

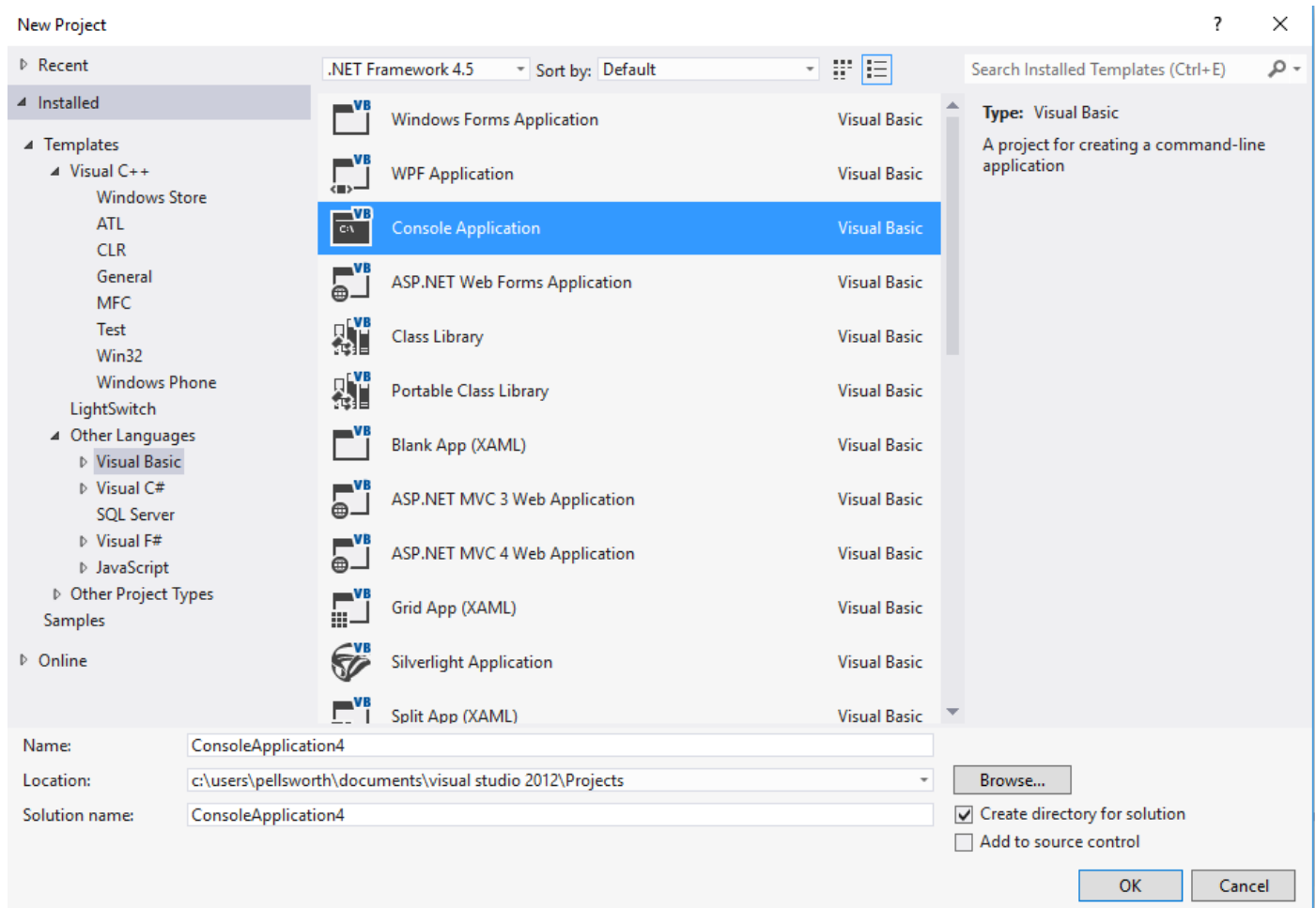
Objective: The student will be able to write a simple program to perform basic arithmetic operations.

Directions: Perform the following steps.

Step 1: Start Visual Studio.

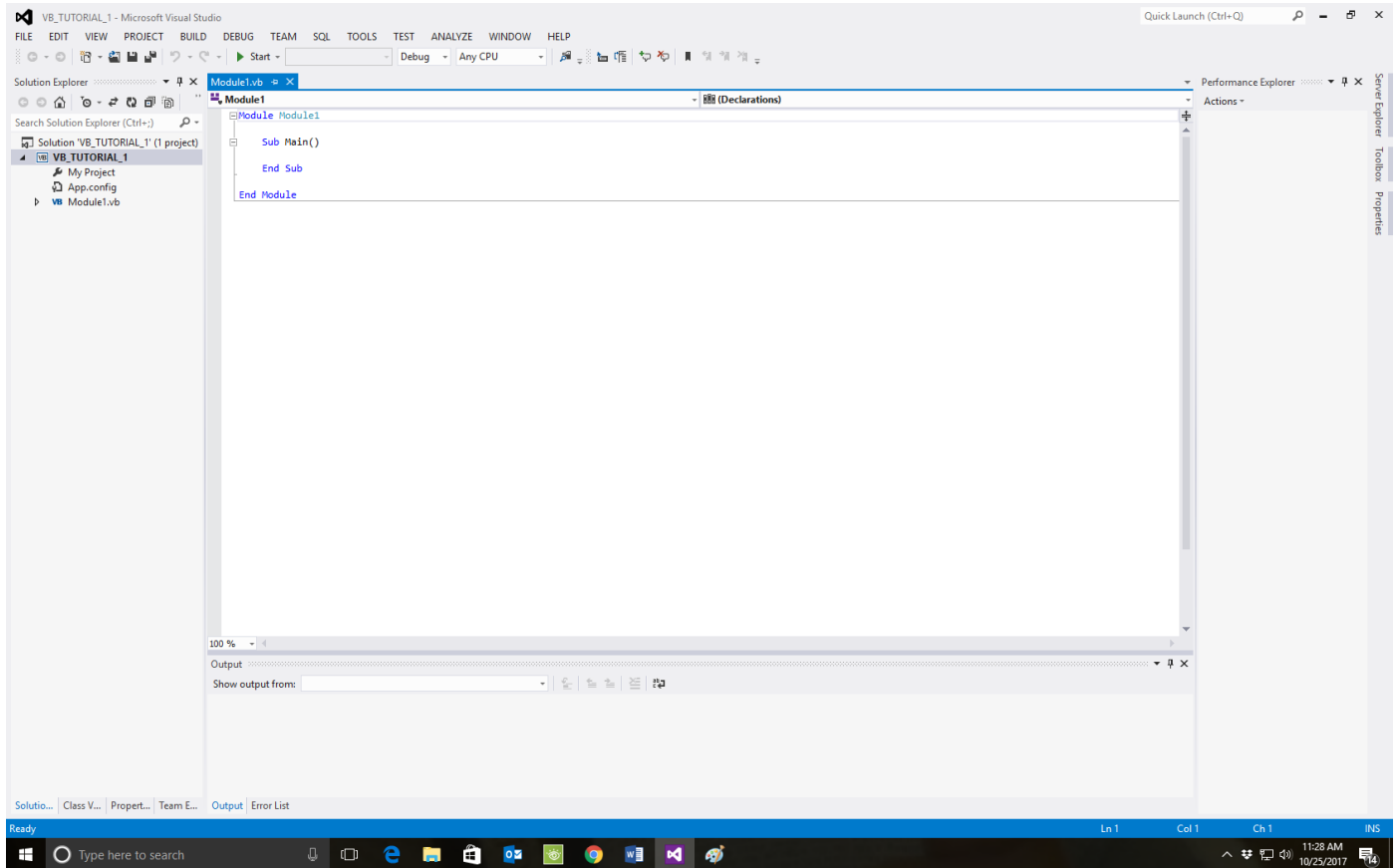
Step 2: Select Visual Basic.

Step 3: Select Empty Project.



Step 4: Give the project a meaningful name.

Step 5: Click on OK.



Step 6: Enter the following program:

```
Sub Main()  
  
    REM Simple Adding Program  
    REM Name:_____ Period:_____ Date:___/___/___  
  
    REM Declare Variables  
  
    Dim number1 As Integer = 0  
    Dim number2 As Integer = 0  
    Dim total As Integer = 0  
  
    REM Input Section  
  
    Console.WriteLine("Please enter the first number: ")  
    number1 = Console.ReadLine()  
  
    Console.WriteLine("Please enter the second number: ")  
    number2 = Console.ReadLine()  
  
    REM Processing Section  
  
    total = number1 + number2  
  
    REM Output Section  
  
    Console.WriteLine("The Total = " & total)  
  
End Sub
```

Step 7: REM stand for remark and this is a method for entering comments into the program. Comments are useful information for the programmer. The compiler ignores REM statements. (Another way of indicate a remark is using the single quotation mark, ' . It does exactly the same as the REM; however, I don't use it because it is so small it does not show up on the screen well).

Step 8: The Dim statement stands for Dimension and it is used to declare a variable of a specific type. The most commonly used data types are Integer, Decimal, Boolean, Char, and String. There are other data types that have been defined but they are used less often.

Integer is the data type used to hold counting numbers, both negative and positive.

Decimal is the data type used to hold numbers with a decimal point.

Boolean is the data type used to hold a true or false value.

Char is the data type to hold a single letter.

String is the data type to hold a group of letters.

The statement:

```
Dim number1 as Integer = 0
```

This statement declares the variable `number1` to be assigned to a memory location that holds data of type integer and assigns an initial value of 0 to it.

Visual Basic will by default assign a value of 0 to the variable. However, you will have a warning indicator, a green squiggly line under the variable name. This warning indicates the variable has not been assigned a value. Once the variable is used in the program the warning will be removed.

Step 9: The statement:

```
Console.WriteLine("Please enter the first number: ")
```

This statement writes the text string enclosed by the double quotes to the screen. In this case we are doing something a little different than the previous program. You will remember that in the previous program we assigned values to our variables when we declared them. In this program we will read the values in from the keyboard in real-time (or when the program is running).

Step 10: The statement:

```
number1 = Console.ReadLine()
```

This statement causes the program to stop and wait for the user to enter a number on the keyboard. In this case the Console Read Device is defined as the keyboard.

Step 11: The statement:

```
total = number1 + number2
```

This is an assignment statement. The variable `total` is assigned the value contained in the variable `number1` added to the value contained in the variable `number2`. The equal sign used in the context of this statement means assignment and is referred to as the assignment operator. Notice the `+` symbol is used to indicate addition. The basic arithmetic operators are:

+	add
-	subtract
*	multiply

/ divide
^ raise to a power

Step 12: The statement:

```
Console.WriteLine("The Total = " & total)
```

This is a statement to write to the screen. The phrase in double quotes is printed on the screen. This is referred to as a literal string. Anytime you have text between double quotes you have a string. The & symbol is used to concatenate multiple piece of data into a single write statement.

This concludes the tutorial for lesson 4.