Tribhuvan University
Faculty of Humanities and Social Sciences

Lab report on:
DotNet Technologies Lab :

Submitted to:
Mr. Jeeewan Rai ,
Department of Computer Application,
Himalaya College of Engineering,
Chyasal,Lalitpur

Submitted by:
Sujal Gurung
Roll no: 34
BCA III/V
March 14, 2024

S.N.	Title	Signature
1)	WAP to check even or odd number taking input from user.	
2)	WAP to input two numbers and perform an operation (+,-,*,x,/) on them and display the results.	
3)	WAP to store elements for 3x3 of 2D array and display the elements.	
4)	WAP to add any two 2X2 two dimensional arrays.	
5)	WAP to display the multiplication table of a given integer.	
6)	WAP to calculate the factorial of a given number.	
7)	WAP where class of one namespace access class of another namespace.	
8)	WAP to implement the use of Default, Parameterized, Static and Copy Constructors.	
9)	WAP for operator overloading of logical operator.	
10)	WAP for user defined exception to show whether candidate is eligible to caste vote	
11)	WAP to select Employees living in Kathmandu and salary is greater than 10000 using LINQ.	
12)	WAP to select Students living in Kathmandu and college is Deerwalk using LAMBDA Expression.	
13)	WAP to create book entry form in one ASP.NET page and display filled data in another page.	
14)	Provided that a mysql database, WAP for following tasks:	

Lab Work

1) Write a program to check even or odd number taking input from user.

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        Console.WriteLine("Enter a number");
        int n = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Number is even? " + (n % 2 == 0));
    }
}
```

2) Write a C# Sharp program that takes two numbers as input and perform an operation (+,-,*,x,/) on them and displays the result of that operation

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        int a, b;
        Console.WriteLine("Enter 2 numbers ");
        a = Convert.ToInt32(Console.ReadLine());
        b = Convert.ToInt32(Console.ReadLine());

        Console.WriteLine($"{a} + {b} = {a + b}");
        Console.WriteLine($"{a} - {b} = {a - b}");
        Console.WriteLine($"{a} * {b} = {a * b}");
        Console.WriteLine($"{a} / {b} = {a / b}");
    }
}
```

3) Write a program to store elements for 3x3 of 2D array and display the elements.

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        int[,] arr = {
            { 1, 2, 3 },
            { 4, 5, 6 },
            { 7, 8, 9 },
        };

        foreach (var i in arr) {
            Console.Write(i + " ");
        }
    }
}
```

4) Write a program to add any two 2X2 two dimensional arrays.

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        int[,] a = {
            {1, 2 },
            {3, 4 },
        };
        int[,] b = {
            {1, 2 },
            {3, 4 },
        };

        for (int i = 0; i < 2; i++) {
            for (int j = 0; j < 2; j++) {
                Console.Write(a[i, j] + b[i, j] + " ");
            }
            Console.WriteLine();
        }
    }
}
```

5) Write a program in C# Sharp to display the multiplication table of a given integer.

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        Console.Write("Enter a number: ");
        int a = Convert.ToInt32(Console.ReadLine());

        for (int i = 1; i <= 10; i++) {
            Console.WriteLine($"{a} * {i} = {a * i}");
        }
    }
}
```

6) Write a C# Sharp program to calculate the factorial of a given number.

```
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        Console.WriteLine("Enter a number ");
        int n = Convert.ToInt32(Console.ReadLine());
        int fact = 1;
        for (int i = 2; i <= n; i++) {
            fact *= i;
        }
        Console.WriteLine($"Factorial of {n} is {fact}");
    }
}
```

7) Write a program of using different namespaces where class of one namespace access class of another namespace program.

```
namespace Name1 {
    public class C1 {
        public C1() {
            Console.WriteLine("Class C1 instantiated");
        }
    }
}

namespace Name2 {
    using Name1;
    public class C2 {
        public static void main() {
            C1 c = new C1();
        }
    }
}
```

8) Write a program to implement the use of Default, Parameterized, Static and Copy Constructors.

```
namespace ConsoleApp1;
public class Class1 {
    public string name;

    public Class1() { } // default constructor

    public Class1(string name) { // parameterized constructor
        Console.WriteLine($"Passed name: {name}");
        this.name = name;
    }

    public Class1(Class1 c1) { // copy constructor
        this.name = c1.name;
        Console.WriteLine($"Copied name {name} to new object");
    }
}

public class Program {
    public static void Main() {
        Class1 c11 = new Class1();
        Class1 c12 = new Class1("Sujal");
        Class1 c13 = new Class1(c12);
    }
}
```

9) Write a program for operator overloading of logical operator.

```
namespace ConsoleApp1;
public class Complex {
    public bool a;
    public bool b;
    public Complex() { }
```

```
    public Complex(bool a, bool b) {
        this.a = a;
        this.b = b;
    }

    public static bool operator &(Complex c1, Complex c2) {
        return c1.a && c1.b && c2.a && c2.b;
    }

    public static bool operator | (Complex c1, Complex c2) {
        return c1.a || c1.b || c2.a || c2.b;
    }
}

public class Program {
    public static void Main() {
        Complex c1 = new Complex(true, true);
        Complex c2 = new Complex(true, false);
        Console.WriteLine($"AND result: { c1 & c2 }");
        Console.WriteLine($"OR result:  { c1 | c2 }");
    }
}
```

10) Write a program for creation of user defined exception to show whether candidate is eligible to caste vote

```
namespace ConsoleApp1;
public class IneligibleVoteException : Exception {
    public IneligibleVoteException() { }
    public IneligibleVoteException(string message): base(message) {}
}

public class Citizen {
    public int age {
        get { return age; }
        set {
            try {
                if (value < 18) throw new IneligibleVoteException("Too young to
                ↪  vote");
                else this.age = value;
            } catch(IneligibleVoteException e) {
                Console.WriteLine("ERROR: " + e.Message);
            }
        }
    }
}

public class Program {
    public static void Main() {
        Citizen c1 = new Citizen();
        c1.age = 17; // throws exception
    }
}
```

11) Write a program to select Employees who are lived in Kathmandu and salary is greater than 10000 using LINQ.

```
namespace ConsoleApp1;
public class Employee {
    public string Name;
    public string Address;
    public int Salary;

    public Employee(string name, string address, int salary) {
        this.Name = name;
        this.Address = address;
        this.Salary = salary;
    }
}

internal class Program {
    public static void Main() {
        List<Employee> employees = new List<Employee>() {
            new Employee("Sujal", "Kathmandu", 9000),
            new Employee("Hari", "Kathmandu", 1500),
            new Employee("Samip", "Pokhara", 20000),
        };
        var filtered = from e in employees
            where e.Address.Contains("Kathmandu") && e.Salary > 10000
            select e;

        Console.WriteLine("employees living in Kathmandu AND having salary
        ↪  greater than 10000: ");
        foreach(Employee e in filtered) {
            Console.WriteLine(e.Name);
        }
    }
}
```

12) Write a program to select Students who are lived in Kathmandu and college is Deerwalk College using LAMBDA Expression.

```
namespace ConsoleApp1;
public class Student {
    public string Name;
    public string Address;
    public string College;

    public Student(string name, string address, string college) {
        this.Name = name;
        this.Address = address;
        this.College = college;
    }
}

internal class Program {
    public static void Main() {
        List<Student> students = new List<Student>() {
            new Student("Sujal", "Kathmandu", "Himalaya"),
            new Student("Hari",  "Kathmandu", "Deerwalk"),
```

```
            new Student("Samip", "Kathmandu", "Deerwalk"),
            new Student("Arpan", "Pokhara",   "Deerwalk"),
          };
        var filtered = students.Where(s => s.Address.Contains("Kathmandu") &&
        ↪  s.College.Contains("Deerwalk") );

        Console.WriteLine("Students living in Kathmandu & studying in Deerwalk
        ↪  are: ");
        foreach(var s in filtered) {
            Console.WriteLine(s.Name);
        }
    }
}
```

13) Write a program to create book entry form in one ASP.NET page and display the filled data with another ASP.NET page.

Form.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master"
↪  AutoEventWireup="true" CodeBehind="Form.aspx.cs"
↪  Inherits="WebApplication1.Form" %>
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
    <h1>Enter book details: </h1>
    <label>ISBN:
        <asp:TextBox ID="ISBN" runat="server" />
    </label>
    <label>Title:
        <asp:TextBox ID="BookTitle" runat="server" />
    </label>
    <label>Author:
        <asp:TextBox ID="Author" runat="server" />
    </label>
    <label>Price:
        <asp:TextBox ID="Price" runat="server" />
    </label>
    <asp:Button ID="Submit" runat="server" Text="Submit" OnClick="Submit_OnClick"
    ↪  />
</asp:Content>
```

Form.aspx.cs

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

        protected void Submit_OnClick(object sender, EventArgs e) {
            Session["BookTitle"] = BookTitle.Text;
            Session["Author"] = Author.Text;
            Session["Price"] = Price.Text;
            Session["ISBN"] = ISBN.Text;
            Response.Redirect("Table.aspx");
        }
    }
}
```

Table.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master"
↪  AutoEventWireup="true" CodeBehind="Table.aspx.cs"
↪  Inherits="WebApplication1.Table" %>
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
    <asp:Table runat="server" border="1">
        <asp:TableRow >
            <asp:TableHeaderCell runat="server" >ISBN</asp:TableHeaderCell>
            <asp:TableHeaderCell runat="server" >Title</asp:TableHeaderCell>
            <asp:TableHeaderCell runat="server" >Author</asp:TableHeaderCell>
            <asp:TableHeaderCell runat="server" >Price</asp:TableHeaderCell>
        </asp:TableRow>
        <asp:TableRow >
            <asp:TableCell runat="server" ID="TableISBN"></asp:TableCell>
            <asp:TableCell runat="server" ID="TableTitle"></asp:TableCell>
            <asp:TableCell runat="server" ID="TableAuthor"></asp:TableCell>
            <asp:TableCell runat="server" ID="TablePrice"></asp:TableCell>
        </asp:TableRow>
    </asp:Table>

    <asp:Label runat="server" ID="TestLabel" ></asp:Label>
</asp:Content>
```

Table.aspx.cs

```
namespace WebApplication1 {
    public partial class Table : System.Web.UI.Page {
        protected void Page_Load(object sender, EventArgs e) {
            TableISBN.Text = Session["ISBN"] as string;
            TableAuthor.Text = Session["Author"] as string;
            TableTitle.Text = Session["BookTitle"] as string;
            TablePrice.Text = Session["Price"] as string;
        }
    }
}
```

14) Provided that a mysql database named "dotnet" with table "Product" with following columns (ProductId as int, ProductName as varchar(20), and UnitPrice as float). Write a program for following tasks:-

1. Connect to the database.
2. Display the product records that have UnitPrice is greater than RS.5000
3. Update the product records such as UnitPrice=10000 from product table whose ProductId=11
4. Delete the product records from product table whose ProductId=22
5. Insert any five product records

```
using MySql.Data.MySqlClient;
namespace ConsoleApp1;
internal partial class Program {
    public static void Main() {
        try {
            // a)
            string connectionString = "server=localhost; user=root;
            ↪  database=dotnet";
            MySqlConnection conn = new MySqlConnection(connectionString);
            conn.Open();

            // b)
            string sql = "select * from product where unitPrice > 5000";
            MySqlCommand cmd = new MySqlCommand(sql, conn);
            MySqlDataReader rdr = cmd.ExecuteReader();

            Console.WriteLine("productId | productName  | unitPrice");
            while(rdr.Read()) {
                Console.WriteLine($"{rdr[0]} \t  | {rdr[1]}\t | {rdr[2]}");
            }
            rdr.Close();

            // c)
            sql = "update product set unitPrice = 10000 where productId = 11";
            cmd = new MySqlCommand(sql, conn);
            cmd.ExecuteNonQuery();

            // d)
            sql = "delete from product where productId = 22";
            cmd = new MySqlCommand(sql, conn);
            cmd.ExecuteNonQuery();

            // e)
            // productId is omitted because it is set to autoincrement
            sql = "insert into product(productName, unitPrice) values ('Monitor',
            ↪  30000), ('Power Bank', 7000), ('Headphones', 4000)" ;
            cmd = new MySqlCommand(sql, conn);
            cmd.ExecuteNonQuery();


            sql = "select * from product";
            cmd = new MySqlCommand(sql, conn);
            rdr = cmd.ExecuteReader();

            Console.WriteLine("New values are: ");
            Console.WriteLine("productId | productName  | unitPrice");
            while(rdr.Read()) {
                Console.WriteLine($"{rdr[0]} \t  | {rdr[1]}\t | {rdr[2]}");
            }
            rdr.Close();
            conn.Close();
        }
        catch (Exception ex) {
            Console.WriteLine(ex.Message);
        }
    }
}
```

Output

- Initial values in database

productId	productName	unitPrice
1	laptop	50000
2	phone	20000
3	mouse	1500
11	Keyboard	4000
22	Tablet	25000

- b) (products with unitPrice > 5000)

productId	productName	unitPrice
1	laptop	50000
2	phone	20000
22	Tablet	25000

- Final values in database

productId	productName	unitPrice
1	laptop	50000
2	phone	20000
3	mouse	1500
11	Keyboard	10000
23	Monitor	30000
24	Power Bank	7000
25	Headphones	4000