# 5 – DAY – TASK (23-08-2024)

1. XML File (Console & Window Application) Program Code:

   Course.cs File Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace XmlTask
{
    internal class Course
    {
        public int Cid {  get; set; }
        public string Cname { get; set; }
        public int Cduration { get; set; }

        public Course(int Cid, string Cname, int Cduration)
        {
            this.Cid = Cid;
            this.Cname = Cname;
            this.Cduration = Cduration;
        }

    }
}
```

   Program.cs File code:

```
using System.Xml;

namespace XmlTask
{
    internal class Program
    {
        private static void Main(string[] args)
```

```
        {
            Course[] courses = new Course[3];
            courses[0] = new Course(11, "DotNetCore", 3);
            courses[1] = new Course(12, "Angular", 1);
            courses[2] = new Course(13, "Data Analytics", 2);
            using (XmlWriter writer = XmlWriter.Create("Courses.xml"))
            {
                writer.WriteStartDocument();
                writer.WriteStartElement("Courses");
                foreach (Course course in courses)
                {
                    writer.WriteStartElement("Course");
                    writer.WriteElementString("CourseID", course.Cid.ToString());
                    writer.WriteElementString("CourseName",
course.Cname.ToString());
                    writer.WriteElementString("DurationInMonths",
course.Cduration.ToString());
                    writer.WriteEndElement();
                }
                writer.WriteEndElement();
                writer.WriteEndDocument();
            }

        }
    }

}
```

Form1.cs File code:

```
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 System.Xml;
using System.Xml.Linq;
```

```
namespace WindowsForm
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            XmlReader xmlread =
XmlReader.Create(@"C:\Users\sanjeevkumar.v\source\repos\23_08_2024\XmlTask\XmlTas
k\bin\Debug\net8.0\Courses.xml");
            while (xmlread.Read())
            {
                switch (xmlread.NodeType)
                {
                    case XmlNodeType.Element:
                        listBox1.Items.Add("<" + xmlread.Name + ">");
                        break;
                    case XmlNodeType.Text:
                        listBox1.Items.Add(xmlread.Value);
                        break;
                    case XmlNodeType.EndElement:
                        listBox1.Items.Add("<" + xmlread.Name + ">");
                        break;
                }
            }



        }

        private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
        {

        }

        private void button2_Click(object sender, EventArgs e)
        {
            XDocument doc =
XDocument.Load(@"C:\Users\sanjeevkumar.v\source\repos\23_08_2024\XmlTask\XmlTask\
bin\Debug\net8.0\Courses.xml");
```

```
            //linq query
            var elets = from ele in doc.Descendants("Course")
                        where
Convert.ToInt32(ele.Element("DurationInMonths").Value) > 1
                        select ele;
            foreach (var el in elets)
            {
                listBox2.Items.Add(el.Name + ":" + el.Value);
            }
        }
    }
}
```

Form1.Designer.cs File code:

```
namespace WindowsForm
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
```

```
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.button1 = new System.Windows.Forms.Button();
            this.button2 = new System.Windows.Forms.Button();
            this.listBox1 = new System.Windows.Forms.ListBox();
            this.listBox2 = new System.Windows.Forms.ListBox();
            this.SuspendLayout();
            //
            // button1
            //
            this.button1.Location = new System.Drawing.Point(272, 73);
            this.button1.Name = "button1";
            this.button1.Size = new System.Drawing.Size(127, 41);
            this.button1.TabIndex = 0;
            this.button1.Text = "Display XML";
            this.button1.UseVisualStyleBackColor = true;
            this.button1.Click += new System.EventHandler(this.button1_Click);
            //
            // button2
            //
            this.button2.Location = new System.Drawing.Point(999, 73);
            this.button2.Name = "button2";
            this.button2.Size = new System.Drawing.Size(117, 41);
            this.button2.TabIndex = 1;
            this.button2.Text = "With LINQ";
            this.button2.UseVisualStyleBackColor = true;
            this.button2.Click += new System.EventHandler(this.button2_Click);
            //
            // listBox1
            //
            this.listBox1.ForeColor = System.Drawing.SystemColors.WindowText;
            this.listBox1.FormattingEnabled = true;
            this.listBox1.ItemHeight = 20;
            this.listBox1.Location = new System.Drawing.Point(215, 157);
            this.listBox1.Name = "listBox1";
            this.listBox1.Size = new System.Drawing.Size(248, 324);
            this.listBox1.TabIndex = 2;
            this.listBox1.SelectedIndexChanged += new
System.EventHandler(this.listBox1_SelectedIndexChanged);
            //
            // listBox2
```

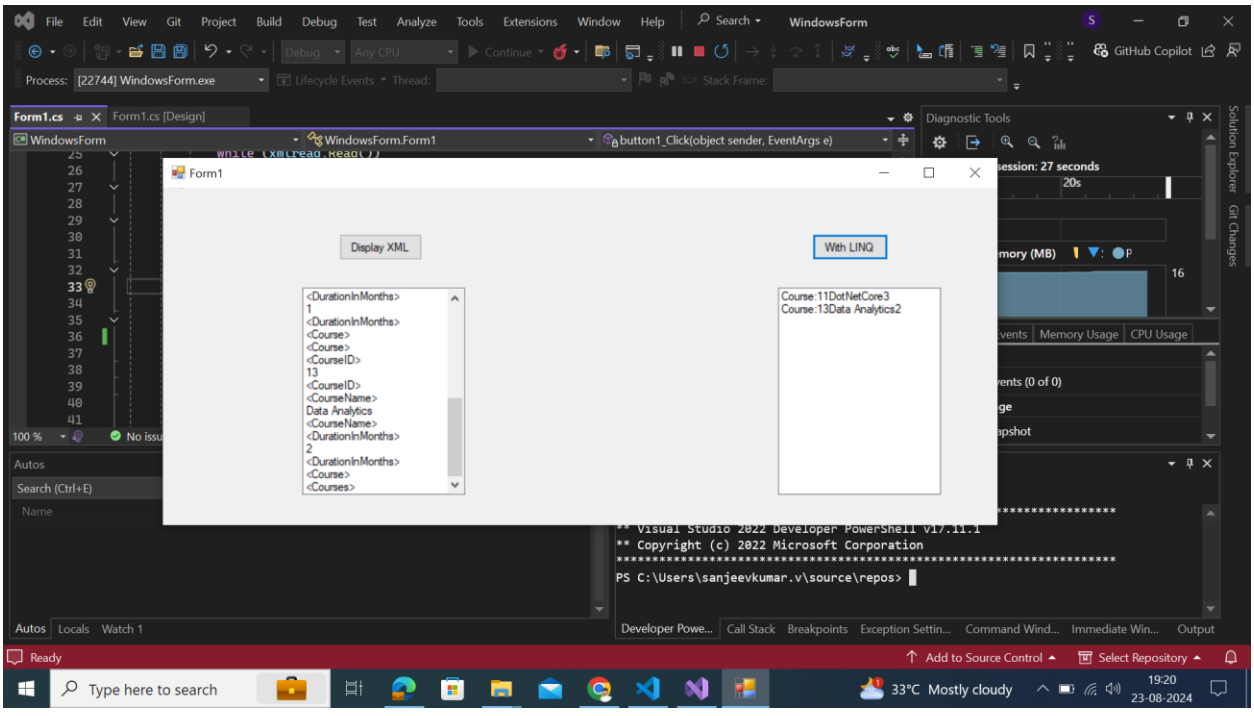
```
            //
            this.listBox2.FormattingEnabled = true;
            this.listBox2.ItemHeight = 20;
            this.listBox2.Location = new System.Drawing.Point(947, 157);
            this.listBox2.Name = "listBox2";
            this.listBox2.Size = new System.Drawing.Size(248, 324);
            this.listBox2.TabIndex = 3;
            //
            // Form1
            //
            this.AutoScaleDimensions = new System.Drawing.SizeF(9F, 20F);
            this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
            this.ClientSize = new System.Drawing.Size(1309, 531);
            this.Controls.Add(this.listBox2);
            this.Controls.Add(this.listBox1);
            this.Controls.Add(this.button2);
            this.Controls.Add(this.button1);
            this.Name = "Form1";
            this.Text = "Form1";
            this.ResumeLayout(false);

        }

        #endregion

        private System.Windows.Forms.Button button1;
        private System.Windows.Forms.Button button2;
        private System.Windows.Forms.ListBox listBox1;
        private System.Windows.Forms.ListBox listBox2;
    }
}
```

Program.cs File Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace WindowsForm
{
    internal static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Output:




2. UNIT Test Program Code:

MathOpe.cs File code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace UnitTest
{
    public class MathOpe
    {
        public int add(int a, int b)
        {
            return a + b;
        }
        public int Sub(int x, int y)
        {
            return x - y;
        }
        public int Pro(int x, int y)
        {
            return x * y;
        }
        public int Div(int x, int y)
        {
            return x / y;
        }
        public virtual bool CheckValues()
        {
            return false;
        }
    }
    public class Employee
    {
        string Name;
        int Age;
        public Employee(string nme, int age)
        {
            Name = nme;
            Age = age;
        }
```

```
        public string name
        {
            get
            {
                return Name;
            }
            set
            {
                Name = value;
            }
        }
        public int age
        {
            get
            {
                return Age;
            }
            set { Age = value; }
        }
    }
}
```

UnitTest1.cs File Code:

```
using Moq;

namespace UnitTest
{
    [TestFixture]
    public class Tests
    {
        public int i = 50, j = 20;
        public bool result;
        [SetUp]
        public void CheckNonNegative()
        {
            if (i > 0 && j > 0)
            {
                result = true;
            }
            else
            {
```

```
            result = false;
        }
    }
    [Test]
    public void TestAdd()
    {
        if (result)
        {

            MathOpe mth = new MathOpe();

            var res = mth.add(20, 20);
            Assert.AreEqual(40, res);
        }
        else
        {
            Assert.Fail();
        }

    }
    [Test]
    [TestCase(100, 25, 4)]
    [TestCase(50, 2, 25)]
    public void TestDiv(int a, int b, int expected)
    {

        MathOpe mth = new MathOpe();

        var res = mth.Div(a, b);

        Assert.AreEqual(expected, res);
    }
    [Test]
    public void TestSub()
    {
        var res = new MathOpe().Sub(10, 30);
        Assert.AreEqual(-20, res);
    }
    [Test]
    public void TestPro()
    {

        MathOpe mth = new MathOpe();
```

```
            var res = mth.Pro(i, j);
            Assert.AreEqual(1000, res);
        }
        [Test]
        public void MockTest()
        {
            Mock<MathOpe> mck = new Mock<MathOpe>();
            mck.Setup(x => x.CheckValues()).Returns(true);
            Assert.AreEqual(true, mck.Object.CheckValues());
        }
    }
}
```

Output: