The City School

North Nazimabad Boys Campus

Reinforcement Objective Worksheet

**Developing Application (Chapter – 4)**

1. When creating a new project in Visual Basic Express, it creates an **empty form**.
2. We need to add some elements in our program to do some useful work, these elements are called **controls**.
3. A **button** is a clickable control that allow you to specify program instructions to be run.
4. A **Label** is just a place that allows you to display information, like a text field for the results.
5. To change the title of your program, write in the **Text Property** of you form.
6. Click the **Auto Hide** button to make the Toolbox permanently visible.
7. When you finish testing your program, close the form window by clicking the **‘x’** just like any other window.
8. A **variable** is a place in the computer’s memory where we can store data.
9. Text Box is another type of control that allows the **user** to input data while the program is running.
10. Dim is short for **dimension** and it’s a keyword used to declare a **new variable**.
11. String is a common variable type which is used for **text**.
12. By default, project’s files are saved in **Libraries > Documents > Visual Studio 2010 >Projects.**
13. Each time you test your program by clicking the **Start Debugging** button.
14. The + operator can also be used to **concatenate String variables**.
15. The = sign does not mean ‘equals’, here = is an **assignment** operator.
16. Variable whose value does not change in the program is called **constant** variable and its keyword is **Const**.
17. The **Public variable** is visible and accessible from within the code blocks.
18. The **Private variable** is accessible for the specific block.
19. Decision making is an **important** and **powerful** tool of every good programming language.
20. Decision means a program that will be able to respond to **user input** during its execution and give different result.
21. The Select Case statement is useful when we want to check a **single variable.**
22. If….Then…ElseIf statement is useful when we want to work with **multiple choices**.
23. Mouse Events are actions that a user can do **with the mouse** when a program is running.
24. Repetitions means user can repeat the task **more than once**.
25. In Repetitions we have two types: **For….To….Next** and **Do….While**.
26. For counter = 1 to 50 means its **initial value** is 1 and **end value** is 50.
27. Every time the variable is incremented, the commands between **For and Next** are executed.
28. For….Next Loops are used when number of repetitions are **known**.
29. Do….While Loops are used when number of repetitions are **unknown**.
30. vbTab is a predefined constant of Visual Basic representing a **tab space**.
31. Do…While loop structure **does not increment** any counter variable automatically.
32. For…Next loop structure **increment** any counter variable automatically.
33. The **ListBox** control displays a list from which the user can select items.
34. An array is a special kind of variable which can hold **more than one value** at a time.
35. The way we store multiple values in arrays is by using an **index**.
36. The line of code that start with ‘(apostrophe) are called **Comments**.
37. We can redefine the size of an array by using the **ReDim statement**.
38. The difference between function and subroutines is; function **return a value** and subroutines **does not return the value**.
39. Comments help you understand what the code that you see, it is very helpful to **remember your code.**
40. ByVal is short for **By Value**.
41. ByVal means you are passing a **copy of a variable**, and you can make changes to copy but original variable will not change.
42. ByRef is short for **By Reference**.
43. ByRef means you are **not passing a copy of the original variable** but pointing to the original variable.
44. Modules keep your code files clean and **easy to read and maintain**.
45. All the controls like labels, textboxes and buttons make up the **user interface**.
46. To add menu at user interface, we use **Menu Strip**.
47. To add toolbox at user interface, we use **Tool Strip**.
48. A good User Interface provides a good **User Experience**.
49. A lot of labels and text boxes grouped together and positioning them using **Snap Lines**.
50. The Microsoft Office’s user interface design now uses a **tabbed ribbon** that replaces older traditional menu based interfaces.
51. A **responsive interface** is fast and does not waste the user’s time unnecessarily.
52. A responsive interface also provides some form of **feedback** to the user.
53. A good interface must be forgiving user errors by providing **undo and redo options**.
54. A **typography-based user interface** is to rely more on typography and less on graphics.
55. A user interface that does not follow the guidelines for a good user experience and cause just frustration is **bad user interface.**
56. Programming errors are broken down into three types: **Design-time, Runtime and Logic** **errors**.
57. Design time errors are also known as **Syntax errors**.
58. Design time errors are the most easy to find and correct, and these occur when you **mistype an instruction**.
59. At the bottom of our code window, there is the **Error list Panel**, which lists all error found in the current code tab.
60. Runtime errors are harder to find because they occur when the **program is running**.
61. Runtime errors usually **crash** your program.
62. Runtime errors are the errors that the programmer should have **predicted** but didn’t.
63. Logic errors are the **most difficult** to find.
64. Logic errors occur while your **program is running**.
65. Logic errors **don’t crash** your program.
66. In order to prevent code from crashing when an error occurs, we can use **Try….Catch statement.**
67. In debug mode, we can get an overview of all your variables and how they **change from line to line**, by using Local panel.
68. The Locals panel appears by selecting **Debug > Windows > Locals** from the menu.