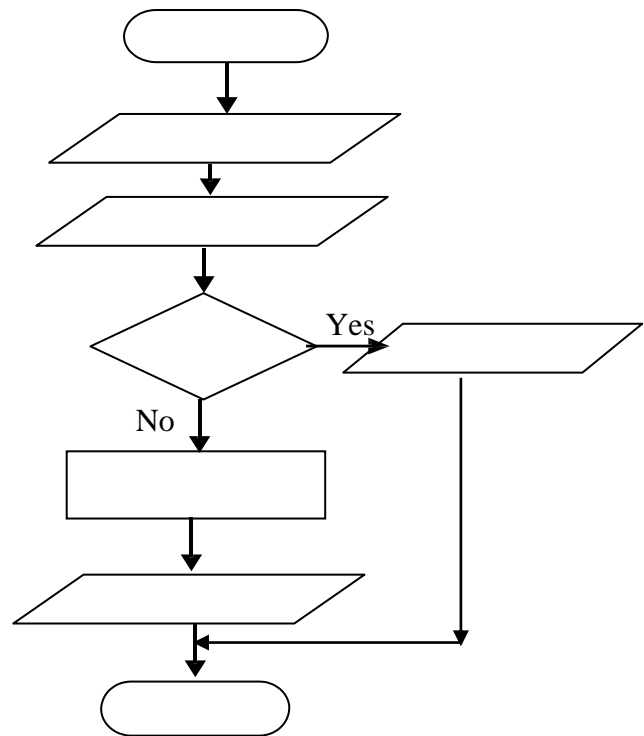
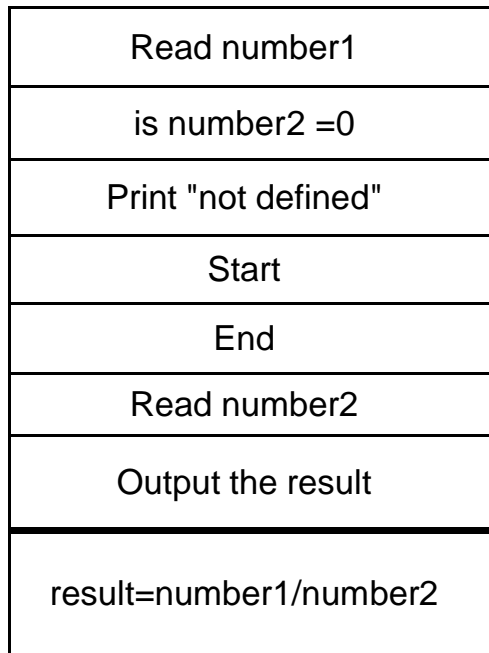


- The Figure shows a flowchart that calculates the interest value for the balance (with fixed interest rate)

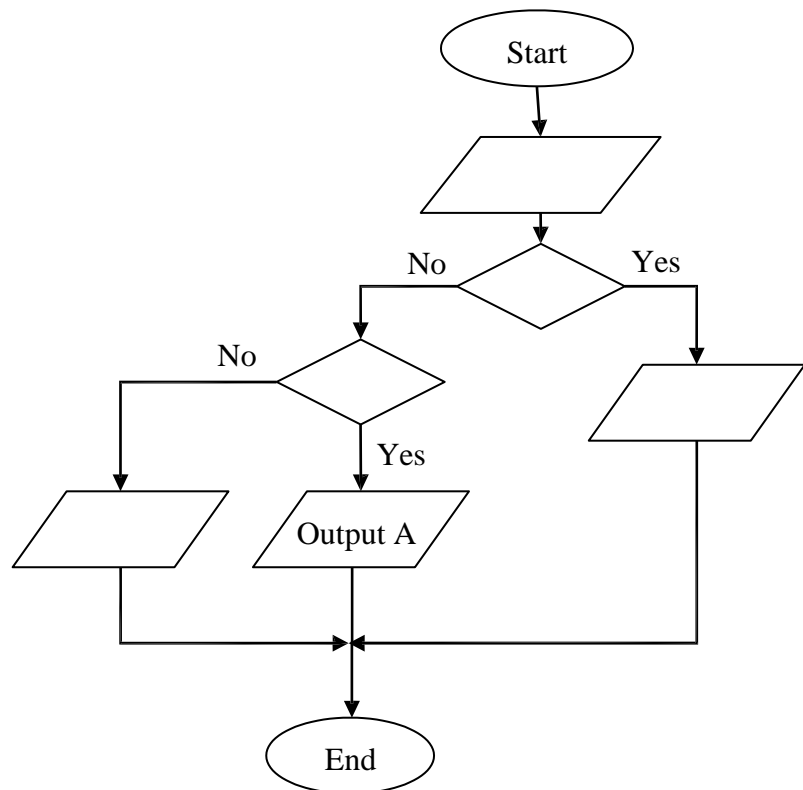
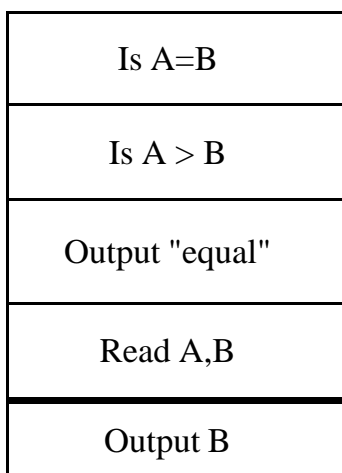
Interest=balance*rate
END
Read name , balance , rate
Output name, balance ,interest
START



3. The Figure shows a flow chart for calculating the division of two numbers.

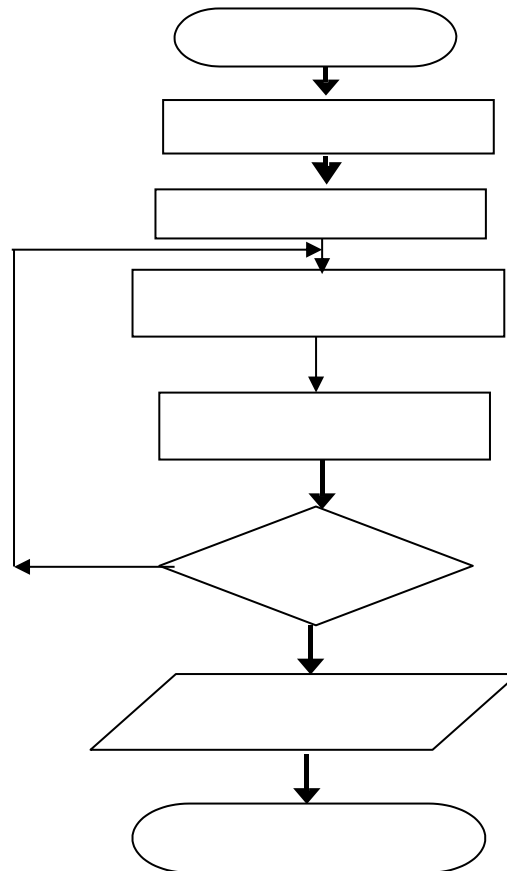


4. The Figure shows a flow chart to find the maximum number among two numbers .In the case of equality output the message "equal"



5. The Figure shows a flow chart that will print even integer numbers from 1 and 10 .

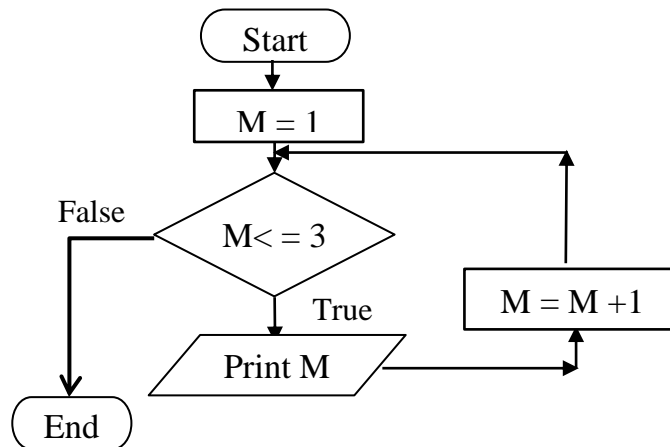
$N = N + 2$
$N > 10$
$Sum = 0$
$Sum = Sum + N$
Print Sum
$N = 2$



Third: 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:
 - problem definition
 - problem
 - problem solving
- On drawing flowcharts we use:
 - Standard symbols and lines.
 - All geometric figures
 - One geometric figure
- A set of procedures arranged logically for solving a specific problem – can be called:
 - problem
 - algorithm
 - program testing
- Making sure that the program is free of errors – can be called:
 - program testing
 - program documentation
 - algorithm
- Writing down all the steps taken to solve a problem errors – can be called:
 - program documentation
 - program testing
 - flowcharts

- 6) Problem-solving approach includes many terminologies, the terminology that expresses the preparation of a cup juice is:
 a. Flowchart b. algorithm c. problem
- 7) Problem-solving approach includes many of the terminologies, the terminology that expresses the mathematical problem is:
 a. Algorithm b. problem c. program design
- 8) 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
- 9) In the following flowchart:



The number of iterations (print the value of M) is:

- a- 2 b- 3 c- 4
- 10) In Flowchart of the previous question, the value of M after the end of the iterative loop equals:
 a. 2 b- 3 c- 4



Chapter Two

Introduction to

VB.Net



VB.net

Preface

In this chapter you will deal with Visual Basic.net program which will enable you to convert the steps of solving a problem into program codes that can be carried out.

Visual Basic .NET Language:

1. It is one of the high level programming languages
2. Designed to be easy to learn as its commands and instructions use English language vocabulary.
3. It can be used to create many applications such as:
 - ♠ Windows applications
 - ♠ Web applications

Programming and Computer memory:

Programming is the Commands and Instructions that are written in **Visual Basic.net** to enable you to create objects in computer memory.

Each 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.

Visual Basic.net is considered:

- **Object oriented** because their programs work through objects in computer's Memory , such as (*Button , TextBox or Label controls*)
- **Event Driven** because commands and instructions are carried out as soon as certain event occurs.

The language of Visual Basic.net and .Net framework:

The Framework.Net provides the following:

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

The Elements of IDE screen:

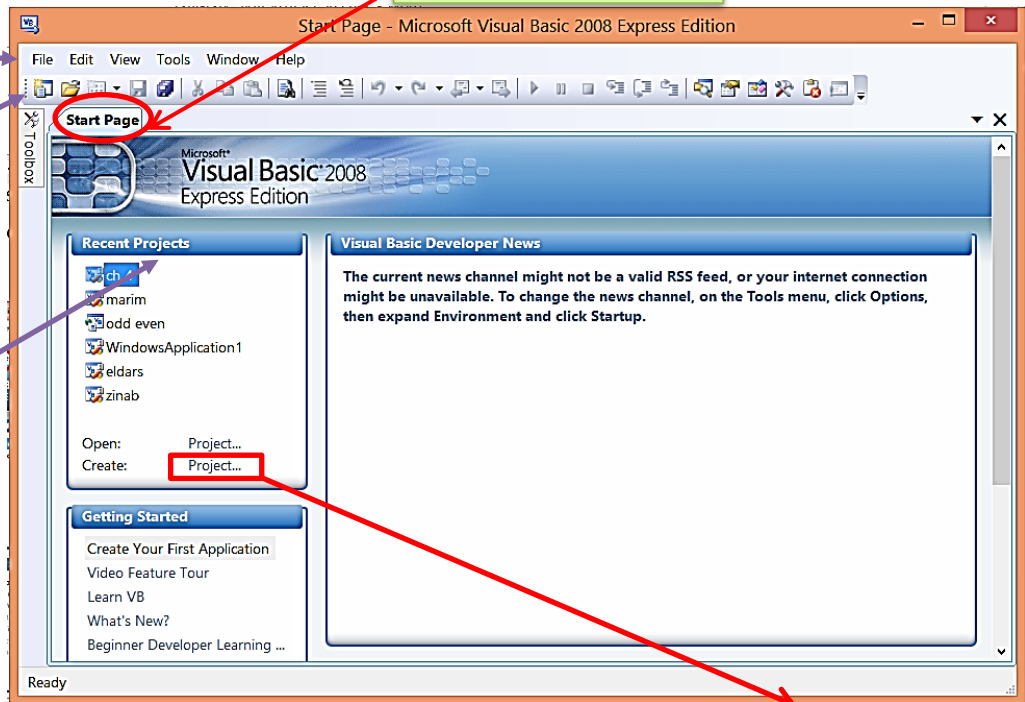
- The term (**IDE**) refers to **I**ntegrated **D**evelopment **E**nvironment.
- It contains all the tools which the programmer of VB.net is needed in create applications (Windows – Mobile – Web -)
- **Visual Studio** represents **IDE**

This page called "Start Page"

Menu Bar

Toolbar

Recent Projects :
used to open and view previous projects OR create new project



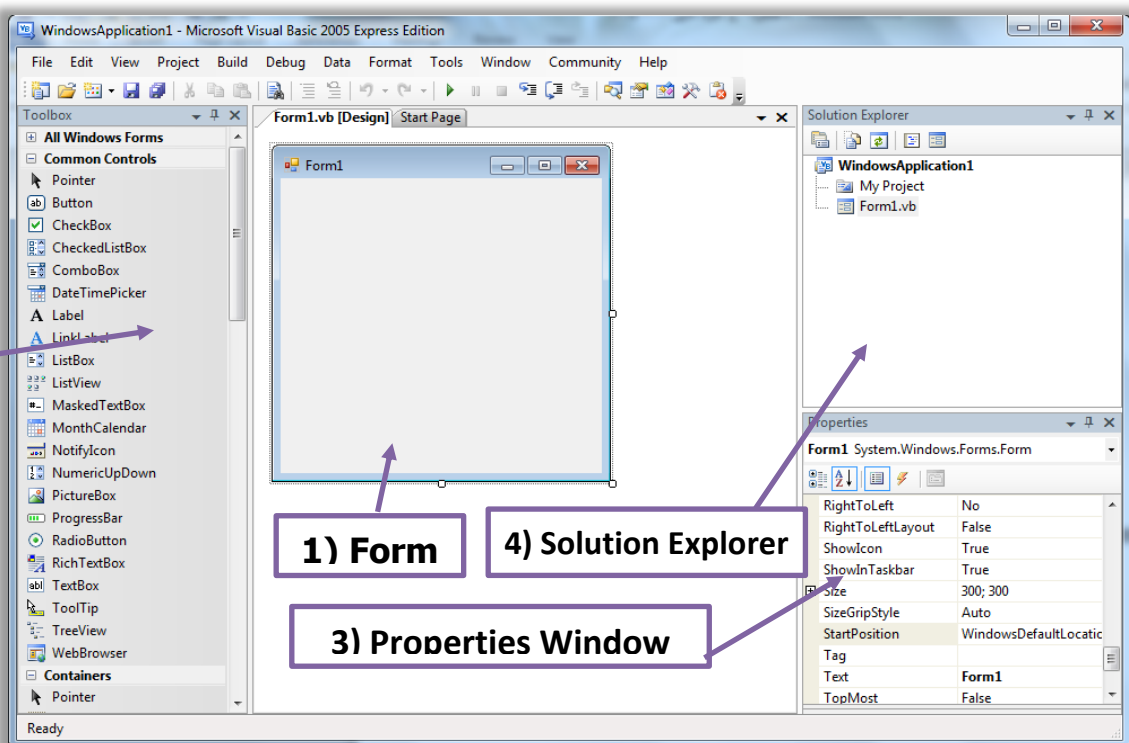
When the programmer creates a new project, by click on "Project" in front of "Create" in "Start Page" window. **The (IDE) screen for new project** appears:

2) Toolbox

1) Form

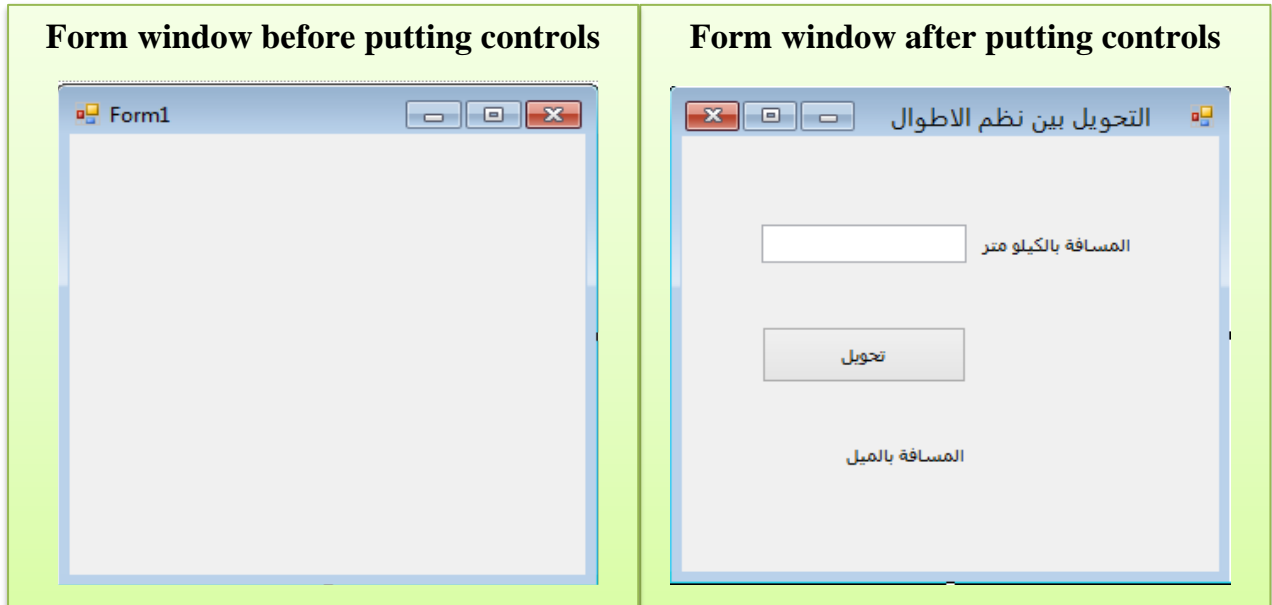
4) Solution Explorer

3) Properties Window



1) Form :

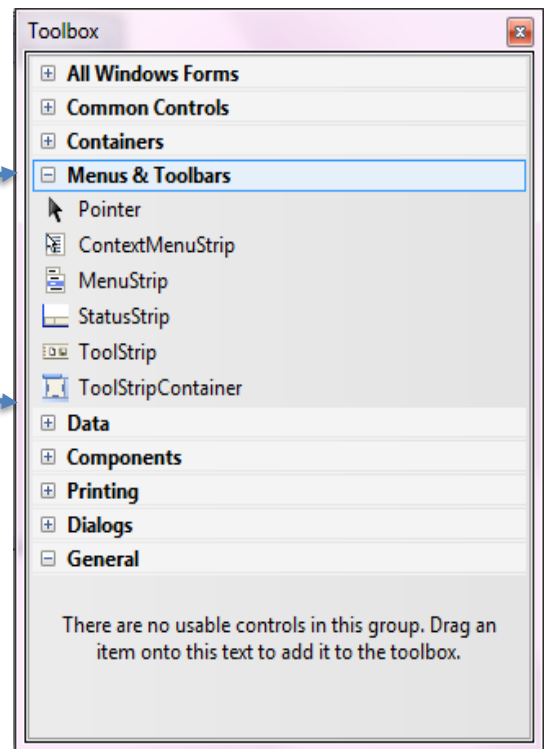
- The form is the interface which the user deals with through different controls such as Button, Textbox, label.....etc.
- A form is the container upon which controls are placed.

**2) Toolbox :**

- The Toolbox contains controls (objects) that the programmer can place on the form.
- these controls are putting Categories (تصنيفات) such as :
 - 1) Common Controls.
 - 2) Menus & Toolbars.

And you can show all tools of controls by choosing the category (*All Windows Forms*)

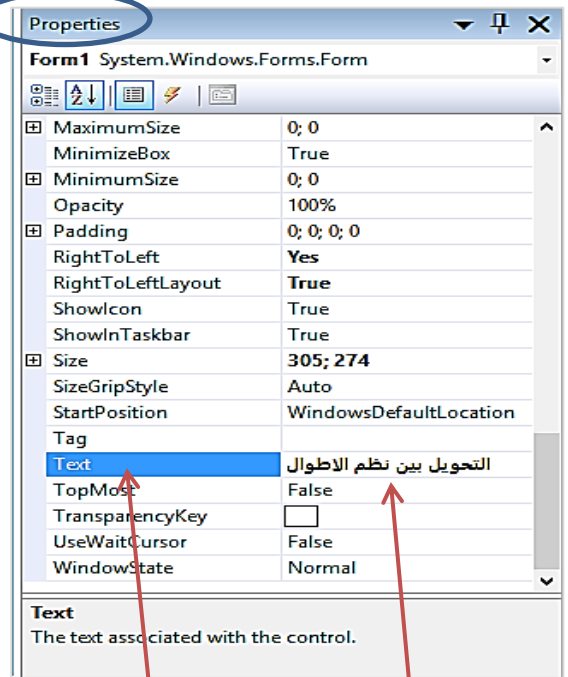
- If you click on a (+) sign, the tab expands (يتمدد / ينسدل) and a set of controls will be displayed.

**Some of the Common Controls :**

ComboBox – CheckBox – RadioButton – Button – TextBox – Label – ListBox

3) Properties window :

- Each control has a group of properties.
- You can be adjusted these properties through "**Properties Window**"
- In (IDE) screen, displayed properties differ upon the selected element.
(تختلف الخصائص IDE ، بناء على العنصر المحدد) (المعروضة في الشاشة).
- The shown Properties in Properties Window are different according to the active part on the IDE screen.

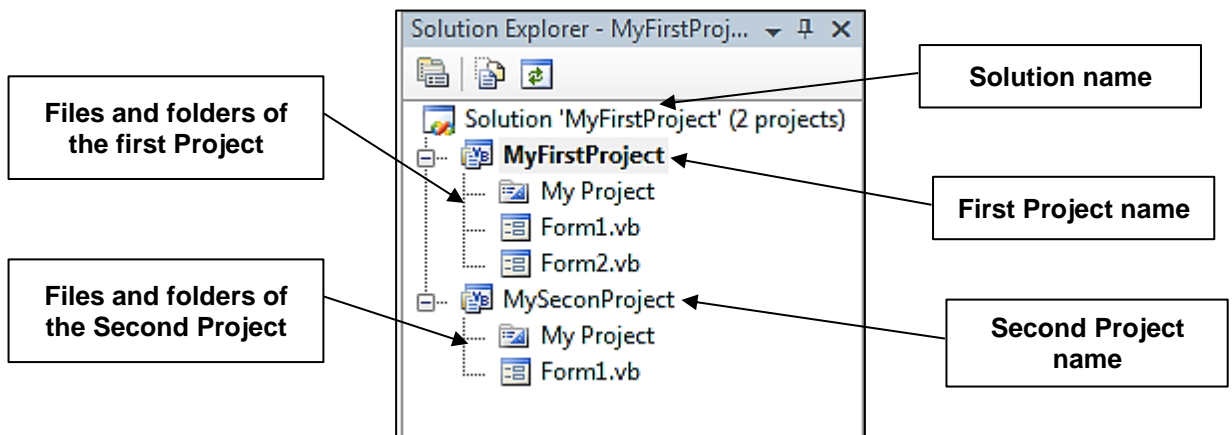


Property name
(Left column)

Property value
(Right column)

4) Solution Explorer window :

- It contains a list of items (Folders and Files) of the current solution; and may contain one or multiple projects.





The Questions

Q (1) : Put (✓) in front of the correct sentence and (×) in front of the wrong one:

1. The VB.net language is one of the high level languages. ()
2. The VB.net language is one of Event Driven languages. ()
3. The VB.net language is the only high level language. ()
4. The VB.net language is considered a high level language because it is easy to learn. ()
5. The VB.net language is used in producing Windows applications and Web applications. ()
6. The VB.net language is used in producing Web applications only. ()
7. The VB.net language can't be used in producing Windows applications. ()
8. Every Object is characterized by certain properties and certain behavior when a certain event occurs on it. ()
9. Events and procedures which belong to any object in VB.net language are called properties. ()
10. The name, the size and color of an object are all samples of events that can occur to the object in VB.net language. ()
11. The name, the size and color of an object are all samples of properties of some objects in VB.net language. ()
12. The Events are the commands and instructions which are carried out when a certain procedure occurs to the object in VB.net language. ()
13. The procedures are the commands and instructions which are carried out when a certain procedure occurs to the object in VB.net language. ()
14. Pressing click and D-click are samples of some events that can occur to an object in VB.net language. ()
15. Framework.net contains Compilers, libraries and runtime environment. ()
16. Compilers in Framework. Net are considered the environment of runtime for applications which are produced in VB.net language. ()
17. Compilers are programs that translate commands and instructions written by the programmer from the high level language into machine language. ()

18. Object oriented programming languages are the languages which work through objects that carry out procedures when a certain event occurs to them. ()
19. All programming languages which carry out a group of commands and instructions are considered as Event Driven languages. ()
20. Visual Studio is considered IDE because it includes a group of tools, elements and characteristics necessary to produce applications. ()

Q (2): 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 colour are called:**
 - a. Properties
 - b. Procedures
 - 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. Procedure
 - c. Events
- 6) **The Properties term refers to:**
 - a. Features that describe the object.
 - b. Events that can occur to the object.
 - a. 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.
- a. 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

The Form:

Remember

- The form is the interface which the user deals with through different controls such as Button, Textbox, label.....etc.
- A form is the container upon which controls are placed.

The Properties assigned to the form:

No	Property	Function
1	Name	Name of the Form used in the code window. Write (frm) in beginning its name
2	Text	Text appearing (displaying) on the title bar of the Form.
3	FormBorderStyle	The Border outline of the Form's window.
4	BackColor	The background color of the Form's window.
5	WindowState	Determine the size of the window on the screen (takes value "Maximized" or "Minimized" or "Normal"))
6	ControlBox	It controls the appearance or disappearance of ControlBox of Form Window. (take a value "True" or "False"))
7	MinimizeBox	It controls the appearance or disappearance of Minimize Box of Form Window. (take a value "True" or "False"))
8	MaximizeBox	It controls the appearance or disappearance of Maximize Box of Form Window. (take a value "True" or "False"))
9	RightToLeftLayout	The Layout direction of (Controls) on the (Form) is from right to left. (take a value "True" or "False"))
10	RightToLeft	The writing direction of (Controls) on the (Form) is from right to left; such as the text direction in the (TextBox) . (take a value " Yes" or "No"))

To adjust the Form properties

(1) Create a new project :

1. Open VB.Net from Start menu.
2. The IDE screen appears.
3. From "**File**" menu choose "**New Project**".
4. **Window template** is displayed.
5. Choose "**Windows Forms Application**".
6. Type the project name in "**Name**" box.
7. Press "**OK**".

(2) Active the Form by clicking on it

By using the "Properties Window", we adjust the properties

as the following:

The name property of Form

The left column display (The name of properties)

The right column display (The value of properties)

Property	Value
RightToLeft	No
RightToLeftLayout	False
ShowIcon	True
ShowInTaskbar	True
Size	300; 300
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Form1
TopMost	False
TransparencyKey	<input type="checkbox"/>
UseWaitCursor	False
WindowState	Normal

Setting "Name" and "Text" properties:

The new name of "Form" after adjust the value of "Name" property

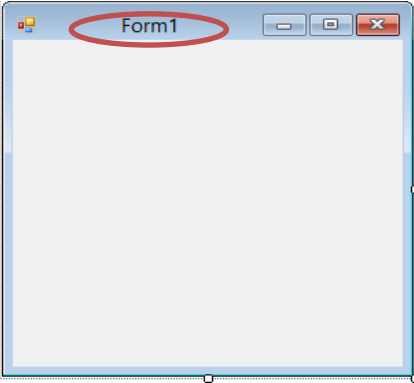
The new value of "Text" property

Property	Value
(ApplicationSettings)	
(DataBindings)	
(Name)	frmSquare
AcceptButton	(none)
AccessibleDescription	

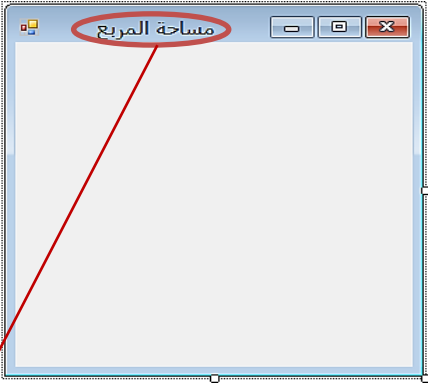
Property	Value
Padding	0; 0; 0; 0
RightToLeft	No
RightToLeftLayout	False
ShowIcon	True
ShowInTaskbar	True
Size	300; 300
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	مساحة المربع
TopMost	False
TransparencyKey	<input type="checkbox"/>

Name & Text Properties

Before setting "Name" and "Text" properties



After setting "Name" and "Text" properties



Shows only the value of property (*Text*) and does not show the value of the property (*Name*); used when writing the code.

NOTICE

- The default values of properties "Name" and "Text" are the same (Form1)
- The value of "Name" property doesn't appear on the "Form" because it is used as Form name in code window.
- The value of property "Text" show during design and running mode
- The value of "Text" property is displayed in the title bar of the "Form"

Setting "BackColor" property:

Select "BackColor" property from left column of the Properties window then choose the color (the value of property) from right column.

Properties

frmSquare System.Windows.Forms.Form

AutoScrollMargin	0; 0
AutoScrollMinSize	0; 0
AutoSize	False
AutoSizeMode	GrowOnly
AutoValidate	EnablePreventFocusChange
BackColor	<input type="checkbox"/> Control
BackgroundImage	<input type="checkbox"/> (none)
BackgroundImageLayout	Tile
CancelButton	(none)

Solution Explorer | Data | Custom | Web | System

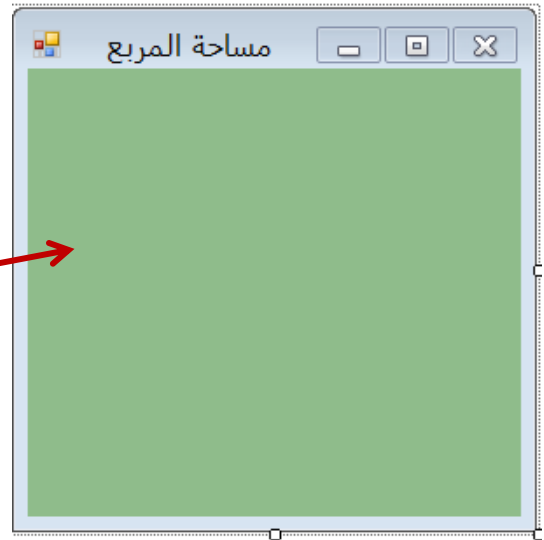
Properties

frmSquare System.Window

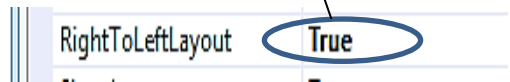
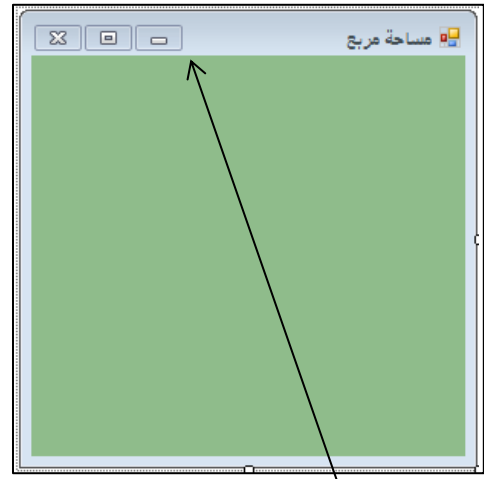
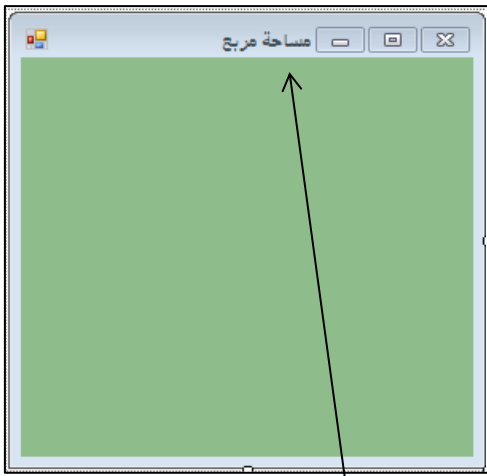
AutoScrollMargin	
AutoScrollMinSize	
AutoSize	
AutoSizeMode	
AutoValidate	
BackColor	<input checked="" type="checkbox"/> DarkSeaGreen
BackgroundImage	<input type="checkbox"/> (none)
BackgroundImageLayout	Tile
CancelButton	(none)
CausesValidation	True



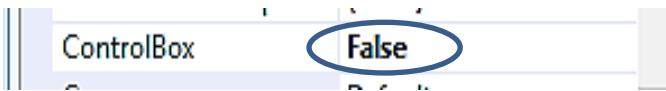
The background color of the Form becomes (DarkSeaGreen)



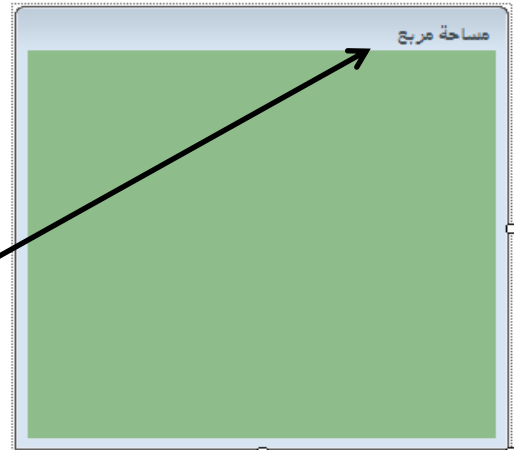
Setting "RightToLeft" and "RightToLeftLayout" property:



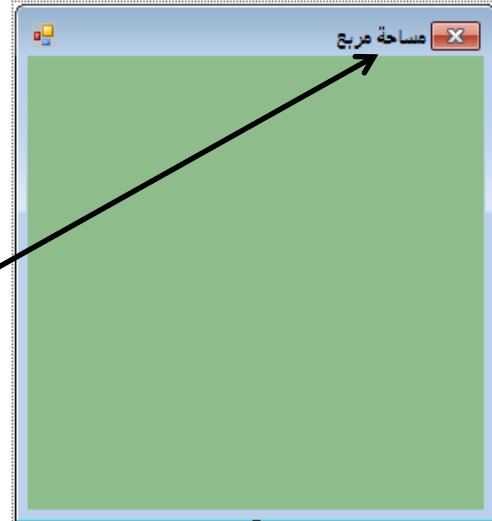
Setting "ControlBox", "MaximizeBox" and "MinimizeBox" property:



The Control Box is hidden



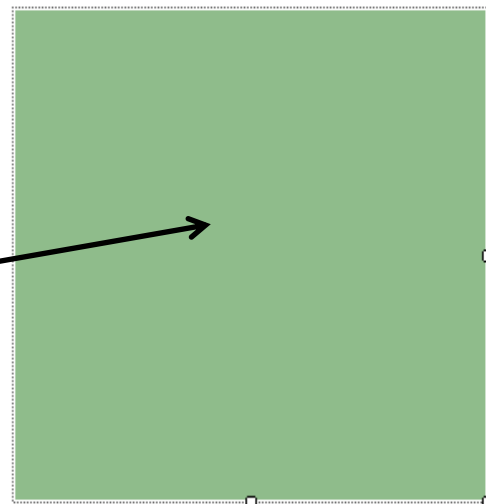
MaximizeBox	False
MaximumSize	0; 0
MinimizeBox	False



Note that the Maximize and Minimize buttons do not exist

Setting "FormBorderStyle" property:

FormBorderStyle	None
-----------------	------

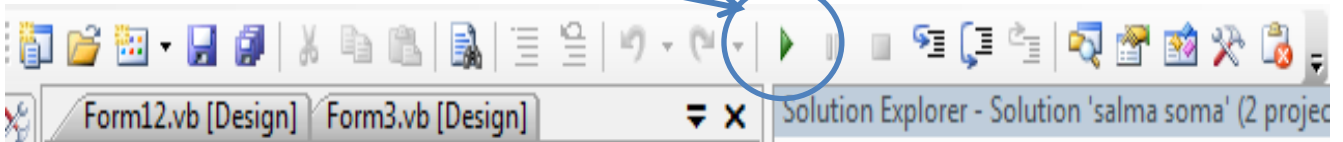


Note that the form's title bar and borders disappear

NOTICE

1. Some properties are common to most controls such as: (Text – Name – Forecolor – BackColor – RightToLeft – Visible – Enable -etc.).
2. Some properties will not be applied to controls placed on a form; unless we set other properties to these controls like : (RightToLeftLayout) and (RightToLeft).
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).
4. The default values of the properties (Text) and (Name) are same.
5. The property (RightToLeftLayout) will not be active, unless the property value of (RightToLeft) is (Yes).

6. The Property (**FormBorderStyle**) takes many values .The value (**Sizable**) makes it possible to control the Form' size; through its borders
يجعل من الممكن التحكم في حجم (Form) من خلال حدودها (إطارها).
7. When you set some properties, the effect of setting appears directly on the Form in design mode of the program.
8. There are some properties which their effect doesn't appear on the Form or Controls until you run the program such as (**WindowState**)
9. You can **initiate** (الشروع/ بدء) the (**Start Debugging mode**) by clicking on the (**Start Debugging**) icon on the Standard Toolbar, **OR** press "**F5**" key .



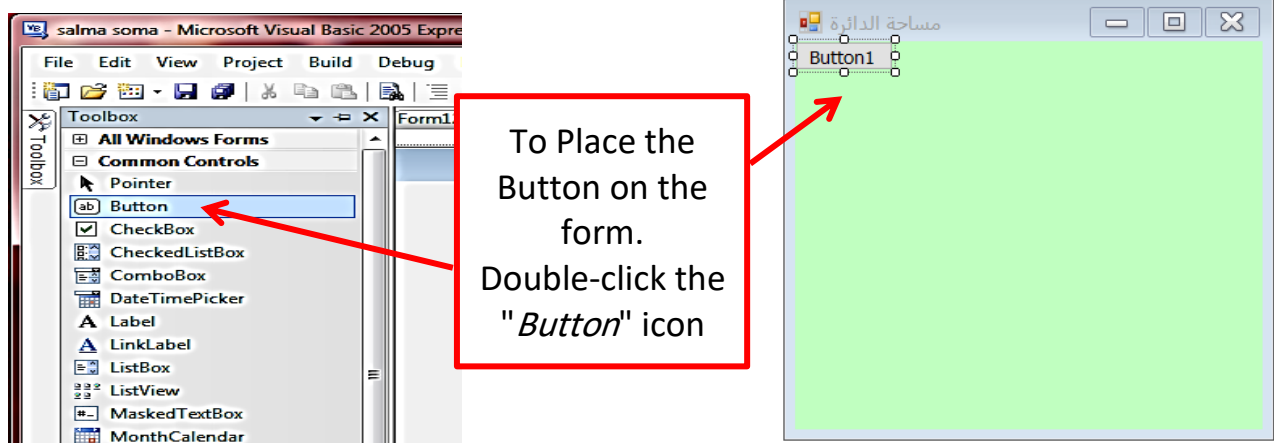
10. You can stop the (**Debugging**) by clicking on (**Stop Debugging**) icon to return to Design mode.



11. When you select the value "**Maximized**" to "**WindowState**" property, the Form's window is displayed in **full screen** ,in Running mode

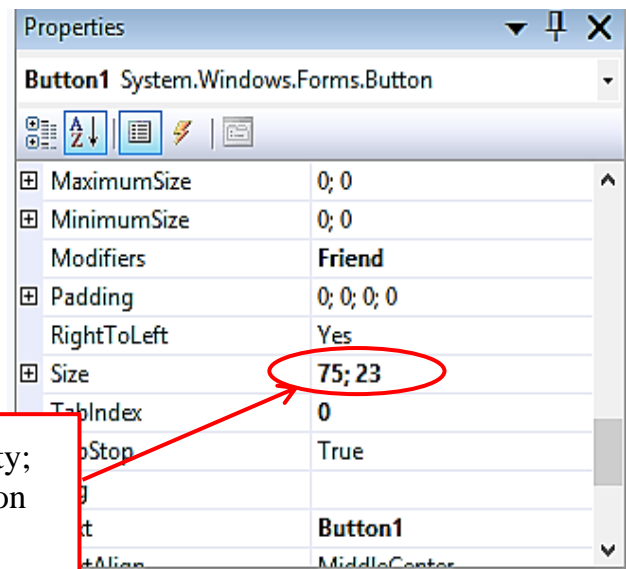
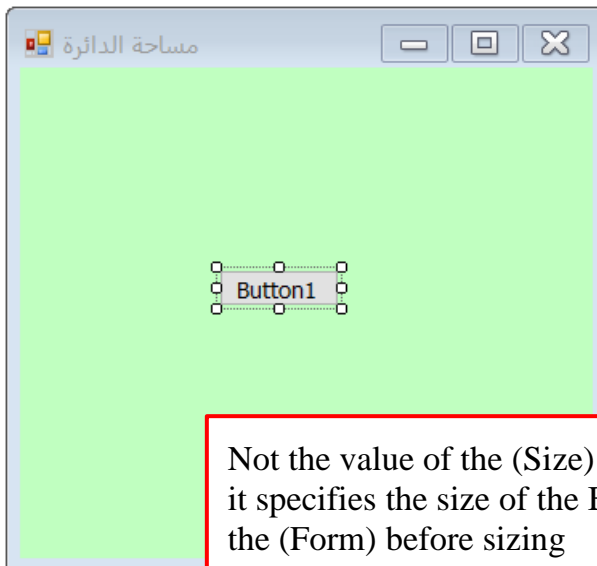
The Button:

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

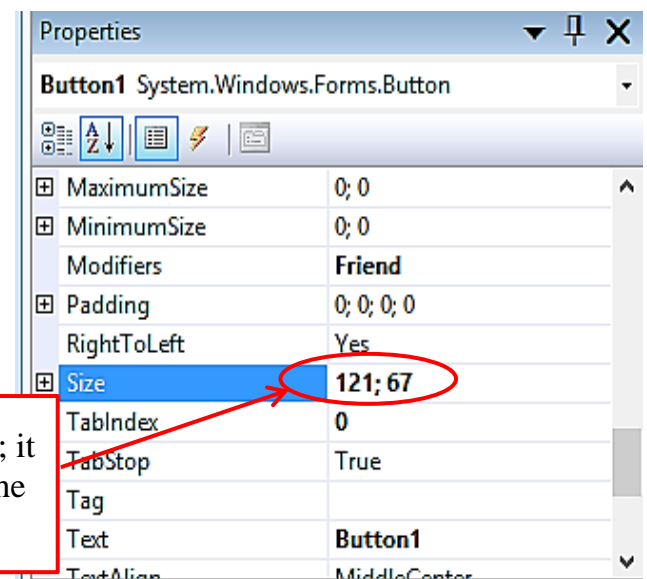
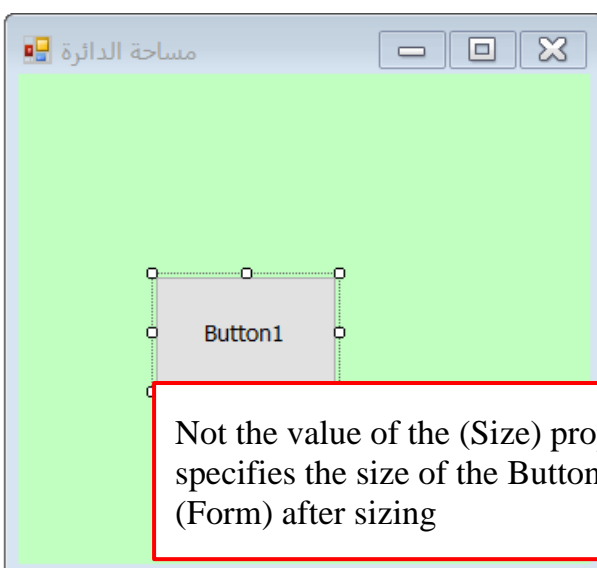


The Properties of Button :

No	Property	Function
1	Location	The location of the (Button) on the Form's window.
2	Size	The height and width of the (Button) on the Form
3	Text	The text is appeared on the (Button).
4	BackColor	The background color for the (Button)
5	Font	The text's (Font, Size and Style) on the (Button).
6	ForeColor	The foreground color of the text on the (Button)

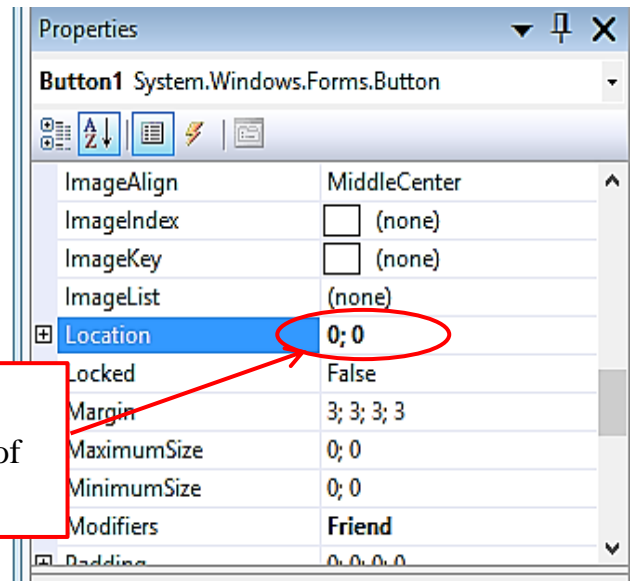
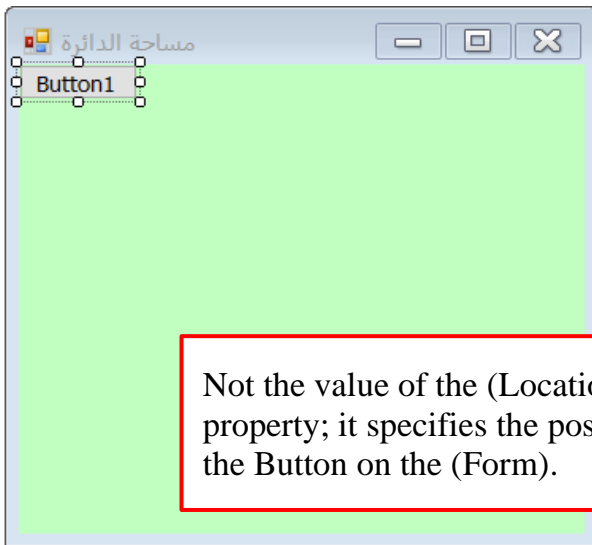
Set "Size" property:

Not the value of the (Size) property; it specifies the size of the Button on the (Form) before sizing

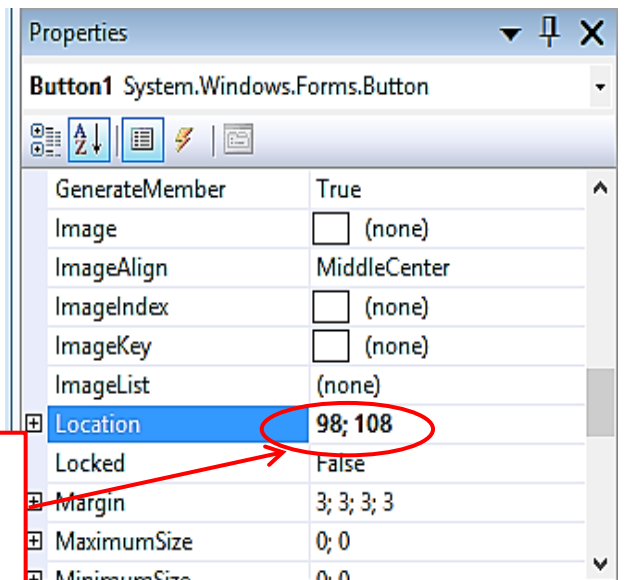
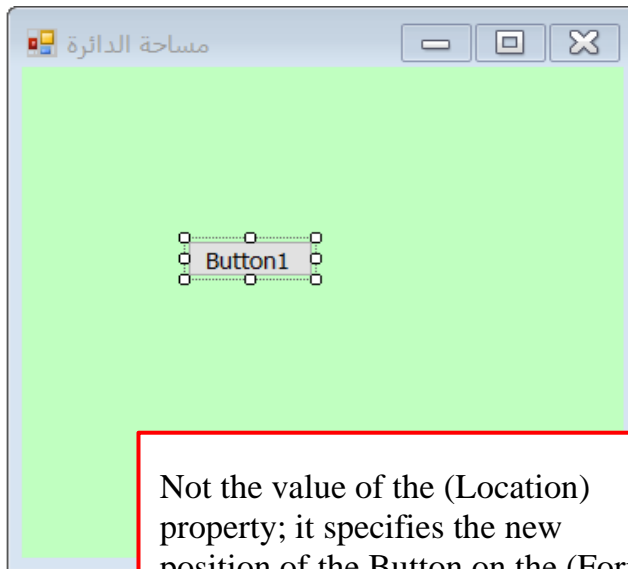


Not the value of the (Size) property; it specifies the size of the Button on the (Form) after sizing

Set "Location" property:

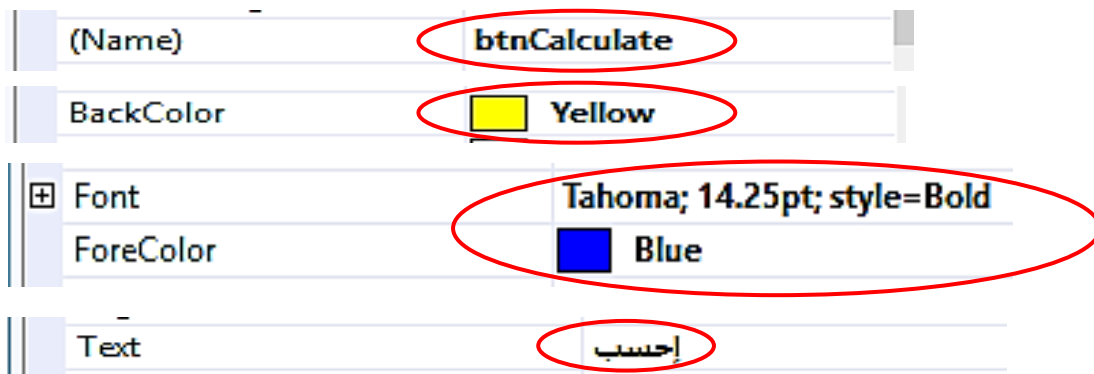


Not the value of the (Location) property; it specifies the position of the Button on the (Form).

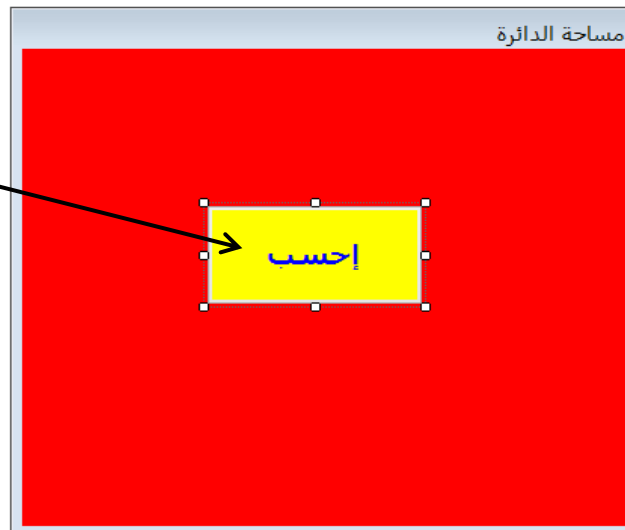


Not the value of the (Location) property; it specifies the new position of the Button on the (Form).

Set "BackColor - Font - ForeColor" property



The background color of button is Yellow and the color of writing is Blue

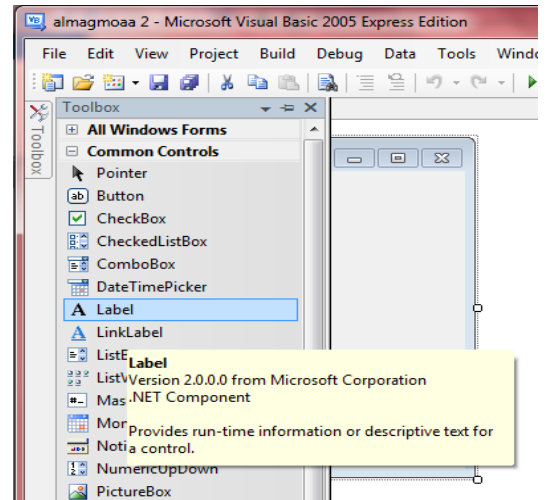


NOTICE

- A **Button** is one of the (Controls) that can be (placed) drawn on the (Form).
- A user will use a button by **Clicking** on it to perform a specific task.
- To place a Command Button on the form; in design mode, From "**Toolbox**" double-click the "**Button**" icon.
- You can change the "**Location**" properties of the Button on the "**Form**" by pressing the left mouse tab on the Button then Drag and Drop it.
- You can change the "**Size**" properties of the Button on the "**Form**" by using Eight boxes (Sizing handles) are shown at the Borders and corners of the (Button).
- You can also change the "**Size**" and "**Location**" properties from "**Windows Properties**" of "**Button**".
- **To Move the (Button):** place the mouse pointer inside the (Button) and hold down the left mouse button, drag the (Button) to the desired position (الوضع المطلوب) and then drop the mouse button.
- The value of the (**Location**) property (**30;100**); it specifies the **position** of the Button on the (Form).
- The value of the (**Size**) property (**121; 62**); it specifies the **width** and **height** of the Button on the (Form).

The Label:

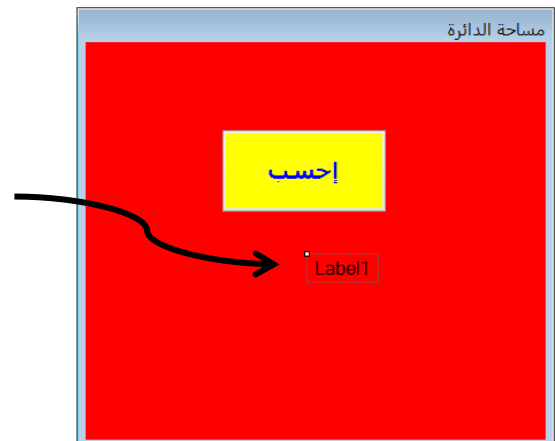
- A **Label** is a control used to provide the user with information.
- It is used in showing a Text on the Form Window which can't be changed during program Runtime.



The Properties of Label :

No	Property	Function
1	AutoSize	Specifies whether (يحدد ما إذا كان) the size of the control (Label) is automatically adjusted by text written
2	BorderStyle	Specifies the border style of the control (Label)

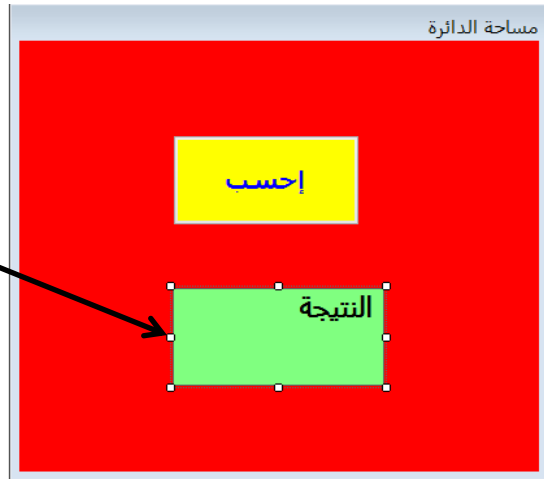
- If the "**AutoSize**" property is set to "**False**", you can manually adjust the size of the label.
- If the "**AutoSize**" property is set to "**True**", the label size is automatically adjusted to fit the text displayed on the label.
- The control (**Label**) has a set of properties like (**Name "lbl" - Text - Font - ForeColor - BackColor - Size - Location - RightToLeft - Image**)



Set "BackColor - Font - ForeColor" property

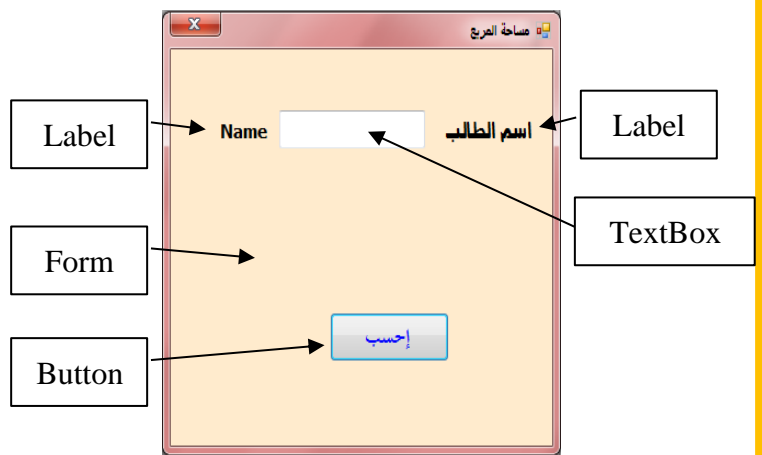
Property	Value
Name	lblResult
Text	"النتيجة:"
ForeColor	Choose your favorite color
BackColor	Choose your favorite color (green)
Font	Choose the font's type, style, and size you prefer.
AutoSize	False
BorderStyle	FixedSingle

The "Label" after setting the properties



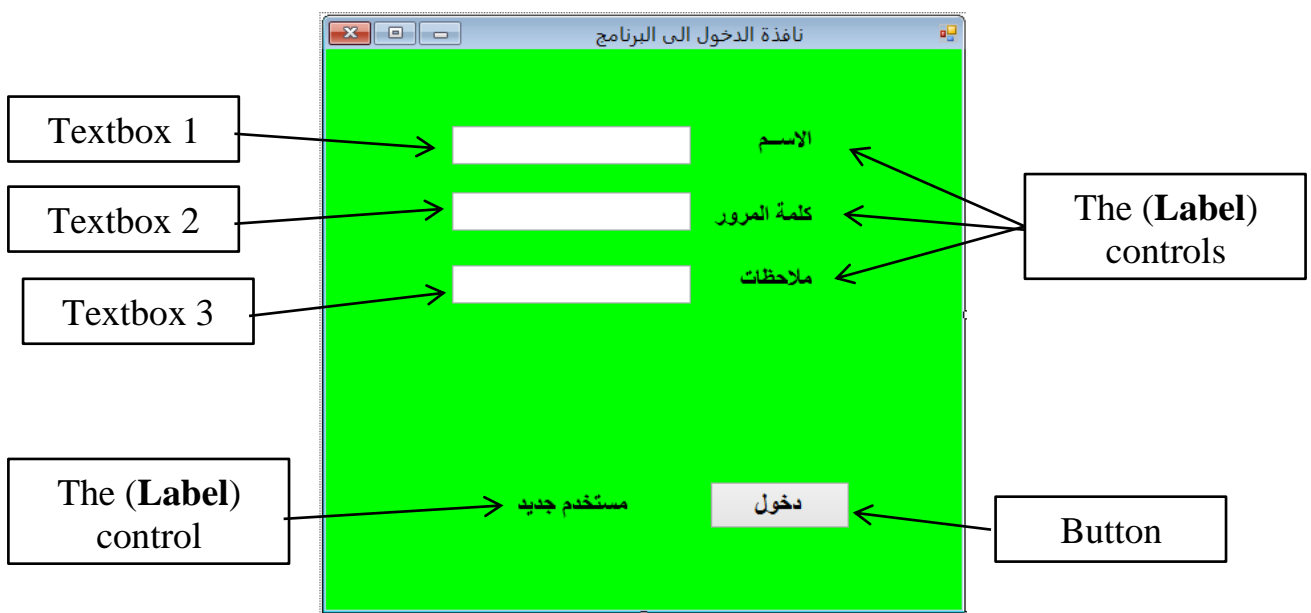
The TextBox:

- ♠ A "TextBox" control can be used for both entering data from the User and displaying results.
- ♠ The control (TextBox) has a set of properties like:



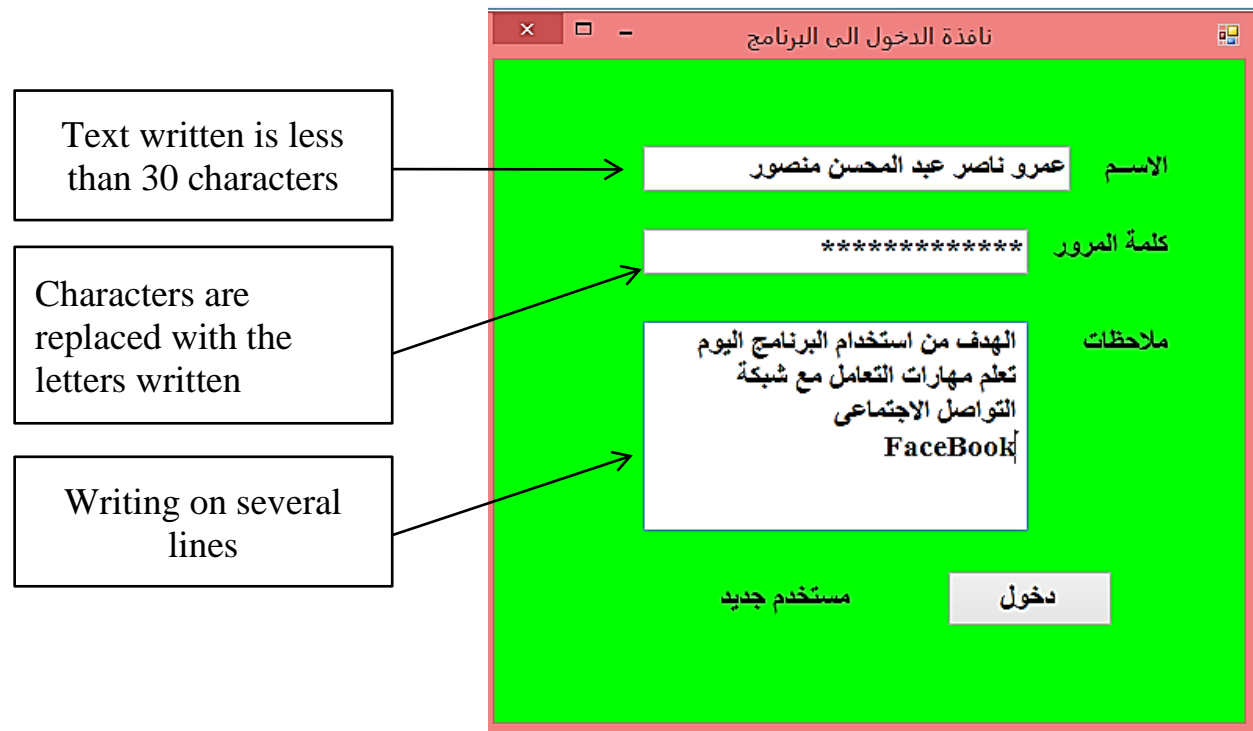
(Name "txt" - Text - Font - ForeColor - Visible - Size - Location - RightToLeft - Enabled)

Set "TextBox" properties



Set the values of properties as follows the table:

Textbox 1 →	Property	value
	MaxLength	30
Textbox 2 →	Property	value
	PasswordChar	*
Textbox 3 →	Property	value
	MultiLine	True

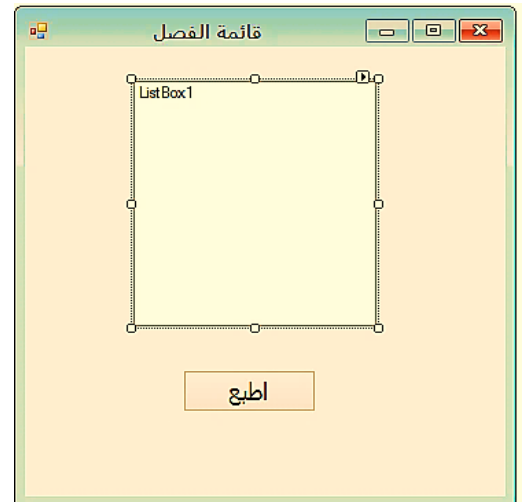


The Properties of "TextBox" :

Property	Function
MaxLength	Specifies the maximum number of characters that user can write in the (TextBox) . its value (10) for example
PasswordChar	Specifies the symbol that will be displayed instead of the text written; as example: creating a Password , its value (*)
MultiLine	Determines whether the (TextBox) control allows multiple lines, its value (True) or (False)

Listbox

- ✂ A "Listbox" control is used for displaying a list of items.
- ✂ Each item in the "Listbox" is written in a separate line.
- ✂ The control (Listbox) has a set of properties like (*Name* – *Visible* – *Size* – *RightToLeft* – *ForeColor* – *Font* – *BorderStyle*).
- ✂ The "Listbox" has not "Text" property.
- ✂ When you make the value of (*SelectionMode*) property is "*MultiExtended*" , you can select more than one item in the "Listbox"



Set the values of properties as follows the table:

Property	Value
Items	عمرو رامي ياسمين سحر
SelectionMode	MultiExtended
Sorted	True

Items (Collection) ... Press this button

This window appears

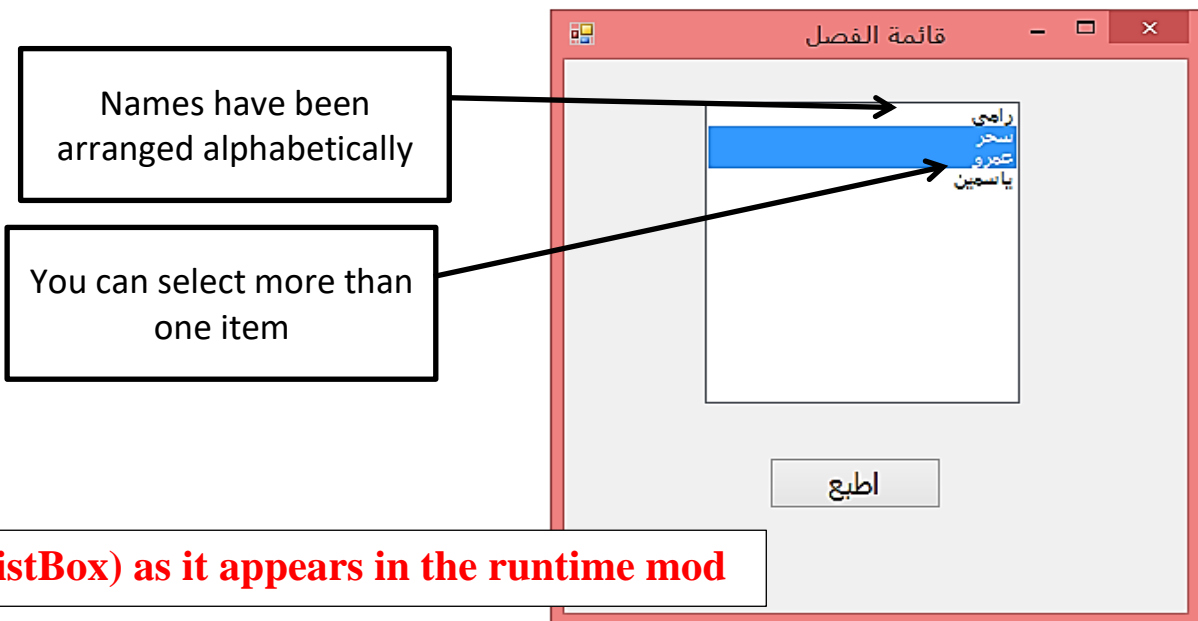
String Collection Editor

Enter the strings in the collection (one per line):

عمرو
رامي
ياسمين
سحر

Note that each item in the (Listbox) is written in a separate line

OK Cancel



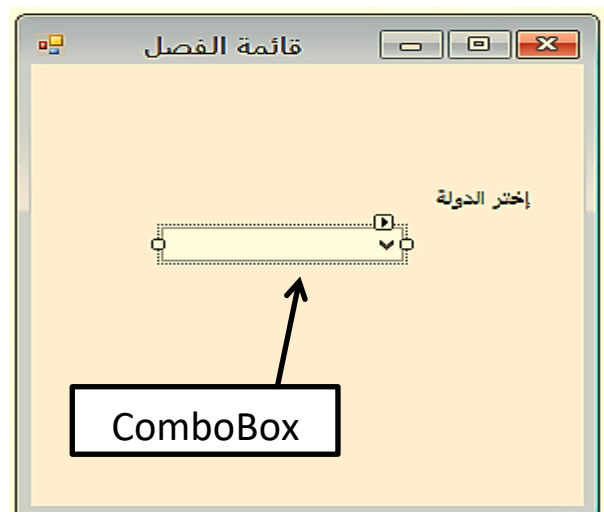
(**ListBox**) as it appears in the runtime mod

The Properties of "ListBox" :

Property	Function
Items	Presents a set of items displayed in the (ListBox)
Sorted	Specify whether the items are arranged (<i>alphabetically</i>) or not, and its value is "True" or "False"
SelectionMode	Determine whether it is possible to select one or more item displayed in the (ListBox).

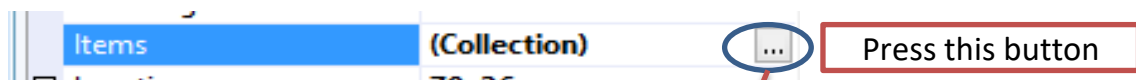
ComboBox

- ⚙️ A "**ComboBox**" control displays a drop-down list from which one item can be selected.
- ⚙️ The control (**ComboBox**) has a set of properties like (*Name - Visible - Size - Location - RightToLeft - ForeColor - Font - BorderStyle - Items*).
- ⚙️ The "**CompoBox**" has "Text" and "Items" property

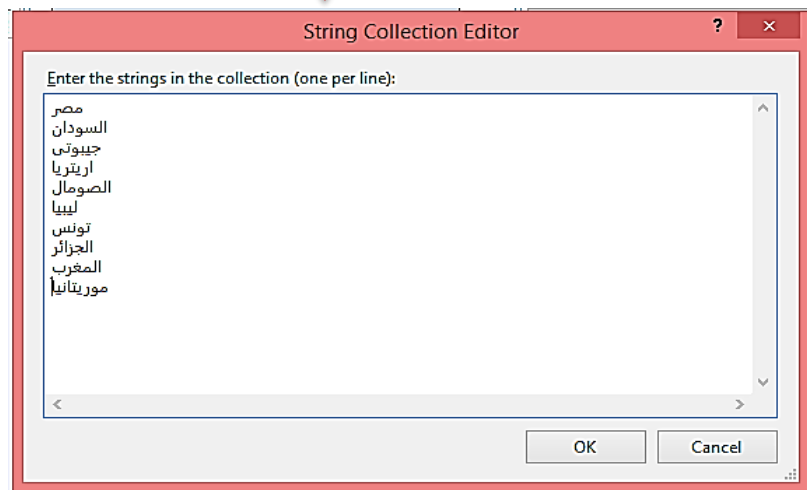


Set the values of properties as follows the table:

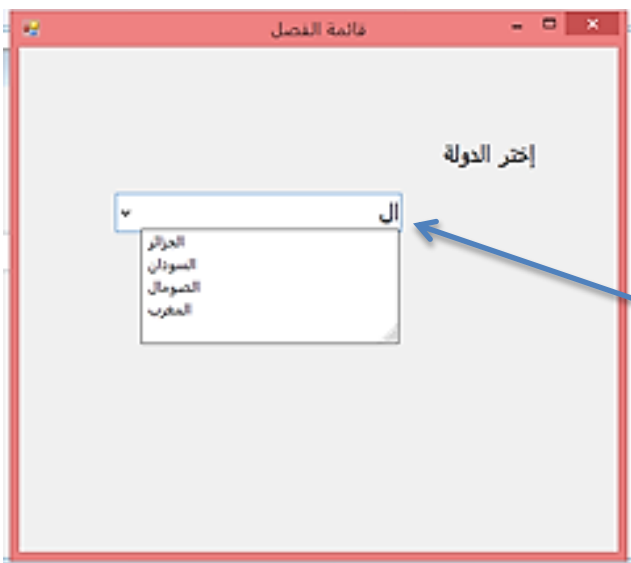
Property	Value
Items	"مصر السودان جيبوتي اريتريا الصومال ليبيا تونس الجزائر المغرب موريتانيا" Note that :each country name ,should be written in a separate line
AutoCompleteMode	Suggest
AutCompleteSource	ListItems



This window appears



Note that each item in the (ComboBox) is written in a separate line



Note that when start typing "ال" all the proposed counties that begin with "ال" are displayed

(ComboBox) control in the run-time mode

The Properties of "CompoBox" :

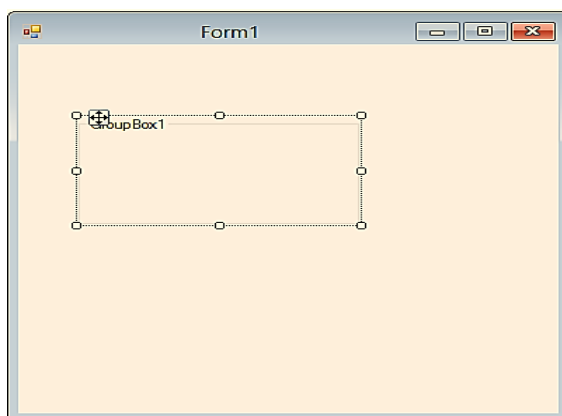
Property	Function
Items	Presents the items in the (ComboBox)
AutCompleteSource	The maintained source of items used for automatic completion of input string.
AutoCompleteMode	The input string or (prefix being entered) that will be compared to the prefixes of all strings in a maintained source; upon which the automatic completion will be done.

GroupBox

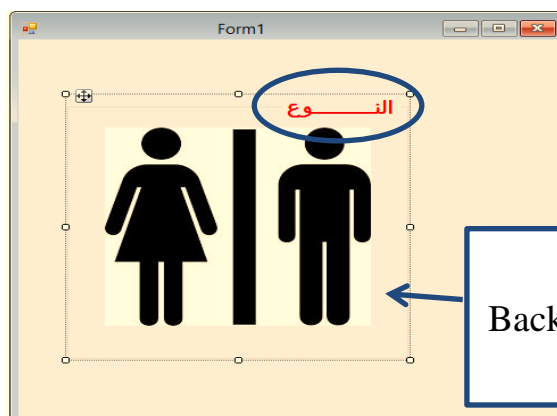
- A "**GroupBox**" control is used to group other controls of same function together on the Form window.
- The control "**GroupBox**" has a set of properties like (**Name** – **Text** – **ForeColor** – **BackgroundImage** – **Enabled** – **Visible** – **Size** – **Location** – **RightToLeft** – **Font** – **BackColor**).
- The property (*BackgroundImage*) is used to display the image in the background of the control "**GroupBox**".

Example:

No.	Property	Value
1	Text	النوع
2	ForeColor	Choose your favorite color
3	RightToLeft	Yes



The **GroupBox** before setting the Properties

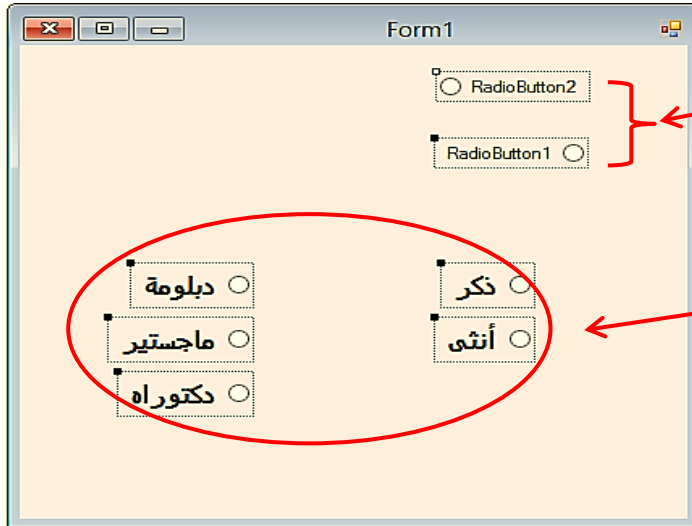


The
BackGroundImage
Property

The **GroupBox** after setting **BackgroundImage** Properties

RadioButton

- ♣ A "**RadioButton**" is used to select one option from a group of mutually exclusive options.
- ♣ The user selects one alternative only.



(**RadioButton**) control before setting the properties

A set of (**RadioButton**) where the "Text" property have been adjusted as shown in Figure

Property	Value
Checked	True
RightToLeft	Yes

- ♣ The effect of many (*Properties*) is not shown during *Design Mode*, but in *Run-time Mode*.

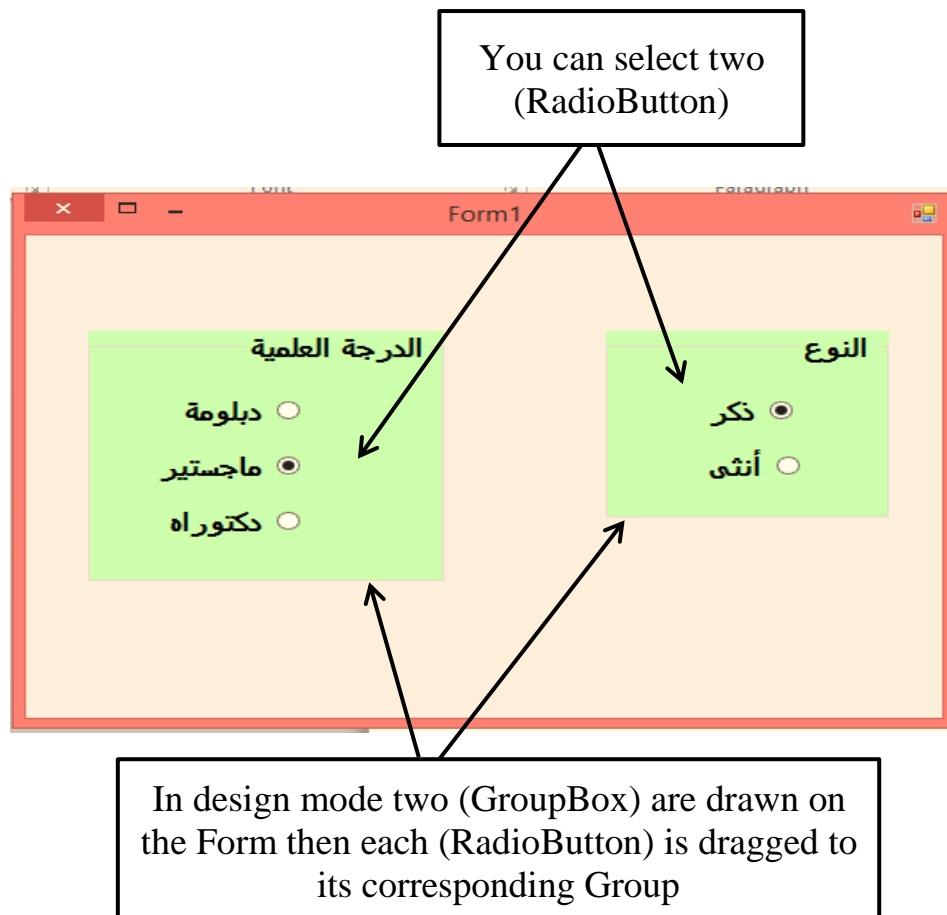


Note that the (**RadioButton**) is the one active only



Note:

The previous figure asks the user to choose the **Gender**, and the **Scientific Degree** as well; but (RadioButton) control doesn't allow this. **To solve this problem**, we set two (GroupBox) controls; the first one contains the **Gender** while the second one contains the **Scientific Degree**

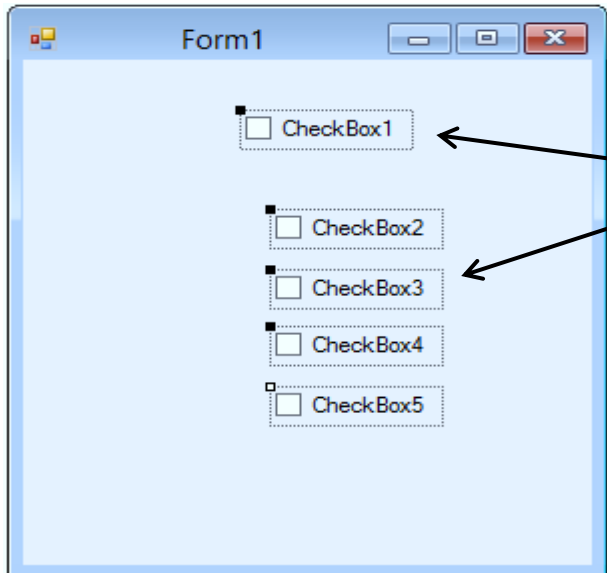
**The Properties of "RadioButton":**

No	Property	Function
1	Checked	Indicates if the (RadioButton) has been selected or not, (take a value " True" or "False")
2	Text	The text displayed on the (RadioButton)

- ♣ The control "RadioButton" has a set of properties like (**Name** – **Visible** – **Size** – **Location** – **RightToLeft** – **ForeColor** – **Font** – **Enabled** – **BackColor**)

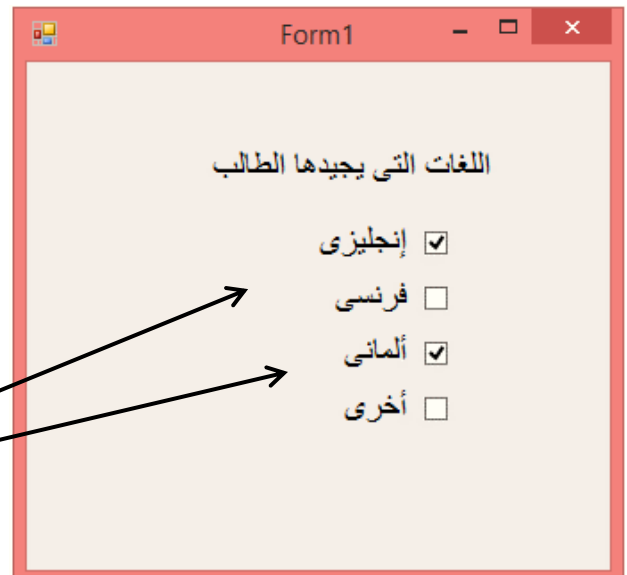
CheckBox

- A "CheckBox" control is used to select one or more options.
- The control "CheckBox" has a set of properties like (**Name** – **Visible** – **Size** – **Text** – **Checked** – **Font** – **ForeColor** – **RightToLeft** – **Location**).



A set of (CheckBox) before setting the properties

A set of (CheckBox) controls after setting the "Text – RightToLeft – Font" properties. You can select more than one item



The Properties of CheckBox:

No	Property	Function
1	Checked	Indicates if the (RadioButton) or (CheckBox) has been selected or not, (take a value " True" or " False")
2	Text	The text displayed on the (RadioButton) or (CheckBox)



The Questions

Q1 : State whether the following statements are true (T) or false (F):

1. The function of the property RightToLeft of the Form is to define the direction of Controls from right to left. ()
2. The function of the property RightToLeft of the Form is to define the state of the Form on the screen in a position of Maximizing or Minimizing. ()
3. Setting the property ControlBox of the Form can control the Form in a position of Maximizing during programe runtime. ()
4. The property "Name" is used in showing a certain Text in the title bar of user window a name of the window. ()
5. The property Text is used in showing a certain text the title bar of of user window. ()
6. Setting some properties of the Form is applied to Controls which are placed on the Form. ()
7. The effect of setting the WindowState property of the form appears only in runtime mode. ()
8. You can change the location of Command Button on the Form through Size property. ()
9. You can change the location of Command Button on the Form through Location property. ()
10. Placing Controls automatically on the Form on the co-ordinate (0-0) is in the middle of the Form. ()
11. You can change the size of Label manually if AutoSize=true. ()
12. You can change the size of Label manually if AutoSize=false". ()
13. Textbox control tool: is the only tool which has the property passwordChar".()
14. Textbox control tool: is the only tool which has the property AutoSize. ()
15. ListBox and ComboBox share in "Item" property. ()
16. ListBox and and ComboBox share in "Suggest" property. ()
17. GroupBox is the tool used in contains a group of controls, these controls have the same function on the Form. ()
18. ListBox is the tool used in contains a group of controls, these controls have the same function on the Form. ()
19. CheckBox can be used on the Form to choose the Gender of student male or female. ()

20. Combobox is the control tool that allows the user to choose one element of several elements in the smallest possible space on the form window. ()

Q 2 : 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 maximizing or minimizing.

2) ControlBox property of the Form is helping to:

- a) Showing or hiding of Maximizing Box.
- b) Control the appearance of the Form whether it is in a position of Minimizing / Maximizing / Normal.
- c) Control the appearance or disappearance of ControlBox in the Form.

3) The used property in showing a certain Text on the titlebar of a Form is :

- a- Name b-Text c-FormBorderStyle

4) On setting some properties of the Form, they are applied on Control Tools

Which are placed on the Form (one of them is):

- a-Name b-Forecolor c-Text

5) The effect of setting this property doesn't appear unless in runtime mode (This property is):

- a-FormBorderStyle b-WindowState c-RightToLeft

6) The property which is responsible for the size, shape and effect of the Text font shown on the Button is :

- a-BackColor b-Forecolor c-Font

7) You can change the position of the Button on the Form through the following processes except for :

- a) Drag and drop by the mouse
- b) setting Size property
- c) setting Location property

8) You can change the position of the Button on the Form through :

- a) setting Location property
- b) setting Size property
- c) The eight squares around the Button

- 9) 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
- 10) The size of Label is defined automatically on the Form if the property is:
- a-AutoSize = False b- BorderStyle= FixedSingle c- AutoSize=True.
- 11) You can change the size of control "Label" manually if the property is :
- a- AutoSize = False- b- BorderStyle= FixedSingle c- AutoSize=True
- 12) The following properties belong to TextBox except for:
- a-AutoSize b-MultiLine c-MaxLength
- 13) The Object TextBox is marked by one property :
- a-AutoSize b-Name c-PasswordChart
- 14) The right value which can be used to set the PasswordChart of the TextBox is :
- a-Pw b-True c- *
- 15) The ListBox and ComboBox share in this property :
- a-Suggest b-Item c-SelectionMode
- 16) The ControlTool which is used in containing a group of controls that have the same function on the Form is:
- a-ComboBox b-ListBox c-GroupBox
- 17) The Control tool which can be used on the Form to choose Gender of the student "male"or "female" is:
- a-RadioButton b-CheckBox c-TextBox
- 18) The ControlTool which can be used on the Form and allows the user to choose more than one item is :
- a-RadioButton b-GroupBox c-CheckBox
- 19) The ControlTool which allows the user to choose one item of 15 in the smallest possible area on the Form is:
- a- ComboBox b- ListBox c- RadioButton



Chapter Four

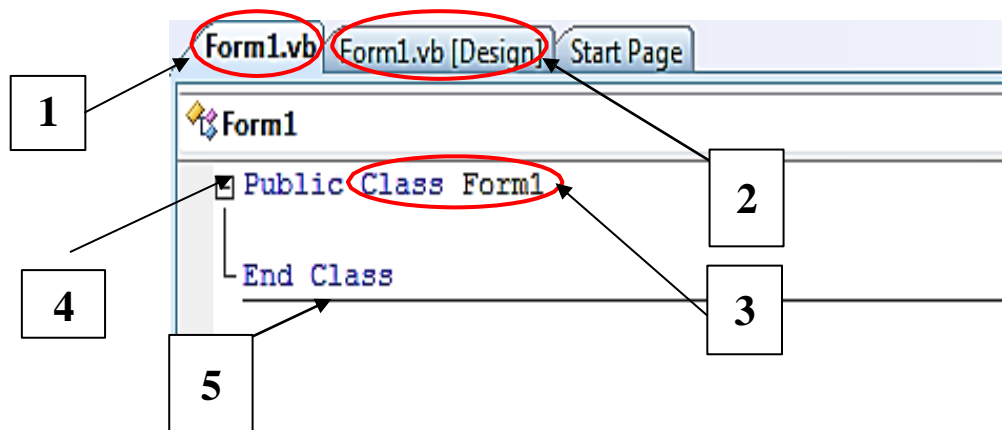
Code Window

Code window :

Through the Code window; we can write instructions and codes using (Visual Basic .Net) language.

To open Code Window of "Form1":

1. Active **Form1** window then Press **F7** key from Keyboard. **OR**
2. Double click on **Form1** window.



- (1) Name of the file where codes are saved.
- (2) Name of the file where the Form window 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.

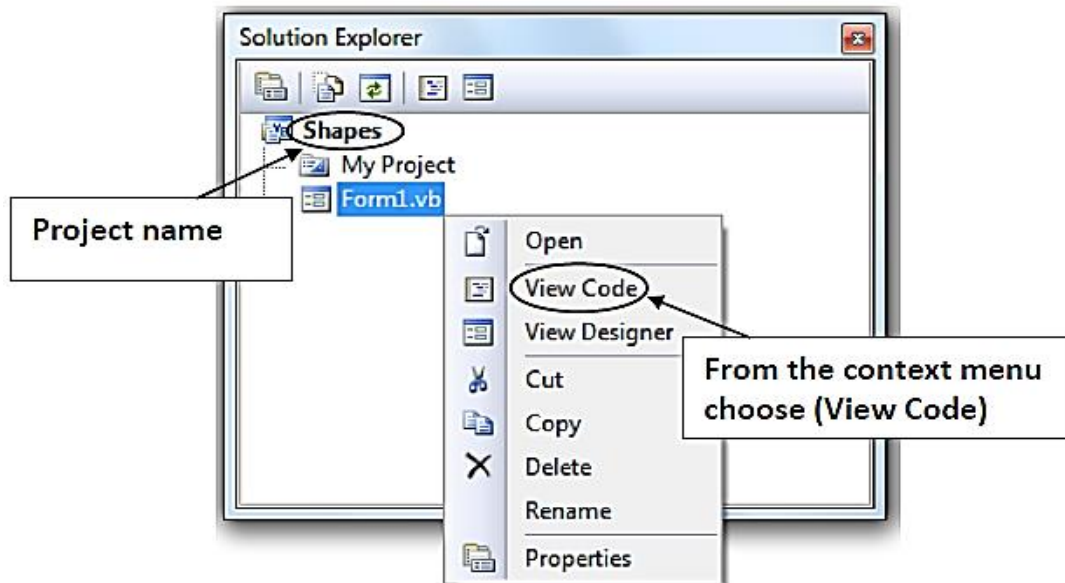
Event Handler :

1. It's a procedure which contains a code that is carried out when a corresponding event occurs. **OR**
2. Event handler is the procedure (called into action) when an event occurs.

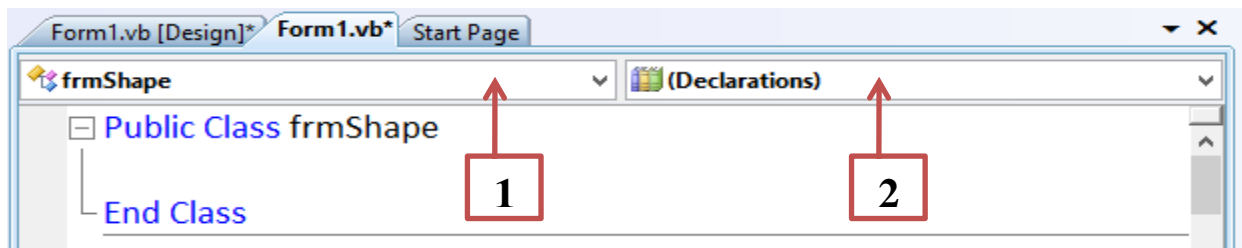
(معالج الحدث Event Handler هو الإجراء الذي يتم استدعائه عند وقوع حدث معين)

To create event handler do the following steps:

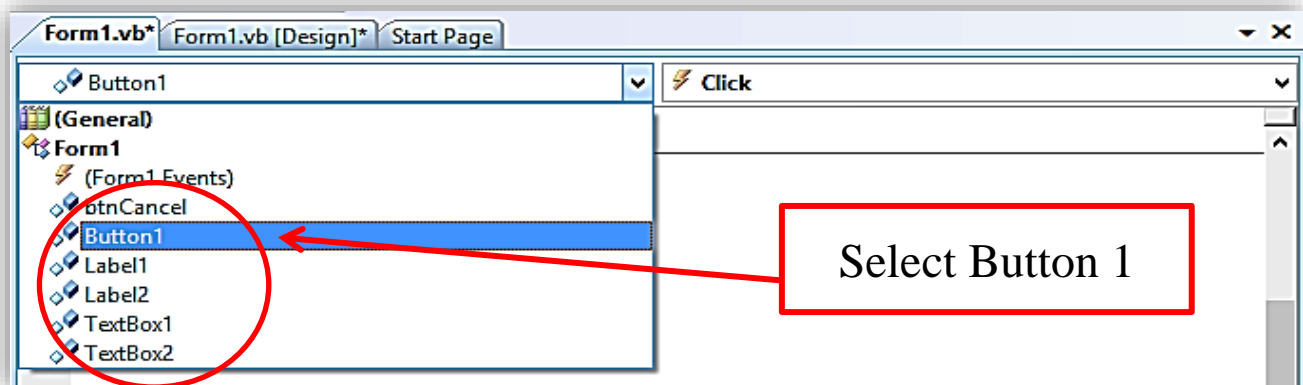
1. In the (Solution Explorer) window, right click the file (Form1.vb) and, select (View Code) from the context menu.



When choosing (View Code) the following window appears

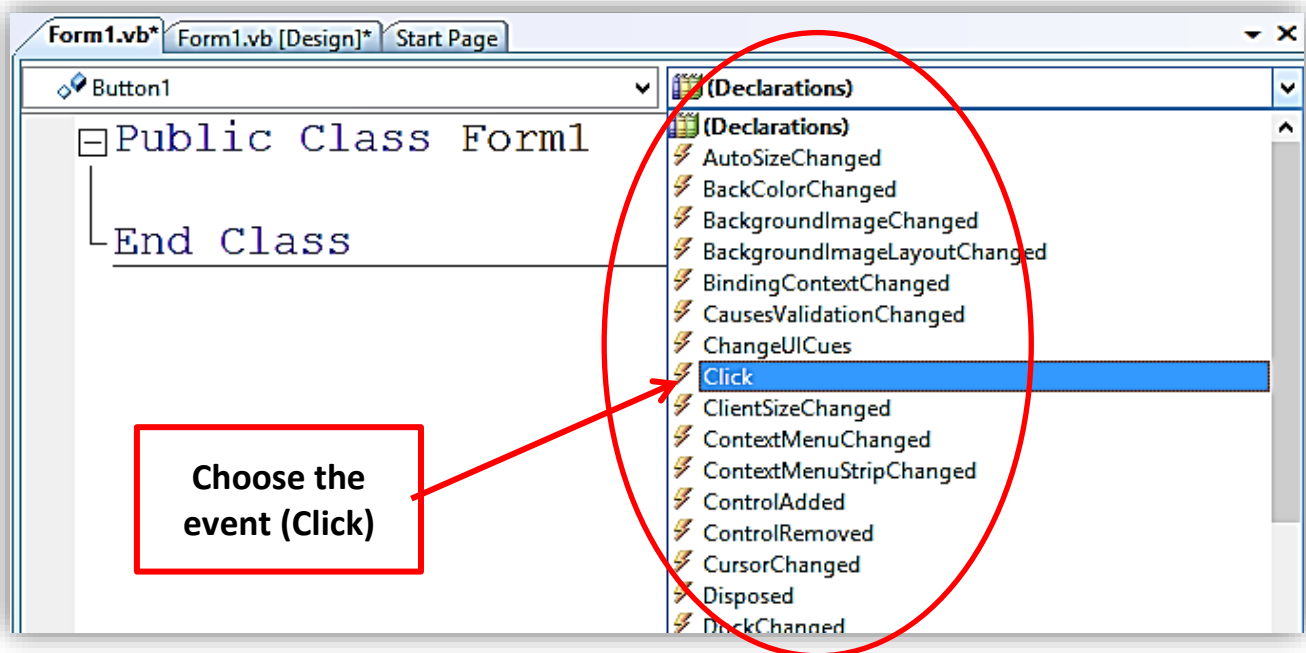


- (1) A drop-down menu of (Class Names), which refers to the names of controls placed on the Form.
- (2) A drop-down menu of (Method Names) or events; associated with the Class selected from the (Class Names) menu.

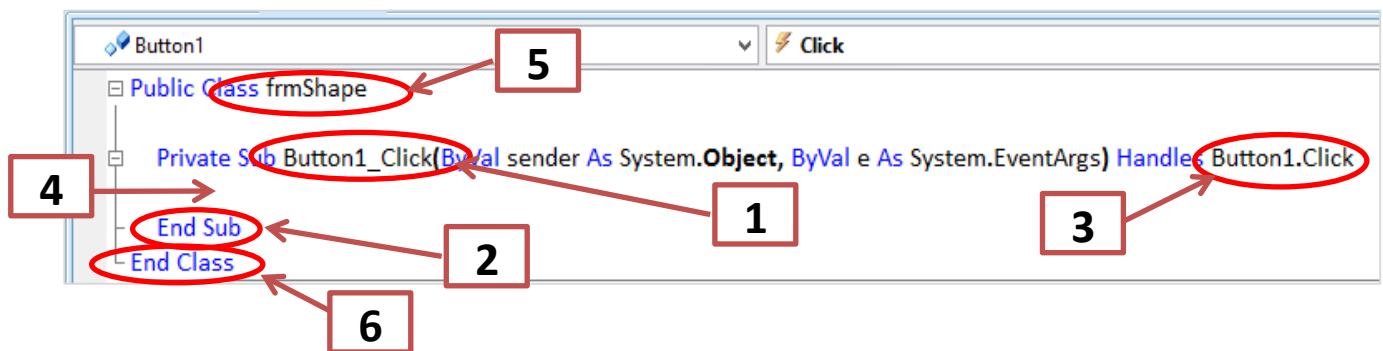


2. Open (Class Name) menu and note that the default names of the controls are listed without setting (Name property) of controls included on the Form.

3. When you select (Button1) from the Class menu drop down list, open (Method name) menu it displays the events associated with (Button1)



4. (Event handler) as shown in figure



- (1) The procedure name composed of (object name, event name)
- (2) End of procedure
- (3) What causes the call of the procedure (event occurrence).
ما هي أسباب استدعاء الإجراء (وقوع الحدث)
- (4) Between the two lines shown; you can write statements or codes that will be executed after invoking the procedure.
- (5) The declaration of the Class (frmSquare)
- (6) The end of the Class.

Setting the "Properties" programmatically :

ControlName. Property = Value

Button1 – lblTitle –
label2 – TextBox1

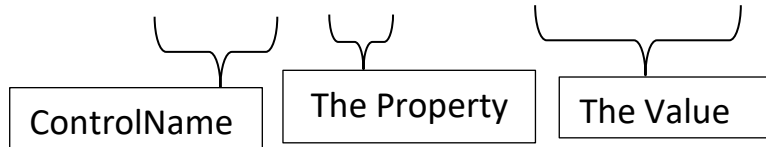
Text – ForeColor –
Visible – Font

"مصر" – Color.Red – True
–New Font ("Arial", 30)

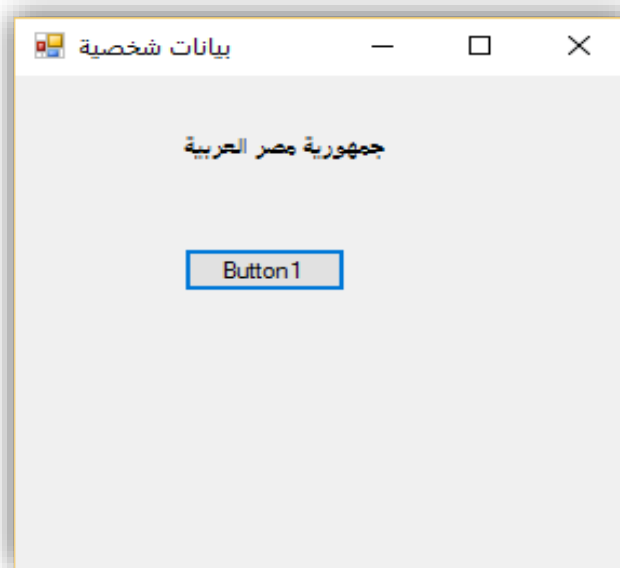
Examples:

- ❖ Adjust the property (Text) for the Label (lbl_Title) to be : "العربية مصر جمهورية"
The statement is:

lbl_Title.Text = "العربية مصر جمهورية"

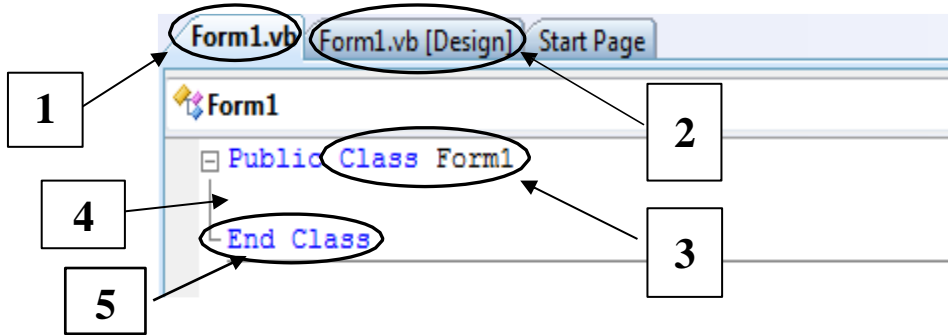


```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.
    lbl_Title.Text = "جمهورية مصر العربية"
End Sub
```



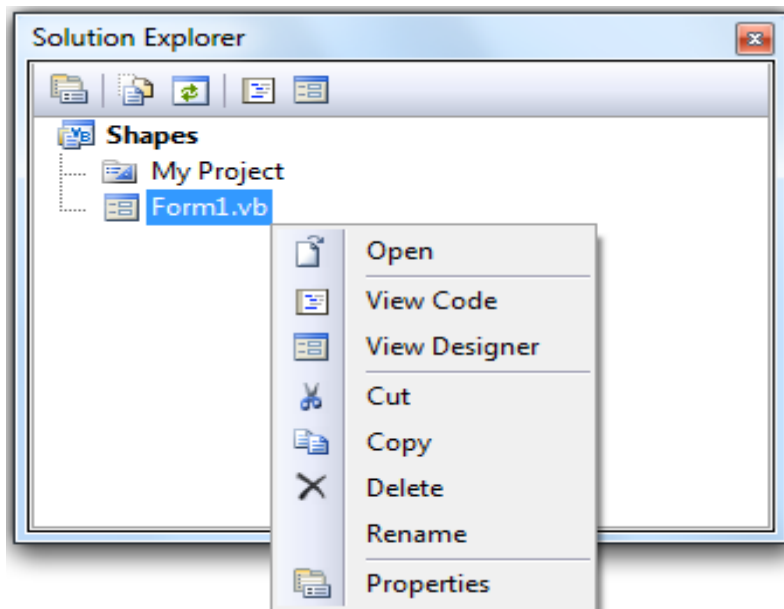
The Questions

Q 1 : In the shown figure what the following numbers indicate?



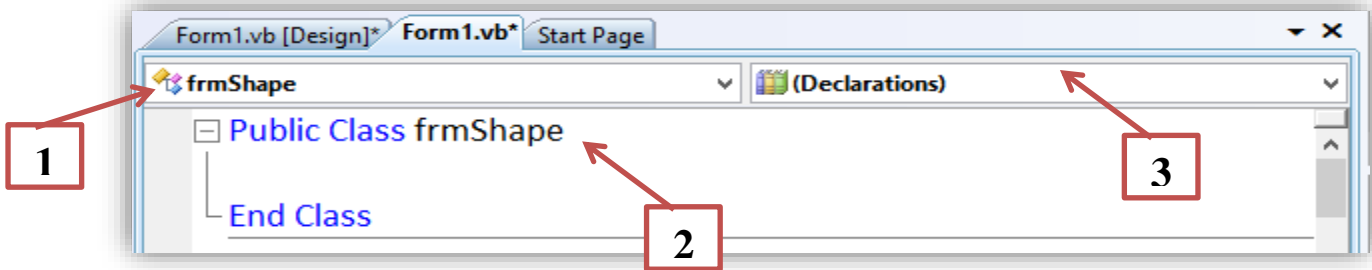
- 1)
- 2)
- 3)
- 4)
- 5)

Q 2: complete the table using the next screen :

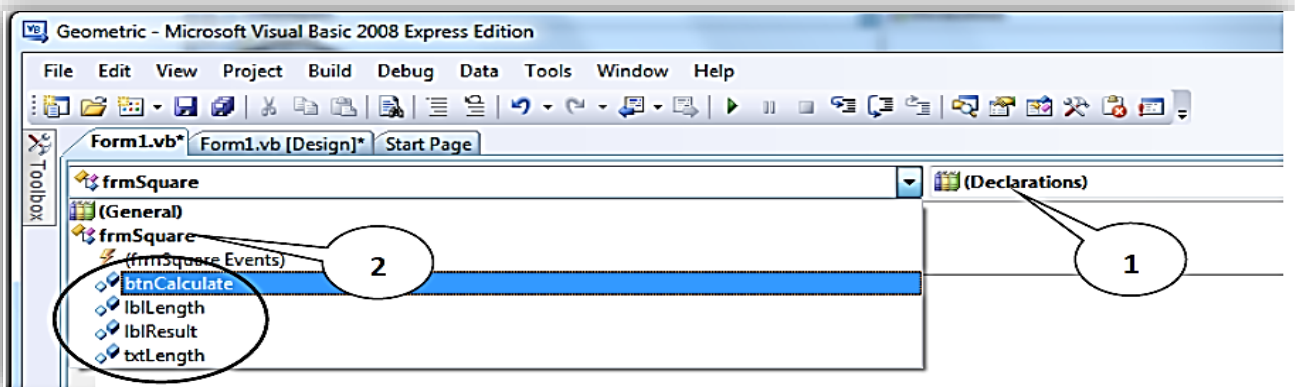


Question	indicates
1	The name of the solution is
2	The name of the project is.....
3	We can enter the code window more than one way, through : <ul style="list-style-type: none"> • Command in the shortcut menu • function key
4	The purpose of properties in the shortcut menu is

Q 3: Complete the table using the following screen :



No	indicates
1	-----
2	-----
3	-----



1. The name of forms is
2. Number (1) refers to

NO.	indicates
1
2
3
4
5
6

Q 7: the general syntax to adjust the properties of controls programmatically:

ControlName.Property = Value

Explain the following codes through your pervious the general syntax

A.) Button2.Text = "END"

.....

B.) Label1.AutoSize = True

.....



Good Luck