Chapter Two

Introduction to Visual Basic.NET

By the end of this chapter, student will be able to:

- Identify the essence of Visual Basic .NET
- Identify the basic components of the IDE screen
- Create a new project
- Add a new Form to the (Project)
- Save the project
- Add a new project to the solution

After studying the Problem solving technique, and learning how to develop logical steps to solve problems, in this chapter you will start working with Visual Basic.Net Language, and you will be able to convert the solution steps of a problem to program codes that can be executed.

2–1 Visual Basic.net Programming Language

Visual Basic .NET is just one of the languages in Visual Studio .NET package that includes other languages, such as C# and J#.

Visual Basic .NET is an object-oriented language that develops event driven Windows and Web applications.

2–1–1 Programming Language

Is a set of rules, symbols and special words you can use to write instructions and construct a computer program; according to the programming language used. Instructions will be translated to machine language for being executed.

A Computer executes only commands written in machine language. As for programmers; they can't write in machine language, so they use programming languages to write programs in English; then comes the role of the compiler (found in the language) that translates program' instructions from English to machine language; for a Computer to understand.

2–1–2 (Visual Basic.Net) is used to create Windows applications

A Windows-based application has a Graphical User Interface (GUI) and appears in a window .Sure you used many Windows-based applications, like Paint , Notepad ,Calculator ,Internet browser ... Etc. Notice that all Windows applications have a graphical interface (a window) that share some common characteristics: such as "Window style, Maximize button, Minimize button and, saving or opening files ... Etc..

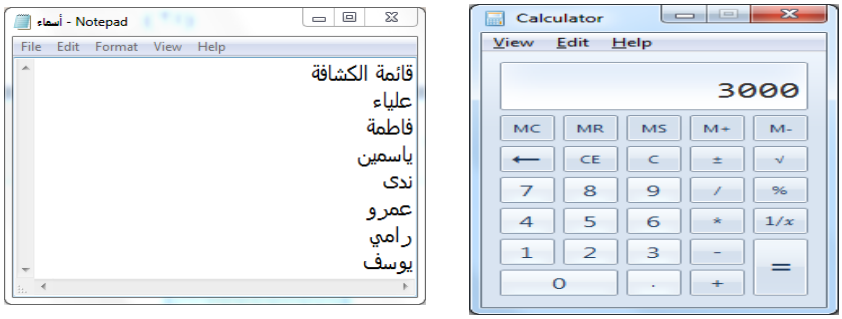


Figure (2-1) Examples of Windows applications

2–1–3 Windows applications are event driven applications:

When using windows applications, like the calculator ; you do an action (event) like pressing a plus (+) sign or an equal (=) sign in the application or from the keyboard, then a specified task will be executed ;so writing programs using programming languages (as mentioned before) is important for responding to certain event.

2–1–4 Visual Basic.Net is an Object Oriented Language

In Visual Basic.Net everything depend on Objects (like: Button, Textbox, ComboBox); which have the following attributes:

1- **Properties that describe the Object**
2- **Events that occur to objects**
3- **Methods that present actions to be performed on objects ; causing certain behavior on the objects ;the Object (Textbox);the Method**

invoked on it is: Copy (which is responsible for copying selected text in the clipboard).

Example

The calculator, is an application where we find many buttons; each button presents an (Object) it has (Properties) like: width, height, text labeled, background color ….etc., and has events like: (Click) that occurs when the user clicks the mouse on a specified button on the calculator, which in turn invokes certain action; if the same event (click) is done on other objects (buttons) it may cause different actions.

This means that all buttons have the same properties with different data values, and then appear with different shapes. The same event (click) if occurred on each button in the calculator will cause different actions, which vary from one button to another, although all buttons are derived from the same (Class).

From the above we conclude that:

2–1–4–1 Object

An (Object) is the basic constructive element in Object Oriented Programming; it is created from a defined class.

2–1–4–2 Class

A (Class) is the blueprint/ plan / template, from which the individual objects, are created. It is the blueprint that describes the details which any object takes (its Properties, Methods and, Events); that are all derived from the (Class).

NOTICE

1. The (Class) implies a definition for the (Object).
2. The (Object) exists only when an instance of the class is created
3. You can create as many objects you need from a class.
4. A place in the memory is reserved for each object in Visual Studio.NET when it is created.

2–2 .NET Framework

The .NET Framework is like the central nervous system for all Visual Basic.Net applications; it is a platform that enables you to:

1- **Develop applications like (Desktop applications) - (Web applications) – (Mobile applications).**

2- **Provide** a development environment for running all applications**.**

The Framework is composed of:

- **The execution engine (CLR) Common Language Runtime**
- **The .NET class libraries (System Class Libraries)**
- **(Compilers)**
- **Other elements**

The .NET Framework is used in the development, design and execution of .NET applications. The .NET Framework can be installed for free on operating systems (and is available with many versions).

2–3 Main elements of (IDE) screen

The term (IDE) refers to Integrated Development Environment

Visual Studio .NET provides an environment known as (IDE) that enables the developer to do as much as possible with visual tools, to quickly design applications (Windows applications) - (Web applications) – (Mobile applications).as shown in figure (2-2).

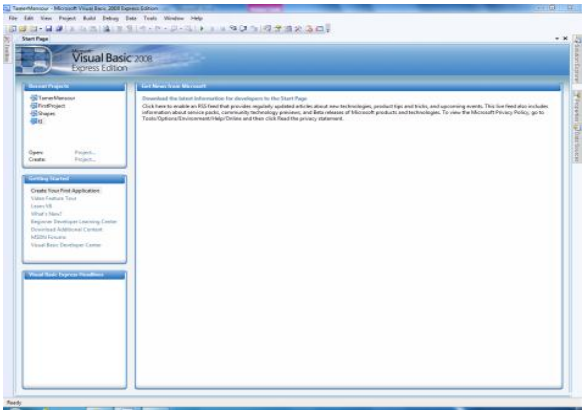


Figure (2-2) (IDE) screen

2–3–1 Form

The form is the window (visible interface) of the application; what users will see and work with when they run this application .A form is the container upon which controls (CommandButton –Textbox- Label..etc) are placed as shown in figure(2-3).

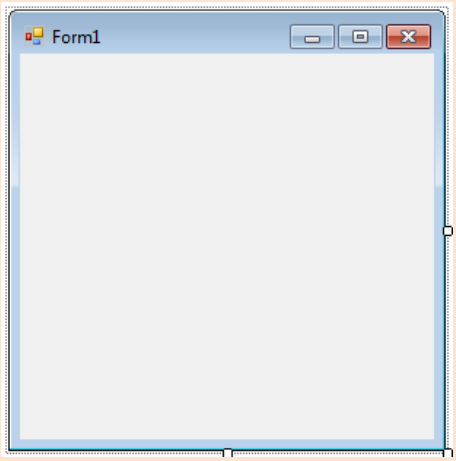
A blank Form before placing any controls on it	A Form after placing controls on it and adjusting some of their properties.
	

Figure (2-3) A Form before and after placing controls

2–3–2 Toolbox

The Toolbox contains controls (objects) that the programmer can place on the form, these controls are available in tabs (categories) as shown in figure (2-4); notice that a (+) sign is displayed with each tab, when we click on it ,the tab expands and a set of controls will be displayed.

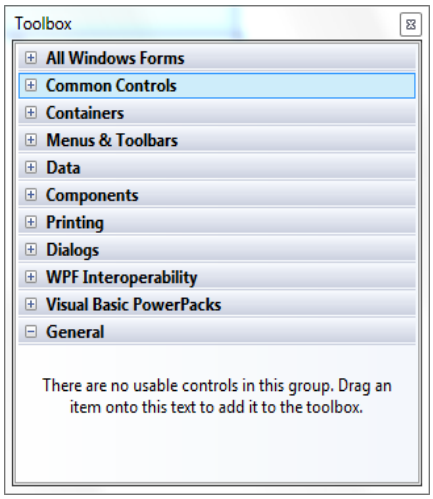


Figure (2-4) the Toolbox

You can display all the (Controls), by choosing (All Windows Forms) Tab or category.

From these categories we also have

- **(Common Controls)**
- **(Menus & Toolbars)**

A set of controls (icons) expands in each tab (category) as shown in figure (2-5)

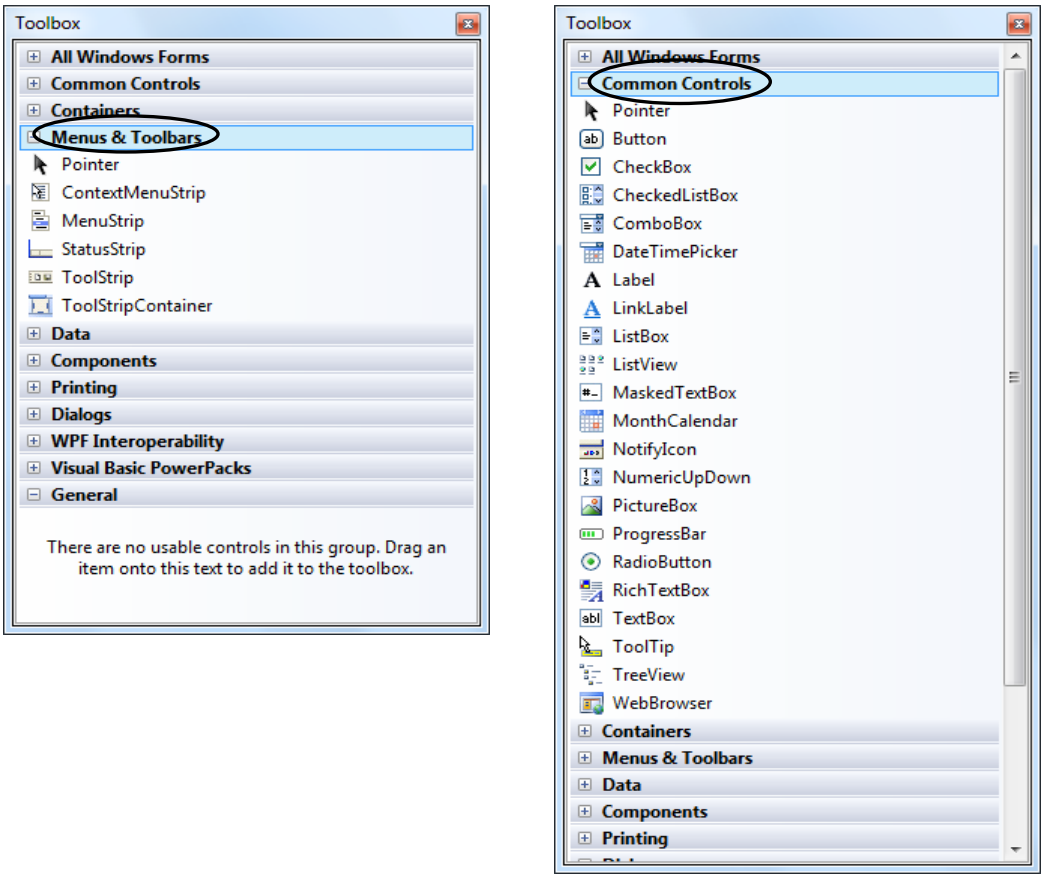


Figure (2-5) Controls in different tabs (categories)

Some of the (Common Controls)

Control	Control
ComboBox	Button
CheckBox	TextBox
RadioButton	Label
	ListBox

Table (2-1) some of the (Common Controls)

2–3–3 Properties window

Each control from the above has properties . A Properties window lists the property settings for the selected Form or control and permits changes to each setting to be made. as shown in figure (2-6).

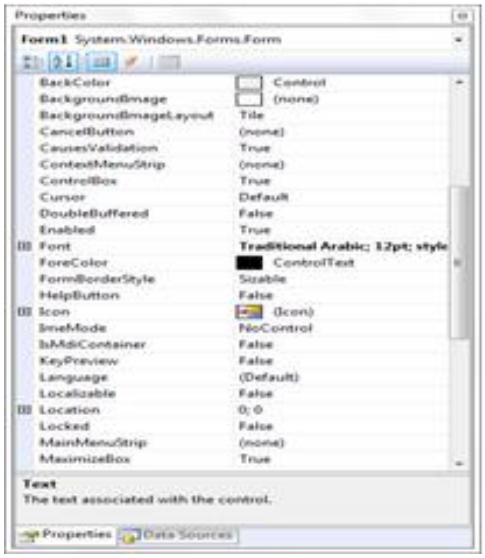


Figure (2-6) (Properties window)

NOTICE

In (IDE) screen, displayed properties differ upon the selected element.

2–3–4 Solution explorer

The Solution explorer window contains a list of items of the current solution ;and may contain one or multiple projects .It also displays a hierarchical list of all the components, (files and folders) organized by project, as shown in figure (2-7).

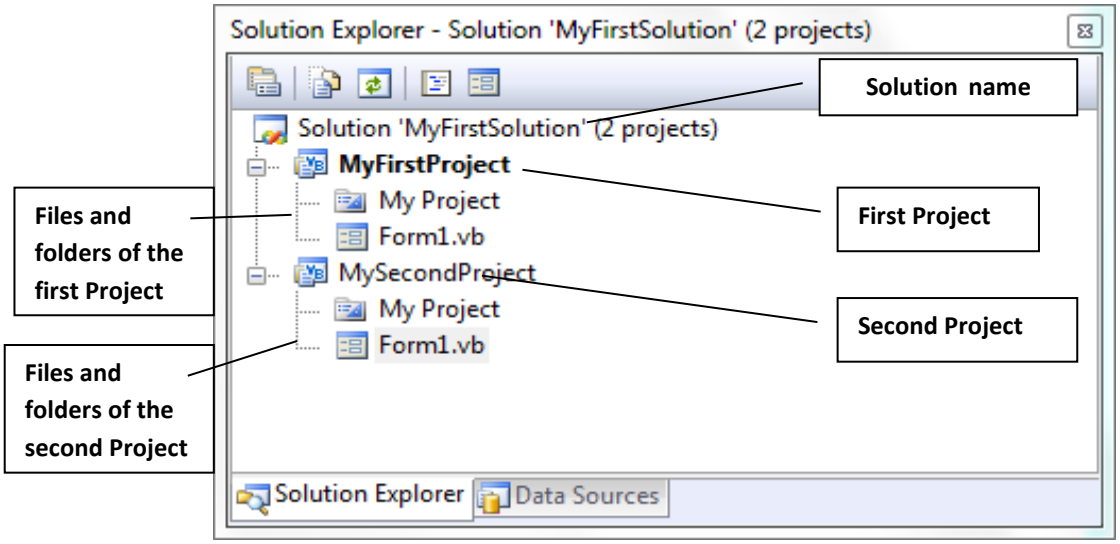


Figure (2-7) (Solution Explorer) window

Exercise (2–1)

Create a New Project

With your teacher's assisstance open the (Visual Studio),available on your computer.

Type the name of the (Visual Studio) opened on the screen.

.....

From (File) menu choose (New Project) .

A new window is displayed as shown in figure (2-8).

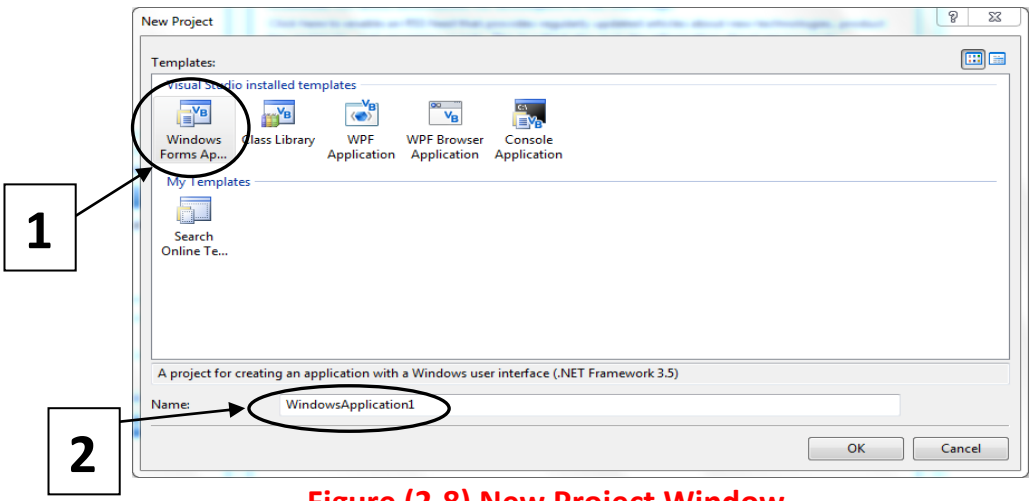


Figure (2-8) New Project Window

The numbers shown in figure (2-8) indicate:

(1) The chosen template (Windows Forms Application).

(2) The place for typing Project's name.

Type the project name (MyFirstProject) than click (OK) as shown in figure (2-9) .

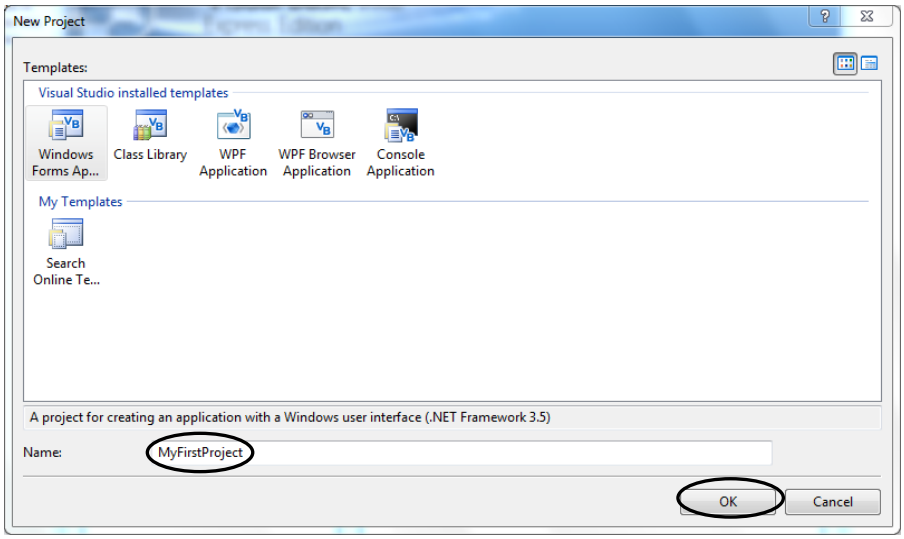


Figure (2-9) New Project Window

When clicking the (OK) button ,the (IDE) window appears as shown in figure (2-10).

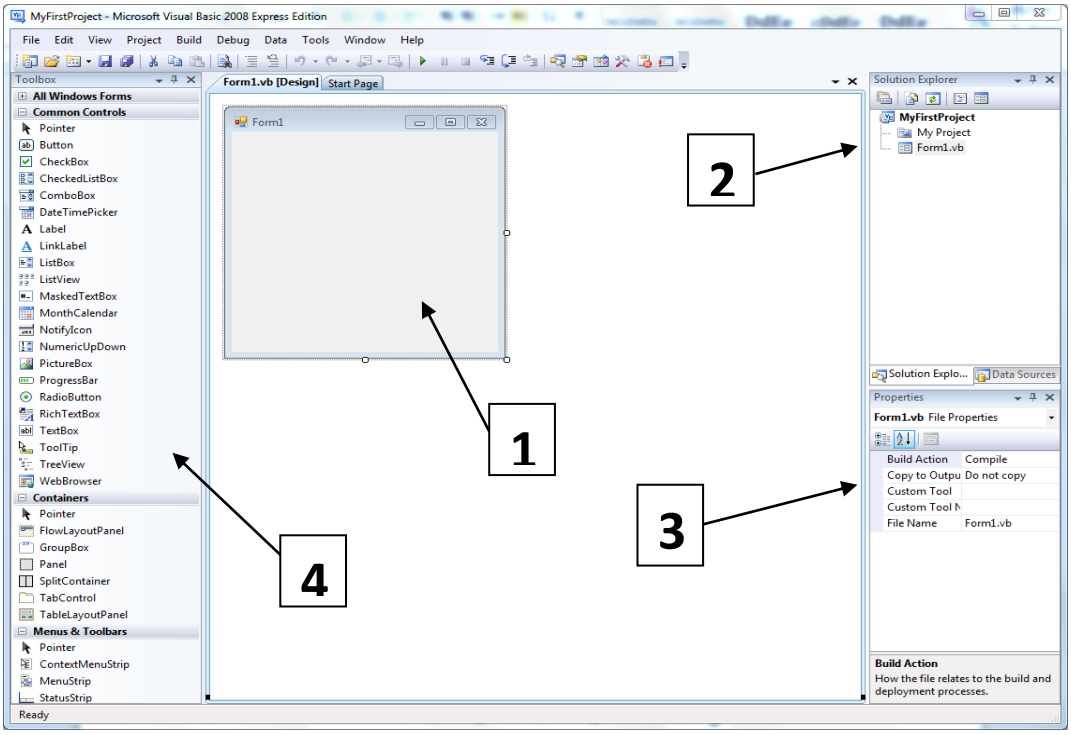


Figure (2-10) (IDE) Window

The numbers shown in figure (2-10) indicate:

(1) The Form window
(2) The Solution Explorer Window
(3) The Properties window
(4) The Toolbox window

Exercise (2–2)

Add a new (Form) to the (Project)

From the (Project) menu choose (Add Windows form) to create a new Form as shown in fig(2-11).

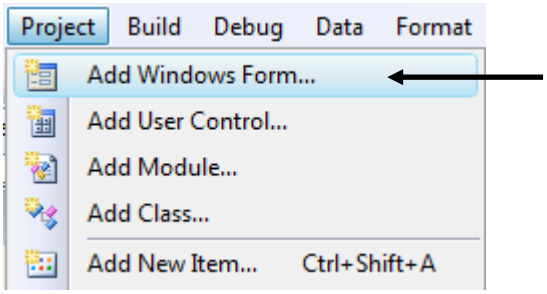


Figure (2-11) (Project) Menu

The window (Add New Item) is displayed as shown in figure (2-12).

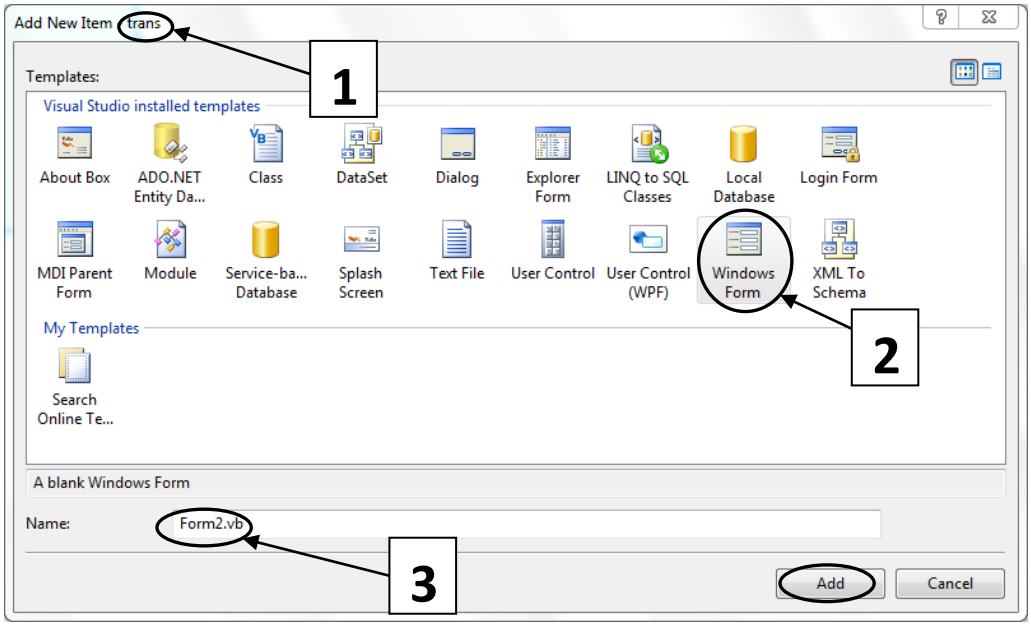


Figure (2-12) (Add New Item) Window

The numbers shown in figure (2-12) indicate:

(1) The Project name given to the project; to which the new (Form) will be added.

(2) The template used to create the (Form).

(3) The suggested file name of the (Form) (that you can change if you wish);

(4) Press the (Add) button, a new window Form will be added as shown in figure (2-13).

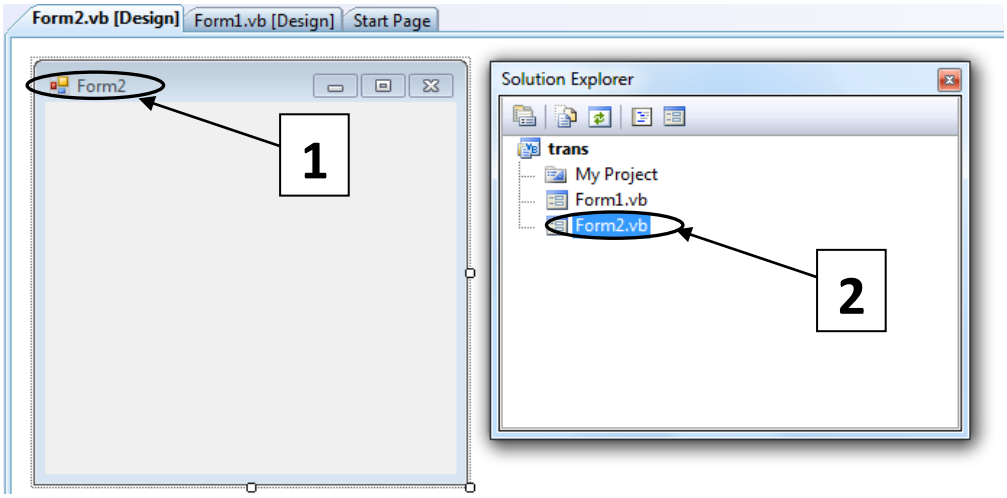


Figure (2-13) Adding a new (Form)

The numbers shown in figure (2-13) indicate:

(1) The window Form (Form2).

(2) The file name assigned to (Form2) inside the (Solution Explorer).

Exercise (2-3)

Save the (Project) in one of the storage devices

When you create a new Project, a copy of the project is saved in the memory; to save it on one of the storage devices, do the following:

Choose (File) menu then select (Save All) as shown in figure (2-14).

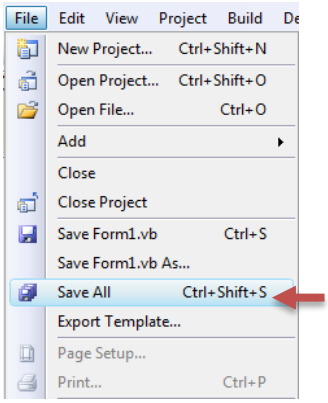


Figure (2-14) Saving the Project

The shown window will be displayed figure (2-15)

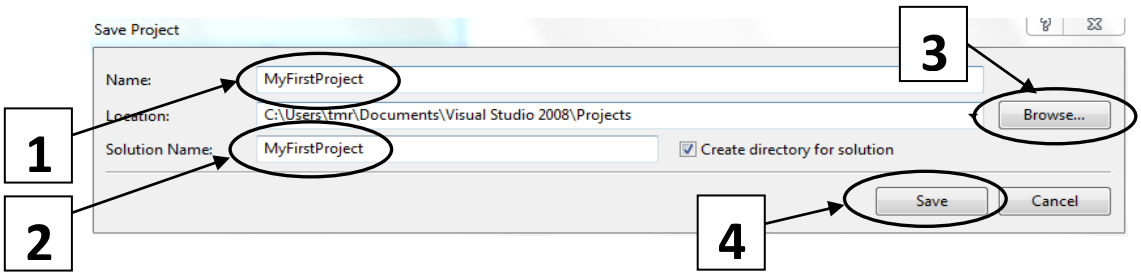


Figure (2-15) Save Project window

The numbers shown in figure (2-15) indicate:

(1) **The Project name given to the project when created (as you learned earlier in this chapter), you can change it if you wish.**

(2) **The Solution name that will include; the Project you want to save, (you can change Solution name if you wish).**

(3) **The Browse button; to explore storage devices available on your computer.**

(4) **With your teacher's help, choose the storage device you prefer for saving your Project, then press the (Save) button for the (Project) to be saved.**

Exercise (2–4)

Add a new (Project) to the (Solution)

Choose (File) menu then select (Add) then (New Project) as shown in figure (2-16).

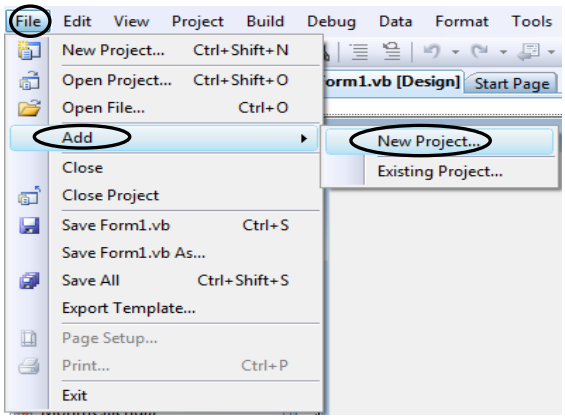


Figure (2-16) Adding a new Project

With your teacher's help, give this name (MySecondProject) to your new Project. The (Solution Explorer) window becomes as shown in figure (2-17).

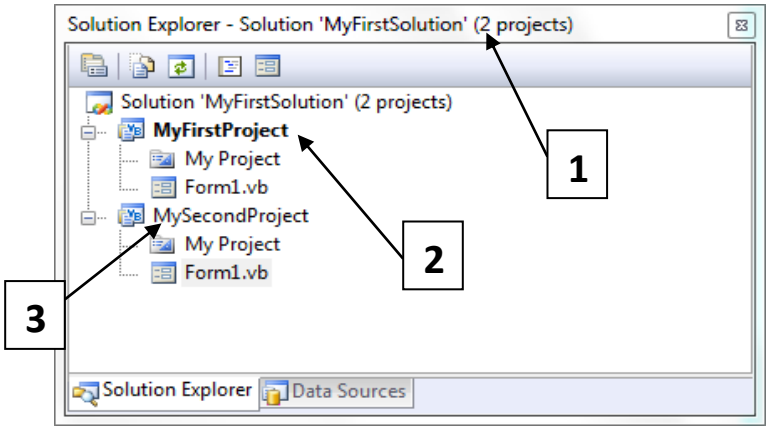


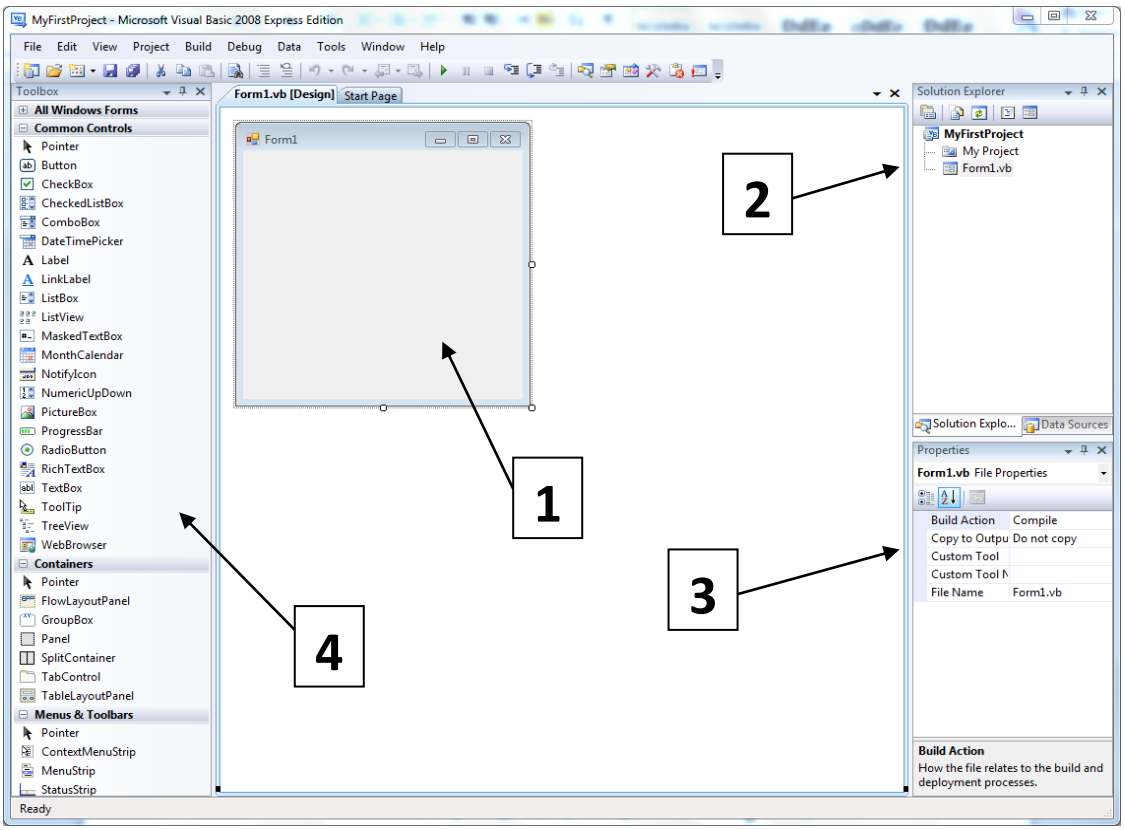
Figure (2-17) The new Project is displayed in the (Solution Explorer)

The numbers shown in figure (2-17) indicate:

(1) The Solution name
(2) The first project name
(3) The second Project name

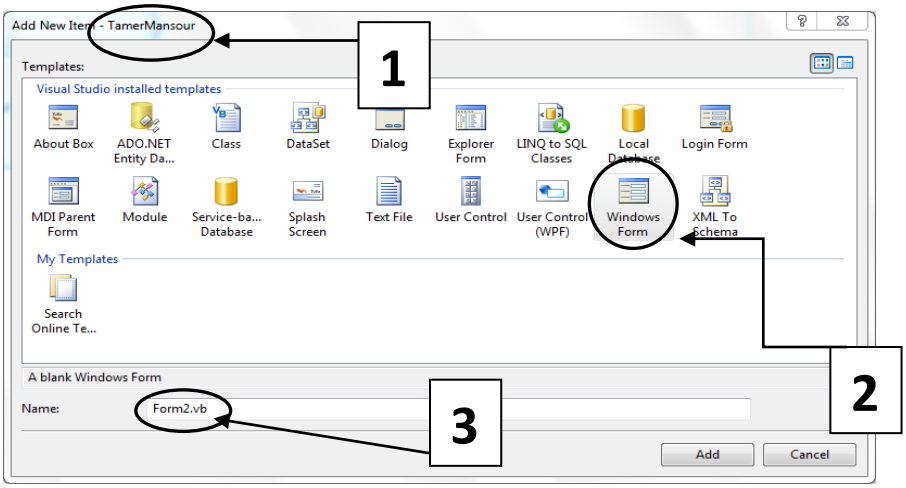
Questions

First: Write in the table below what the following numbers indicate



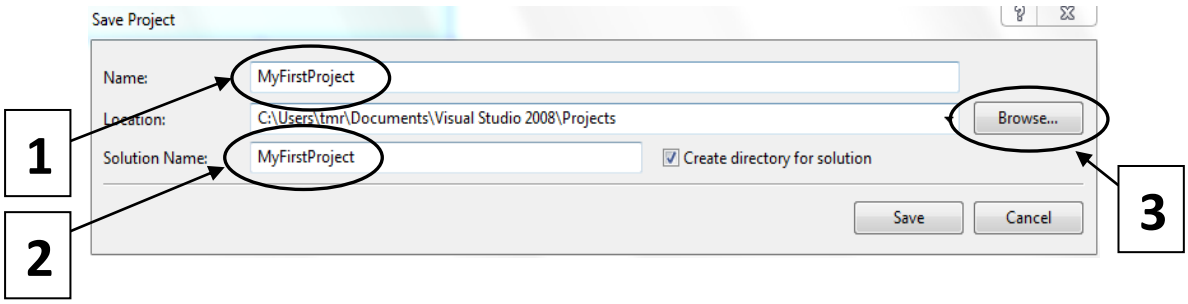
Number	Indicates
1	.....
2	.....
3	.....
4	.....

Second: In the following (**Add New Item**) window, what the following numbers indicate?



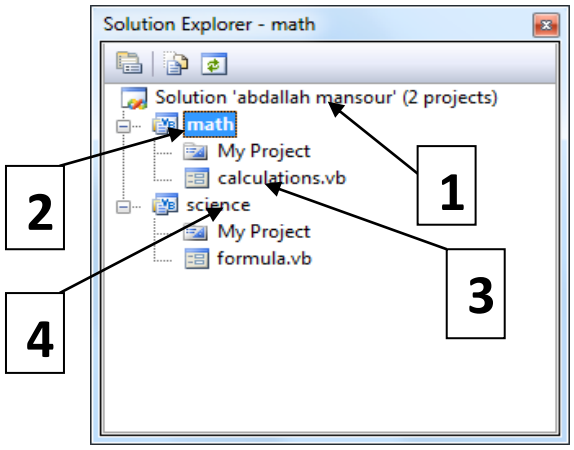
Number	Indicates
1	.....
2	.....
3	.....

Third: Refer to the following figure then, explain what the following numbers indicate?



What is the benefit of this window? ……
Number 1 indicates: ……
Number 2 indicates: ……
Number 3 indicates: ……

Fourth: In the displayed (**Solution Explorer**) window what the following numbers indicate?



Number	Indicates
1	……
2	……
3	……
4	……

Fifth: Complete the sentences with the following:

Object – Class –Visual Studio –.NET framework – Programming language – Form

1- ..... is used in the development, and design of (Desktop application) – (Web application) – (Mobile application).

2- Through ....., Instructions, and commands are written according to certain rules; and then translated to machine language.

3- .....provides an environment , that enables developers to design and execute (.NET applications).

4- .....is defined as: The basic constructive element in Object Oriented Programming; it is created from a defined class.

5- The blueprint, from which the individual objects are created; is called .....