**Part 70 - Making method parameters optional by using OptionalAttribute.**

In this video, we will discuss making **method parameters optional** by using **OptionalAttribute** that is present in System.Runtime.InteropServices namespace  
  
**This method allows us to add any number of integers.**

public static void AddNumbers(int firstNumber, int secondNumber,   
    int[] restOfTheNumbers)  
{  
    int result = firstNumber + secondNumber;  
    foreach (int i in restOfTheNumbers)  
    {  
        result += i;  
    }  
  
    Console.WriteLine("Total = " + result.ToString());  
}

If we want to add 5 integers - 10, 20, 30, 40 and 50. We call the method as shown below.  
**AddNumbers**(10, 20, new int[]{30, 40, 50});  
  
**At the moment all the 3 parameters are mandatory**. If I want to add just 2 numbers, then I can invoke the method as shown below. Notice that, I am passing an empty integer array as the argument for the 3rd parameter.  
AddNumbers(10, 20, new int[]{});  
  
**We can make the 3rd parameter optional** by using OptionalAttribute that is present in System.Runtime.InteropServices namespace. Make sure you have "using" declaration for System.Runtime.InteropServices namespace.  
public static void AddNumbers(int firstNumber, int secondNumber,  
    [Optional] int[] restOfTheNumbers)  
{  
    int result = firstNumber + secondNumber;  
  
    // loop thru restOfTheNumbers only if it is not null  
    // otherwise you will get a null reference exception  
    if (restOfTheNumbers != null)  
    {  
        foreach (int i in restOfTheNumbers)  
        {  
            result += i;  
        }  
    }  
  
    Console.WriteLine("Total = " + result.ToString());  
}  
  
So, if we want to add just 2 numbers, we can now use the function as shown below.  
AddNumbers(10, 20);