

DOT NET

LAB MANUAL

DAVE PRERANA  
170473107003  
VVP, CE, SEM 6

## Table of Contents

Practical-1 .....	1
Practical-2 .....	11
Practical-3 .....	16
Practical-4 .....	21
Practical-5 .....	24
Practical-6 .....	27
Practical-7 .....	37
Practical-8 .....	40

## Practical-1

AIM-

-Introduction to c#:

Variables:

- Initialization
- Scope
- Constant

-Predefined Data Types

- Value Types
- Reference Types

-Flow Control

- Conditional Statements(if, switch)
- Loop(for, while, dowhile, foreach)
- Jump(goto, break, continue, return)

-Enumerations

-Passing Arguments

```
using System;
```

```
using System.Threading;
```

```
namespace P1
```

```
{
```

```
    class P1
```

```
    {
```

```
        static int j = 90;
```

```
        public enum TimeOfDay
```

```
        {
```

```
            Morning = 0,
```

```
            Afternoon = 1,
```

```
            Evening = 2
```

```
        }
```

```
        public static void Main(string[] args)
```

```
        {
```

```
Console.WriteLine("First Program");

int i;

i = 25;

Console.WriteLine("Scope of Variables.\n1:");

int j;

for (/*int*/ j = 0; j < 2; j++) //removing

comment from for loop will raise error

{

    //int j;

    //uncomment above line to error "A local

variable named 'j' cannot be declared in this

    Console.Write("{0} {1}\n", j, P1.j);

}

Console.WriteLine("2:");

for (int k = 0; k < 3; k++)

{

    Console.Write("{0} ", k);

}

Console.Write("\n");

for (int k = 3; k > 0; k--)

{

    Console.Write("{0} ", k);

}

Console.WriteLine("Constants");

const int valConst = 100; // This value cannot be

changed.

Console.WriteLine("{0} is constant value",
```

```
valConst);

    //const only allow constant variables into

the expression

    const int valConst2 = valConst + 9 /* + j*/;

    Console.WriteLine("Another Constant: {0}",

valConst2);

    Console.WriteLine("\nPredefined Data Types\n

\nValue Types and Reference Types");

    int vali = 2, valj = vali;

    Console.WriteLine("vali is: {0} and valj is:

{1}", vali, valj);

    valj = 90;

    Console.WriteLine("vali is: {0} and valj is:

{1}", vali, valj);

    Vector x, y;

    x = new Vector();

    x.value = 3;

    y = x;

    Console.WriteLine("x is: {0} and y is:{1}",

x.value, y.value);

    y.value = 234;

    Console.WriteLine("x is: {0} and y is:{1}",
```

```
x.value, y.value);

    y = null;
    Console.WriteLine("\nInteger Types");
    sbyte sb = 33;
    short s = 33;
    int _i = 33;
    long l = 33L;
    //Unsigned Integers
    byte b = 33;
    ushort us = 33;
    uint ui = 33U;
    ulong ul = 33UL;
    Console.WriteLine("{0} {1} {2} {3} {4} {5} {6}
{7}", sb, s, _i, l, b, us, ui, ul);
    //Floating point types
    float f = 11.22334455F;
    double d = 11.2233445566778899;
    Console.Write("\nFloat and Double:\n");
    Console.WriteLine("{0} and \n{1}", f, d);
    //Decimal Type
    decimal dec = 111.222333444555666777888999M;
    Console.WriteLine("Decimal:\n{0}", dec);
    //Boolean
    Console.WriteLine("\nBoolean:");
    bool valBoolean = true;
    Console.WriteLine("Status: " + valBoolean);
    //Character
    Console.WriteLine("\nCharacter:\nSingle Quote
```

```
\'");

    Console.WriteLine("Double Quote \");
    Console.WriteLine("Back Slash \\");
    char charA = 'A';
    Console.WriteLine(charA);
    charA = '\0';
    Console.WriteLine("Now null: " + charA);
    object o1 = "Hi, I am an Object";
    object o2 = 34;
    string strObj = o1 as string;
    Console.WriteLine(strObj);
    Console.WriteLine(o1.GetHashCode() + " " +
o1.GetType());
    Console.WriteLine(o2.GetHashCode() + " " +
o2.GetType());
    Console.WriteLine(o1.Equals(o2));

    //string
    string s1, s2;
    s1 = "String 1";
    s2 = s1;
    Console.WriteLine("S1 is: {0} and s2 is {1}", s1,
s2);
    s2 = "New String 1";
    Console.WriteLine("S1 is: {0} and s2 is {1}", s1,
s2);
    s1 = "c:\\NewFolder\\Hello\\P1.cs";
    Console.WriteLine(s1);
```

```
s1 = @"c:\NewFolder\Hello\P1.cs";
Console.WriteLine(s1);
s1 = @"We can also write
like this";
Console.WriteLine(s1);

//Flow Control
//The if Statement
bool isZero;
Console.WriteLine("\nFlow Control: (if)\ni is " +

i);

if (i == 0)
{
    isZero = true;
    Console.WriteLine("i is Zero");
}
else
{
    isZero = false;
    Console.WriteLine("i is Non - zero");
}

int integerA = 2;
Console.WriteLine("\nSwitch:");

switch (integerA)
{
    case 1:
        Console.WriteLine("integerA = 1");
        break;
```



```
        case 2:
            Console.WriteLine("integerA = 2");
            //goto case 3;
            break;
        case 3:
            Console.WriteLine("integerA = 3");
            break;
        default:
            Console.WriteLine("integerA is not 1, 2,
or 3");

            break;
    }

    WriteGreeting(TimeOfDay.Morning);
    Console.WriteLine("Argument is: {0}", args[1]);
}

static void WriteGreeting(TimeOfDay timeOfDay)
{
    switch (timeOfDay)
    {
        case TimeOfDay.Morning:
            Console.WriteLine("Good morning!");
            break;
        case TimeOfDay.Afternoon:
            Console.WriteLine("Good afternoon!");
            break;
        case TimeOfDay.Evening:
            Console.WriteLine("Good evening!");
            break;
        default:
    }
```

```
        Console.WriteLine("Hello!");
        break;
    }
}

public class Vector
{
    public int value;
}
}
```

**OUTPUT**

First Program

Scope of Variables.

1:

0 90

1 90

2:

0 1 2

3 2 1 Constants

100 is constant value

Another Constant: 109

Predefined Data Types

Value Types and Reference Types

vali is: 2 and valj is: 2

vali is: 2 and valj is: 90

x is: 3 and y is:3

x is: 234 and y is:234

Integer Types

33 33 33 33 33 33 33 33

Float and Double:

11.22334 and

11.2233445566779

Decimal:

111.2223334444555666777888999

Boolean:

Status: True

Character:

Single Quote '

Double Quote "

Back Slash \

A

Now null:

Hi, I am an Object

-1735802816 System.String

34 System.Int32

False

S1 is: String 1 and s2 is String 1

S1 is: String 1 and s2 is New String 1

c:\NewFolder\Hello\P1.cs

c:\NewFolder\Hello\P1.cs

We can also write like this

Flow Control: (if)

i is 25

i is Non - zero

Type in a string:

abhay

The string had at least 5 but less than 10

characters The string was abhay

Switch:

integerA = 2

Good morning!

## Practical-2

### AIM:GTU Programs

1. Write console based program in code behind language VB or C# to print following pattern.

```
@ @ @ @ @
 @ @ @ @
  @ @ @
   @ @
    @
     @
```

```
using System;
namespace Pattern
{
    class PatternExample
    {
        public static void Main()
        {
            int i,j=5;
            for (; j > 0; j--)
            {
                for (i = j; i > 0; i--)
                    Console.Write("@ ");
                Console.WriteLine();
            }
        }
    }
}
```

2. Write console based program in code behind language VB or C# to print following pattern.

```
1
1 2
1 2 3
1 2 3 4
```

```
using System;
namespace Pattern
{
    class patternExample
    {
        public static void Main()
        {
            int i, j;
            for (j = 1; j < 5; j++)
            {
                for (i = 1; i <= j; i++)
                    Console.Write(i + " ");
                Console.WriteLine();
            }
        }
    }
}
```

3. Write C# code to prompt a user to input his/her name and country name and then the output will be shown as an example below:  
Hello Ram from country India

```
using System;
public class userdata
{
    public static void Main()
    {
        string name, country;
        Console.Write("Enter Your Name: ");
        name = Console.ReadLine();
        Console.Write("Enter Your Country: ");
        country = Console.ReadLine();
        Console.WriteLine("Hello " + name + " from country " + country);
    }
}
```

**OUTPUT:**

E:\SEM-6 .NET\VS\p2\p2>Read.exe

Enter your name:

Prerana

Enter your Country:

India

Hello Prerana from Country India

4. What is inheritance? Create C# console application to define Car class and derive Maruti and Mahindra from it to demonstrate inheritance.  
using System;

```
public class Car
{
    protected string name;
    public Car(string name)
    {
        this.name = name;
    }
    public Car()
    {

```

```
    }
    public virtual string Name
    {
        get{return name;}
        set
        {
            if(value.Length>3)
                name = value;
            else
                name="Unknown";
        }
    }
}

public class Maruti : Car
{
    public Maruti(string name) : base(name)
    {
    }
    public override string Name
    {
        get{return name;}
        set
        {
            if(value.Length>3)
                name = value + " -Maruti";
            else
                name="Unknown";
        }
    }
    public bool haveAGS;
}

public class Mahindra : Car
{
    public Mahindra(string name) : base(name)
    {
    }
    public Mahindra(){}
    public override string Name
    {
        get{return name;}
        set
        {
            if(value.Length>3)
                name = value + " -Mahindra";
            else
                name="Unknown";
        }
    }
}

public class Program
{
    public static void Main()
    {
    }
}
```



```
        Maruti car1 = new Maruti("Swift");
        car1.haveAGS = true;
        car1.Name = "Swift";
        Console.WriteLine("Details Car 1: {0} and
{1}", car1.Name, car1.haveAGS==true?"Have AGS":"not Have AGS");
        Mahindra car2 = new Mahindra();
        car2.Name = "XUV500";
        Console.WriteLine("Car 2: {0}", car2.Name);
    }
}
```

**OUTPUT:**

E:\SEM-6 .NET\VS\p2\p2>Inheritance.exe

This is maruti class

This is Mahindra class...

### Practical-3

#### AIM : Metod & Constructor Overloading

1. Write a c# program to add two integers, two vectors and two metric using method overloading.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace p3
{
    public class Add
    {
        public void add()
        {
            int[,] m1 = new int[50, 50];
            int[,] m2 = new int[50, 50];
            int[,] m3 = new int[50, 50];
            Console.WriteLine("enter size of array:");
            int size = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("enter first array:");
            for (int i = 0; i < size; i++)
            {
                for (int j = 0; j < size; j++)
                {
                    m1[i, j] = Convert.ToInt32(Console.ReadLine());
                }
            }
            Console.WriteLine("enter second array:");
            for (int i = 0; i < size; i++)
            {
                for (int j = 0; j < size; j++)
                {
                    m2[i, j] = Convert.ToInt32(Console.ReadLine());
                }
            }

            for (int i = 0; i < size; i++)
            {
                for (int j = 0; j < size; j++)
                {
                    m3[i, j] = m1[i, j] + m2[i, j];
                }
            }

            Console.WriteLine("addition array:");

            for (int i = 0; i < size; i++)
            {
```

```

        Console.WriteLine("\n");
        for (int j = 0; j < size; j++)
        {
            Console.WriteLine("{0}\t", m3[i, j]);
        }
        Console.WriteLine("\n");
    }
}
public int add(int a, int b)
{
    return (a + b);
}
}
public class Vector
{
    public void add()
    {
        Console.WriteLine("enter first vector");
        int x = Convert.ToInt32(Console.ReadLine());
        int y = Convert.ToInt32(Console.ReadLine());
        int z = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("enter second vector");
        int x1 = Convert.ToInt32(Console.ReadLine());
        int y1 = Convert.ToInt32(Console.ReadLine());
        int z1 = Convert.ToInt32(Console.ReadLine());
        int x2 = x + x1;
        int y2 = y + y1;
        int z2 = z + z1;
        Console.WriteLine("<" + x2 + "," + y2 + "," + z2 + ">");
    }
}
class Program
{
    static void Main(string[] args)
    {
        Add a1 = new Add();
        Vector v1 = new Vector();

        v1.add();

        a1.add();

        int res=a1.add(1, 2);

        Console.WriteLine("method overloading for addtion{0}",res);

        Console.ReadLine();

    }
}
}

```

**OUTPUT:**

E:\SEM-6 .NET\VS\p2\p2>P3.1.exe

Enter Number 1:

1

Enter Number 2:

2

Addition of Number:3

Enter Vector 1:

1

2

Enter Vector 2:

3

1

Addition of vector: x=4, y=3

Addition of two metrics:

Addition: 6

Addition: 8

Addition: 10

Addition: 12

**2. Write a c# program that create student object. Overload constructor to create new instant with following details.**

- 1. Name**
- 2. Name, Enrollment**
- 3. Name, Enrollment, Branch**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Reflection;
namespace p3a1
{
    class Program
    {
        public int ID { get; set; }
        public string Name { get; set; }
        String name, branch;
        int enrol;
        public Program(String name)
        {
            this.name = name;
            Console.WriteLine("constructor 1:" + name);
        }
        public Program(String name, int enrol)
        {
            this.name = name;
            this.enrol = enrol;
            Console.WriteLine("constructor 2:" + name + " " + enrol);
        }
        public Program(String name, int enrol, String branch)
        {
            this.name = name;
            this.enrol = enrol;
            this.branch = branch;
            Console.WriteLine("constructor 3:" + name + " " + enrol + " " + branch);
        }
        static void Main(string[] args)
        {
            Program p1 = new Program("bob");
        }
    }
}
```

```
Program p2 = new Program("bob", 1);  
Program p3 = new Program("bob", 1, "computer");  
Console.ReadLine(); } }
```

**OUTPUT:**

**Constructor 1:bob**

**Constructor 2:bob 1**

**Constructor 3:bob 1 computer**

### Practical-4

AIM:find Methods, Properties and Constructors from class of running program

1. Create a c# program to find Methods, Properties and Constructors from class of running program.(Use Class from previous practical)

```
Using System;
using System.Reflection;

namespace ReflectionExample
{
    class MainClass
    {
        static void Main()
        {
            Type T = Type.GetType("ReflectionExample.Customer");
            MethodInfo[] methods = T.GetMethods();
            foreach (MethodInfo method in methods)
            {
                Console.WriteLine(method.ReturnType + " " + method.Name);
            }

            PropertyInfo[] properties = T.GetProperties();

            Console.WriteLine("\nProperties");
            foreach (PropertyInfo property in properties)
            {
                Console.WriteLine(property.PropertyType+" "+ property.Name);
            }

            Console.WriteLine("\nConstructors");
            ConstructorInfo[] constructors = T.GetConstructors();
            foreach (ConstructorInfo constructor in constructors)
            {
                Console.WriteLine(constructor.ToString());
            }
        }
    }
    class Customer
    {
        public int ID { get; set; }
        public string Name { get; set; }
        public Customer(int ID, string Name)
        {
            this.ID = ID;
            this.Name = Name;
        }
        public Customer()
        {
            this.ID = -1;
        }
    }
}
```

```
        this.Name = string.Empty;
    }
    public void printID()
    {
        Console.WriteLine("ID is: {0}", this.ID);
    }
    public void printName()
    {
        Console.WriteLine("Name is: {0}", this.Name);
    }
}
}
```

**OUTPUT:**

E:\SEM-6 .NET\VS\p2\p2>Reflection.exe

System.Int32 get\_ID

System.Void set\_ID

System.String get\_Name

System.Void set\_Name

System.Void printID

System.Void printName

System.String ToString

System.Boolean Equals

System.Int32 GetHashCode

System.Type GetType

Properties



System.Int32 ID

System.String Name

### Constructors

Void .ctor(Int32, System.String)

Void .ctor()

## Practical-5

### AIM : HANDLING FILE

1. Write a C# program to copy data from one file to another using StreamReader and StreamWriter class.

```
using System;
using System.IO;

class CopyFile
{
    public static void Main()
    {
        string file1 = @"abc.txt";
        string file2 = @"xyz.txt";

        using (StreamReader reader = new StreamReader(file1))
        using (StreamWriter writer = new StreamWriter(file2))

            writer.Write(reader.ReadToEnd());
    }
}
```

#### OUTPUT:

abc.txt:

Hello World.....

hii

how

are you

???

xyz.txt:

Hello World.....

hii

how

**2. Write a C# Program to Read Lines from a File until the End of File is Reached.**

```
using System;
using System.IO;

public class CopyFileContain
{
    public void copyFile(string file1, string file2)
    {
        using(StreamReader reader = new StreamReader(file1))
        using (StreamWriter writer = new StreamWriter(file2))
        {
            string line = null;
            while ((line = reader.ReadLine()) != null)
                writer.WriteLine(line);
        }
    }
}
```

**3. Write a C# Program to List Files in a Directory.**

```
using System;
using System.IO;

class MyClass
{
    public static void Main()
    {
        string[] Directories = Directory.GetDirectories(@"F:\assignment\DOT
NET\Topics");

        foreach (string dir in Directories)
            Console.WriteLine(dir);

        string[] files = Directory.GetFiles(@"F:\assignment\DOT NET\Topics");

        foreach (string file in files)
            Console.WriteLine(file);
    }
}
```

```
}
```

```
}
```

**OUTPUT:**

E:\SEM-6 .NET\VS\p2

E:\SEM-6 .NET\VS\Assignment.docx

E:\SEM-6 .NET\VS\C# word.txt

E:\SEM-6 .NET\VS\Doc1.docx

E:\SEM-6 .NET\VS\P1-master.zip

E:\SEM-6 .NET\VS\p1.cs

E:\SEM-6 .NET\VS\p1.exe

E:\SEM-6 .NET\VS\VS.docx

E:\SEM-6 .NET\VS\~\$VS.docx

### Practical-6

AIM: Create Windows Form Application for Student Registration and store student Details in DataBase.

```

Namespace StudentReistration
{
    partial class Form1
    {
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        private void InitializeComponent()
        {
            this.groupBox1 = new System.Windows.Forms.GroupBox();
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.txtFname = new System.Windows.Forms.TextBox();
            this.tctMname = new System.Windows.Forms.TextBox();
            this.txtLname = new System.Windows.Forms.TextBox();
            this.radioButton1 = new System.Windows.Forms.RadioButton();
            this.rdoFemale = new System.Windows.Forms.RadioButton();
            this.imgStudent = new System.Windows.Forms.PictureBox();
            this.btnImage = new System.Windows.Forms.Button();
            this.label3 = new System.Windows.Forms.Label();
            this.txtMobile = new System.Windows.Forms.TextBox();
            this.label4 = new System.Windows.Forms.Label();
            this.txtEmail = new System.Windows.Forms.TextBox();
            this.label5 = new System.Windows.Forms.Label();
            this.dateDob = new System.Windows.Forms.DateTimePicker();
            this.openFileDialog1 = new System.Windows.Forms.OpenFileDialog();
            this.btnSave = new System.Windows.Forms.Button();
            this.btnCancel = new System.Windows.Forms.Button();
            this.groupBox1.SuspendLayout();

            ((System.ComponentModel.ISupportInitialize)(this.imgStudent)).BeginInit();
            this.SuspendLayout();
            //
            // groupBox1
            //
            this.groupBox1.Controls.Add(this.dateDob);
            this.groupBox1.Controls.Add(this.btnImage);
            this.groupBox1.Controls.Add(this.rdoFemale);
        }
    }
}

```

```

        this.groupBox1.Controls.Add(this.imgStudent);
        this.groupBox1.Controls.Add(this.radioButton1);
        this.groupBox1.Controls.Add(this.txtLname);
        this.groupBox1.Controls.Add(this.tctMname);
        this.groupBox1.Controls.Add(this.txtEmail);
        this.groupBox1.Controls.Add(this.txtMobile);
        this.groupBox1.Controls.Add(this.txtFname);
        this.groupBox1.Controls.Add(this.label4);
        this.groupBox1.Controls.Add(this.label5);
        this.groupBox1.Controls.Add(this.label3);
        this.groupBox1.Controls.Add(this.label2);
        this.groupBox1.Controls.Add(this.label1);
        this.groupBox1.ForeColor =
System.Drawing.SystemColors.ButtonHighlight;
        this.groupBox1.Location = new System.Drawing.Point(24, 23);
        this.groupBox1.Name = "groupBox1";
        this.groupBox1.Size = new System.Drawing.Size(600, 174);
        this.groupBox1.TabIndex = 0;
        this.groupBox1.TabStop = false;
        this.groupBox1.Text = "Personal Details";
        //
        // label1
        //
        this.label1.AutoSize = true;
        this.label1.ForeColor =
System.Drawing.SystemColors.ButtonHighlight;
        this.label1.Location = new System.Drawing.Point(25, 25);
        this.label1.Name = "label1";
        this.label1.Size = new System.Drawing.Size(38, 13);
        this.label1.TabIndex = 0;
        this.label1.Text = "Name:";
        this.label1.TextAlign =
System.Drawing.ContentAlignment.MiddleRight;
        //
        // label2
        //
        this.label2.AutoSize = true;
        this.label2.ForeColor =
System.Drawing.SystemColors.ButtonHighlight;
        this.label2.Location = new System.Drawing.Point(18, 61);
        this.label2.Name = "label2";
        this.label2.Size = new System.Drawing.Size(45, 13);
        this.label2.TabIndex = 1;
        this.label2.Text = "Gender:";
        this.label2.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
        //
        // txtFname
        //
        this.txtFname.Location = new System.Drawing.Point(70, 21);
        this.txtFname.Name = "txtFname";
        this.txtFname.Size = new System.Drawing.Size(119, 20);
        this.txtFname.TabIndex = 2;
        //
        // tctMname
        //

```

```
this.tctMname.Location = new System.Drawing.Point(195, 21);
this.tctMname.Name = "tctMname";
this.tctMname.Size = new System.Drawing.Size(119, 20);
this.tctMname.TabIndex = 2;
//
// txtLname
//
this.txtLname.Location = new System.Drawing.Point(320, 21);
this.txtLname.Name = "txtLname";
this.txtLname.Size = new System.Drawing.Size(119, 20);
this.txtLname.TabIndex = 2;
//
// radioButton1
//
this.radioButton1.AutoSize = true;
this.radioButton1.ForeColor =
System.Drawing.SystemColors.ButtonHighlight;
this.radioButton1.Location = new System.Drawing.Point(81, 59);
this.radioButton1.Name = "radioButton1";
this.radioButton1.Size = new System.Drawing.Size(48, 17);
this.radioButton1.TabIndex = 3;
this.radioButton1.TabStop = true;
this.radioButton1.Text = "Male";
this.radioButton1.UseVisualStyleBackColor = true;
//
// rdoFemale
//
this.rdoFemale.AutoSize = true;
this.rdoFemale.ForeColor =
System.Drawing.SystemColors.ButtonHighlight;
this.rdoFemale.Location = new System.Drawing.Point(134, 59);
this.rdoFemale.Name = "rdoFemale";
this.rdoFemale.Size = new System.Drawing.Size(59, 17);
this.rdoFemale.TabIndex = 3;
this.rdoFemale.TabStop = true;
this.rdoFemale.Text = "Female";

        this.rdoFemale.UseVisualStyleBackColor = true;

//

// imgStudent

//

this.imgStudent.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;

this.imgStudent.Location = new System.Drawing.Point(483, 19);

this.imgStudent.Name = "imgStudent";

this.imgStudent.Size = new System.Drawing.Size(95, 113);

this.imgStudent.TabIndex = 1;
```

```
this.imgStudent.TabStop = false;

//

// btnImage

//

this.btnImage.ForeColor = System.Drawing.SystemColors.ActiveCaptionText;
this.btnImage.Location = new System.Drawing.Point(483, 138);
this.btnImage.Name = "btnImage";
this.btnImage.Size = new System.Drawing.Size(95, 23);
this.btnImage.TabIndex = 2;
this.btnImage.Text = "Upload Photo";
this.btnImage.UseVisualStyleBackColor = true;
this.btnImage.Click += new System.EventHandler(this.btnImage_Click);

//

// label3

//

this.label3.AutoSize = true;
this.label3.ForeColor = System.Drawing.SystemColors.ButtonHighlight;
this.label3.Location = new System.Drawing.Point(22, 99);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(41, 13);
this.label3.TabIndex = 1;
this.label3.Text = "Mobile:";
this.label3.TextAlign = System.Drawing.ContentAlignment.MiddleRight;

//

// txtMobile

//

this.txtMobile.Location = new System.Drawing.Point(70, 95);
this.txtMobile.Name = "txtMobile";
this.txtMobile.Size = new System.Drawing.Size(119, 20);
this.txtMobile.TabIndex = 2;
```



```
//  
// label4  
//  
this.label4.AutoSize = true;  
this.label4.ForeColor = System.Drawing.SystemColors.ButtonHighlight;  
this.label4.Location = new System.Drawing.Point(210, 98);  
this.label4.Name = "label4";  
this.label4.Size = new System.Drawing.Size(35, 13);  
this.label4.TabIndex = 1;  
this.label4.Text = "Email:";  
this.label4.TextAlign = System.Drawing.ContentAlignment.MiddleRight;  
//  
// txtEmail  
//  
this.txtEmail.Location = new System.Drawing.Point(255, 94);  
this.txtEmail.Name = "txtEmail";  
this.txtEmail.Size = new System.Drawing.Size(184, 20);  
this.txtEmail.TabIndex = 2;  
//  
// label5  
//  
this.label5.AutoSize = true;  
this.label5.ForeColor = System.Drawing.SystemColors.ButtonHighlight;  
this.label5.Location = new System.Drawing.Point(22, 138);  
this.label5.Name = "label5";  
this.label5.Size = new System.Drawing.Size(31, 13);  
this.label5.TabIndex = 1;  
this.label5.Text = "DoB:";  
this.label5.TextAlign = System.Drawing.ContentAlignment.MiddleRight;  
//
```

```
// dateDob
//
this.dateDob.Location = new System.Drawing.Point(70, 138);
this.dateDob.Name = "dateDob";
this.dateDob.Size = new System.Drawing.Size(200, 20);
this.dateDob.TabIndex = 4;
//
// openFileDialog1
//
this.openFileDialog1.FileName = "openFileDialog1";
//
// btnSave
//
this.btnSave.Location = new System.Drawing.Point(433, 406);
this.btnSave.Name = "btnSave";
this.btnSave.Size = new System.Drawing.Size(75, 23);
this.btnSave.TabIndex = 1;
this.btnSave.Text = "Save";
this.btnSave.UseVisualStyleBackColor = true;
this.btnSave.Click += new System.EventHandler(this.btnSave_Click);
//
// btnCancel
//
    this.btnCancel.Location = new System.Drawing.Point(527, 406);
    this.btnCancel.Name = "btnCancel";
    this.btnCancel.Size = new System.Drawing.Size(75, 23);
    this.btnCancel.TabIndex = 1;
    this.btnCancel.Text = "Cancel";
    this.btnCancel.UseVisualStyleBackColor = true;
    this.btnCancel.Click += new
System.EventHandler(this.btnCancel_Click);
//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
```

```
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.BackColor = System.Drawing.SystemColors.Desktop;
        this.ClientSize = new System.Drawing.Size(637, 495);
        this.Controls.Add(this.btnCancel);
        this.Controls.Add(this.btnSave);
        this.Controls.Add(this.groupBox1);
        this.Name = "Form1";
        this.Text = "Student Registration";
        this.groupBox1.ResumeLayout(false);
        this.groupBox1.PerformLayout();

        ((System.ComponentModel.ISupportInitialize)(this.imgStudent)).EndInit();
        this.ResumeLayout(false);

    }
}
```

#endregion

```
private System.Windows.Forms.GroupBox groupBox1;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.RadioButton rdoFemale;
private System.Windows.Forms.RadioButton radioButton1;
private System.Windows.Forms.TextBox txtLname;
private System.Windows.Forms.TextBox tctMname;
private System.Windows.Forms.TextBox txtFname;
private System.Windows.Forms.PictureBox imgStudent;
private System.Windows.Forms.Button btnImage;
private System.Windows.Forms.TextBox txtMobile;
private System.Windows.Forms.Label label3;
private System.Windows.Forms.TextBox txtEmail;
private System.Windows.Forms.Label label4;
private System.Windows.Forms.DateTimePicker dateDob;
private System.Windows.Forms.Label label5;
private System.Windows.Forms.OpenFileDialog openFileDialog1;
private System.Windows.Forms.Button btnSave;
private System.Windows.Forms.Button btnCancel;
    }
}
```

```
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.Data.SqlClient;
using System.IO;

namespace StudentReistration
{
    public partial class Form1 : Form
    {
        string imgPath;
        public Form1()
        {
            InitializeComponent();
        }

        private void radioButton2_CheckedChanged(object sender, EventArgs e)
        {

        }

        private void btnImage_Click(object sender, EventArgs e)
        {
            openFileDialog1.Filter = "Jpg|*.jpg";
            if (openFileDialog1.ShowDialog() == DialogResult.OK)
            {
                imgPath = @"C:\Users\CRP\Desktop\Images\" +
openFileDialog1.SafeFileName;
                imgStudent.Image =
Image.FromFile(openFileDialog1.FileName);
                //MessageBox.Show(imgPath);
            }
        }

        private void btnCancel_Click(object sender, EventArgs e)
        {
            Environment.Exit(0);
        }

        private void btnSave_Click(object sender, EventArgs e)
        {
            string source = @"Data Source=crp-pc\mydatabase;Initial
Catalog=temp1;Integrated Security=True";
        }
    }
}
```

```

        string select = "select count(*) from tblStudent";
        SqlConnection conn = new SqlConnection(source);
        SqlCommand cmd = new SqlCommand(select, conn);
        conn.Open();
        int i = Convert.ToInt16(cmd.ExecuteScalar());
        int pkStudent = i + 1;

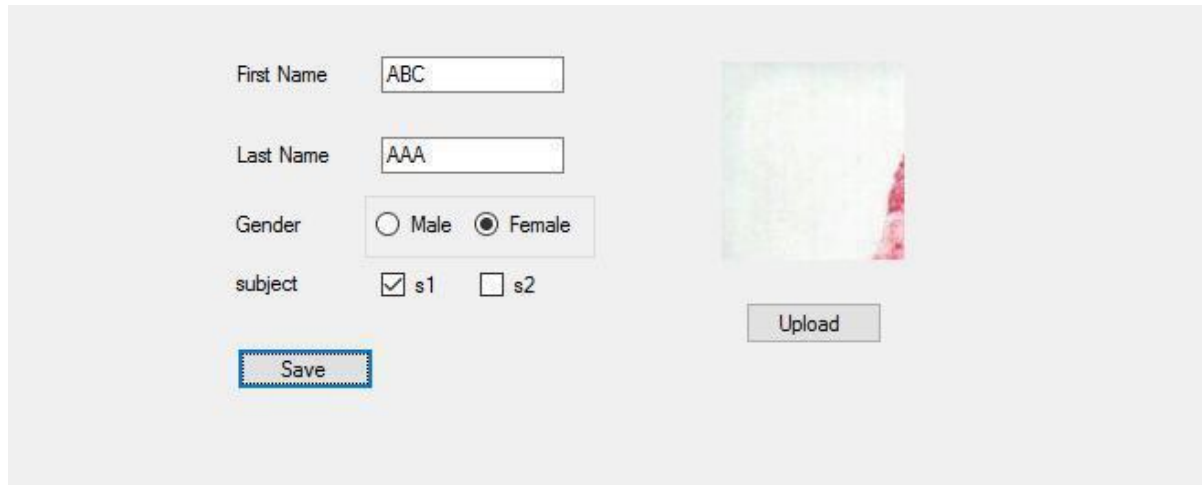
        string insert = "insert into tblStudent (pkStudent, fName,dob, imgStudent)
        values ( "+pkStudent+", '"+txtFname.Text+"', '"+dateDob.Value.Date +"' ,'" +
        (imgPath==null?"":imgPath) +"' )";
        cmd = new SqlCommand(insert,conn);

        i = cmd.ExecuteNonQuery();
        if(imgPath!=null)
            imgStudent.Image.Save(imgPath);
        MessageBox.Show("You are Done!!!");
        InitializeComponent();
    }
}

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

namespace StudentReistration
{
    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 :**

The screenshot displays a web form on a light gray background. On the left side, there are four labeled input fields: 'First Name' with the text 'ABC', 'Last Name' with the text 'AAA', 'Gender' with radio buttons for 'Male' (unselected) and 'Female' (selected), and 'subject' with checkboxes for 's1' (checked) and 's2' (unchecked). Below these fields is a 'Save' button with a blue border. To the right of the form fields is a square image placeholder showing a blurry, abstract pattern of green, white, and red. Below the image is an 'Upload' button.

## Practical-7

Aim : ASP.NET Validation Control

RequiredFieldValidator, CompareValidator, RegularExpressionValidator, CustomValidator, RangeValidator, ValidationSummary .

[illegible]

```

                MaximumValue="30"
                Display="Dynamic"
                Type="Integer"

        ErrorMessage="RangeValidator"></asp:RangeValidator>
                <br />
                <br />
        <asp:Label ID="Label4" runat="server" Text="Password"></asp:Label>
                <asp:TextBox ID="TextBox2"
runat="server"></asp:TextBox>
                </td>
        </tr>
        <tr>
                <td>
                        <asp:Label ID="Label3" runat="server" Text="confirm
password"></asp:Label>
                                <asp:TextBox ID="TextBox1"
runat="server"></asp:TextBox>
                                <asp:CompareValidator ID="CompareValidator1"
runat="server"
                                        ControlToCompare="TextBox2"
ControlToValidate="TextBox1"
                                        ErrorMessage="not comprae"></asp:CompareValidator>
                                </td>
                </tr>
        <tr>
                <td>
                        &nbsp;   Email
                                <asp:TextBox ID="TextBox3"
runat="server"></asp:TextBox>
                                <asp:RegularExpressionValidator
ID="RegularExpressionValidator1" runat="server"
                                        ControlToValidate="TextBox3" ErrorMessage="please
enter right email address"
                                        ValidationExpression="\w+([-+.' ]\w+)*@\w+([-
.] \w+)*\.\w+([-.] \w+)*"></asp:RegularExpressionValidator>
                                &nbsp;  </td>
                </tr>
        </table>
</div>
<p>
                <asp:Button ID="btnSave" runat="server" Text="Save" />
</p>
</form>
</body>
</html>

```



**OUTPUT :**

Name	<input type="text"/>	RequiredFieldValidator
Email	<input type="text" value="abcde"/>	RegularExpressionValidator
Password	<input type="password" value="..."/>	
Confirm Password	<input type="password" value="..."/>	CompareValidator
Sem	<input type="text" value="9"/>	RangeValidator

- RequiredFieldValidator
- RegularExpressionValidator
- CompareValidator
- RangeValidator

## Practical-8

### AIM :Introduction to Mater Page

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace WebApplication1
{
    public partial class Site2 : System.Web.UI.MasterPage
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        public Label LblHeader
        {
            get
            {
                return Label1;
            }
        }
    }
}

//Site1.master.designer.cs
namespace WebApplication1 {
    public partial class Site2 {
        /// <summary>
        /// head control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind
file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.ContentPlaceholder head;
        /// <summary>
        /// form1 control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind
file.
        /// </remarks>
        protected global::System.Web.UI.HtmlControls.HtmlForm form1;
        /// <summary>

```

```

        /// Label1 control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind
file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.Label Label1;

        /// <summary>
        /// ContentPlaceHolder1 control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind
file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.ContentPlaceHolder
ContentPlaceHolder1;
    }
}

// WebForm1.aspx
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace WebApplication1
{
    public partial class WebForm4 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            ((Site2)Master).LblHeader.Text = TextBox1.Text;
        }
    }
}

// WebForm4.aspx.designer
namespace WebApplication1 {

    public partial class WebForm4 {

```

```
    /// <summary>
    /// TextBox1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind
file.
    /// </remarks>
    protected global::System.Web.UI.WebControls.TextBox TextBox1;

    /// <summary>
    /// Button1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind
file.
    /// </remarks>
    protected global::System.Web.UI.WebControls.Button Button1;
    }
}
```

**OUTPUT :**

ABC

search	<input type="text"/>	ABC	Set Header
--------	----------------------	-----	------------

Footer

Header

search		<b>pkstudent</b>	<b>fname</b>	<b>lname</b>	<b>gender</b>	<b>subject</b>	<b>imgStudent</b>
A		22	ABC	AAA	f	s1	IMG-20170326-WA0009.jpg

Footer