|  |
| --- |
| Day18 Morning Assignment  By  Anusha Bellala  16-2-2022 |

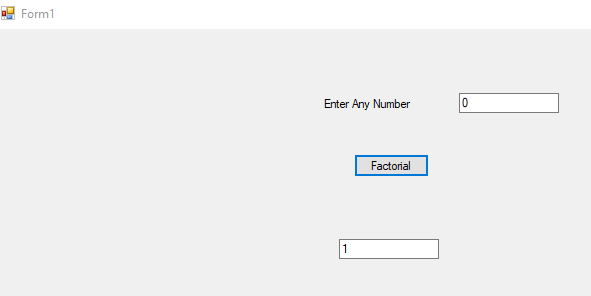
|  |
| --- |
| 1. What is the use of XML |
| * Xml is used for Universal data transfer mechanism to send data across different platforms. * User can define, his own tags. Hence it is called as user Defined Tags. * It is not a Platform dependent. It can be used in any platform with ease. * The redundancy in syntax of XML causes higher storage and transportation cost when the volume of data is large. |

|  |
| --- |
| 2. Write the points discussed about xml in the class |
| * XML stands for extensible Markup Language. * XML has user defined tags. * It is a Case Sensitive. * XML Has only one Root tag as an Entry. * It’s not a platform dependent. * XML syntax is verbose and redundant compared to other text-based data transmission formats such as JSON. |

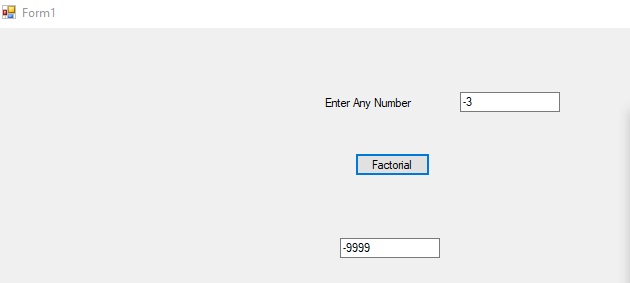
|  |
| --- |
| 3. Create a simple xml to illustrate:  a. Tag based xml with 10 products  b. Attribute based xml |
| a.)**Tag Based XML:** |
| **Attribute Base XML:** |

|  |
| --- |
| 4. Convert the above xml to JSON and display the JSON data |
| 5. Research and write the benefits of JSON over XML ( 2 or 3 points ) |
| * JSON Occupies less file size, comparatively to XML file. * JSON is faster because it is designed specifically for data interchange. * JSON encoding is terse, which requires less bytes for transit. * JSON parsers are less complex, which requires less processing time and memory overhead. * XML is slower, because it is designed for a lot more than just data interchange |

|  |
| --- |
| 6. For the below requirement, create a layered architecture project with separate class library for Business logic.    create console application  create windows(or desktop) application  Business Requirement:  FIND FACTORIAL OF A NUMBER:  0 = 1    positive number (upto 7) = factorial answer  > 7 = -999 (as answer)  < 0 = -9999 (as answer)  put the screen shots of the output and project (solution explorer) screen shot. |
| **AnushaLibrary:**  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace AnushaLibrary  {  public class Algebra  {  public static int Factorial(int n)  {  if (n == 0)  return 1;  else if (n < 0)  return -9999;  else if (n > 7)  return -999;  else  {  int fact = 1;  for (int i = 1; i <= n; i++)  {  fact \*= i;  }  return fact;  }  }  }  } |
| **Program.cs:**  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using AnushaLibrary;  namespace Day18Project1  {  internal class Program  {  static void Main(string[] args)  {  Console.Write("Enter Any Number : "); int n = int.Parse(Console.ReadLine());  Console.WriteLine("\n\n Factorial Result is : {0}", Algebra.Factorial(n));  Console.ReadLine();  }  }  } |
| **MyWindowsForms:**  using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  using AnushaLibrary;  namespace MyWindowsForms  {  public partial class Form1 : Form  {  public Form1()  {  InitializeComponent();  }  private void button1\_Click(object sender, EventArgs e)  {  int n = int.Parse(textBox1.Text);  int result = Algebra.Factorial(n); textBox2.Text = result.ToString();  Console.ReadLine();  }  }  } |
| Ouput: |







|  |
| --- |
| 7. For the above method, Implement TDD  and write 4 test cases and put the code in word document.  put the screen shot of all test cases failing.  make the test cases pass.  put the screen shot |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using AnushaLibrary;  namespace AnushaLibrary  {  public class Algebra  {  public static int Factorial(int n)  {  if (n == 0)  return 1;  else if (n < 0)  return -9999;  else if (n > 7)  return -999;  else  {  int fact = 1;  for (int i = 1; i <= n; i++)  {  fact \*= i;  }  return fact;  }  }  }  } |
| **AlgebraTests.cs:**  using Microsoft.VisualStudio.TestTools.UnitTesting;  using AnushaLibrary;  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace AnushaLibrary.Tests  {  [TestClass()]  public class AlgebraTests  {  [TestMethod()]  public void FactorialTest\_Zero\_Input()  {  // Arrange  int n = 0;  int expected = 1;  // Act  int actual = Algebra.Factorial(n);  // Assert  Assert.AreEqual(expected, actual);  }  [TestMethod()]  public void FactorialTest\_Negative\_Input()  {  // Arrange  int n = -55;  int expected = -9999;  // Act  int actual = Algebra.Factorial(n);  // Assert  Assert.AreEqual(expected, actual);  }  [TestMethod()]  public void FactorialTest\_Greater\_than\_seven\_Input()  {  // Arrange  int n = 10;  int expected = -999;  // Act  int actual = Algebra.Factorial(n);  // Assert  Assert.AreEqual(expected, actual);  }  [TestMethod()]  public void FactorialTest\_Input()  {  // Arrange  int n = 5;  int expected = 120;  // Act  int actual = Algebra.Factorial(n);  // Assert  Assert.AreEqual(expected, actual);  }  }  } |
| **Outputs:**  **All TestCases Failed:** |
| **All TestCases Passed:** |

|  |
| --- |
| 8. Add one more method to check if the number is palindrome or not in the above Algebra class and write test case for the same. |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using AnushaLibrary;  namespace AnushaLibrary  {  public class Algebra  {  public static bool IsPalindrome(int n)  {  int rev = 0, rem, m;  m = n;  while (m > 0)  {  rem = m % 10;  m = m / 10;  rev = rev \* 10 + rem;  }  if (n == rev)  return true;  else  return false;  }  }  } |
| **Ouputs:** |
| **All TestCases Passed:** |
|  |