1. **Write a program to become familiar with C# compiler options and interacts with user by creating the response file.**

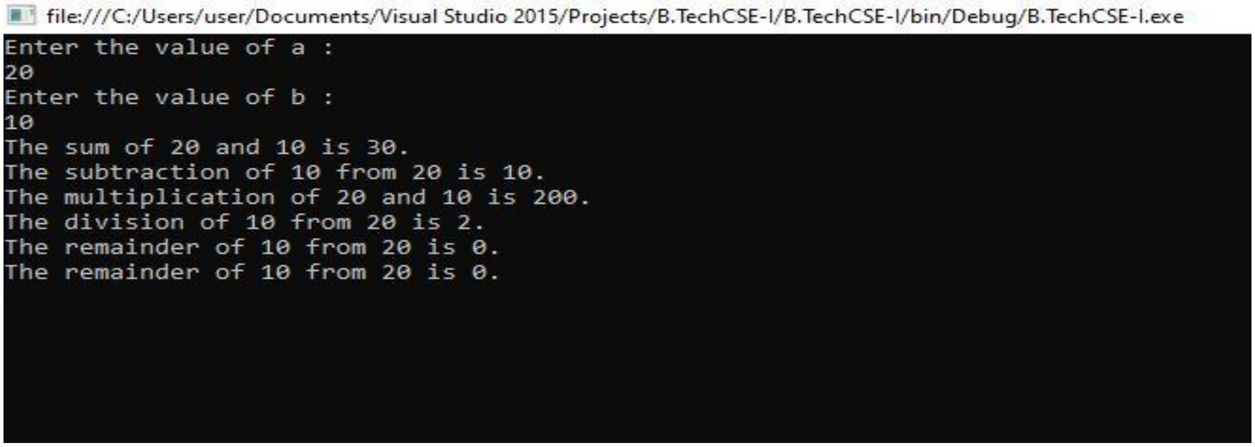
i. The first step is to create dll file .

```
using System;
namespace B.TechCSE_I
{
   class Program2
   {
      public int getAddition(int a, int b)
      {
         return a + b;
      }
      public int getSubtraction(int a, int b)
      {
         return a - b;
      }
      public int getMultiplication(int a, int b)
      {
         return a * b;
      }
      public int getDivision(int a, int b)
      {
         return a / b;
      }
      public int getRemainder(int a, int b)
      {
         return a % b;
      }
      public int getrem(int a, int b)
      {
         return a % b;
      }
   }
}
```

ii. The second step is to call the class after adding the reference of dll file.

```
using System;
namespace B.TechCSE_I
{
    class ArithmeticOP
    {
        static void Main(string[] args)
        {
            B.TechCSE_I.Program2 obj = new B.TechCSE_I.Program2();
            int a, b;
           Console.WriteLine("Enter the value of a :");
            a = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter the value of b :");
            b = Convert.ToInt32(Console.ReadLine());
 Console.WriteLine("The sum of " + a + " and " + b + " is " + obj.getAddition(a, b) );
 Console.WriteLine("The subtraction of " + b + " from " + a + " is " +
obj.getSubtraction(a, b) );
Console.WriteLine("The multiplication of " + a + " and " + b + " is " +
obj.getMultiplication(a, b) );
Console.WriteLine("The division of " + b + " from " + a + " is " + obj.getDivision(a, b));
Console.WriteLine("The remainder of " + b + " from " + a + " is " + obj.getRemainder(a,
b));
Console.WriteLine("The remainder of " + b + " from " + a + " is " + obj.getrem(a, b));
Console.Read();
        }
    }
}
```

Output :

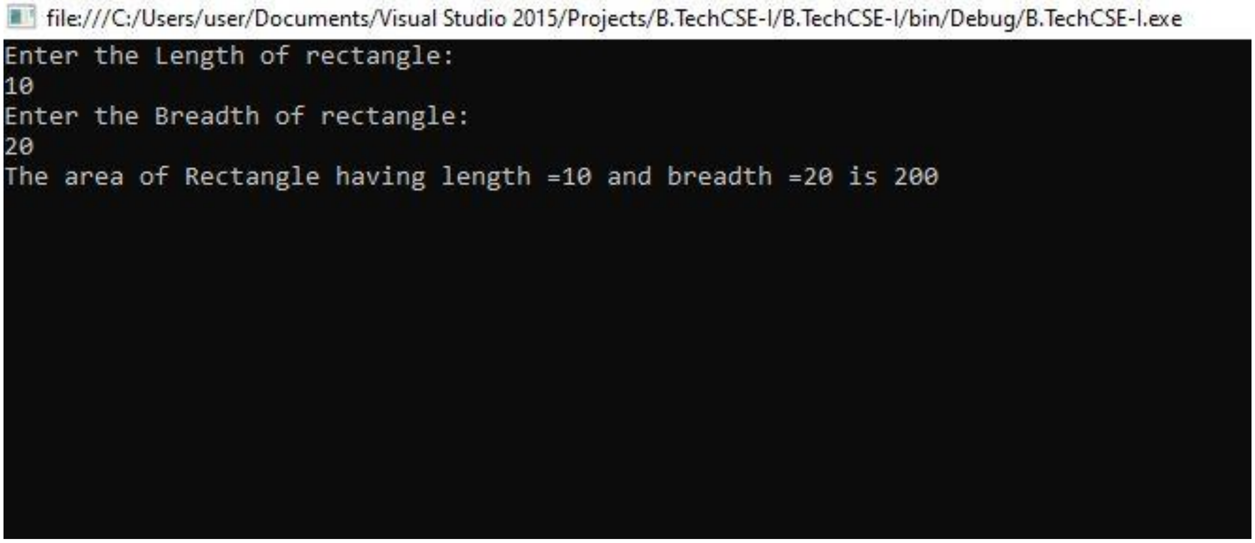

file:///C:/Users/user/Documents/Visual Studio 2015/Projects/B.TechCSE-I/B.TechCSE-I/bin/Debug/B.TechCSE-I.exe

```
Enter the value of a :
20
Enter the value of b :
10
The sum of 20 and 10 is 30.
The subtraction of 10 from 20 is 10.
The multiplication of 20 and 10 is 200.
The division of 10 from 20 is 2.
The remainder of 10 from 20 is 0.
The remainder of 10 from 20 is 0.
```

2. **Write a function to show the use of basic Input/output functions.**

```
using System;
namespace B.TechCSE_I
{
    class Program_2
    {
        static void Main(string[] args)
        {
            int Length, breadth;
            Console.WriteLine("Enter the Length of rectangle:");
            Length = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter the Breadth of rectangle:");
            breadth = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("The area of Rectangle having length =" + Length + " and
breadth =" + breadth + " is "+ Length*breadth);
            Console.Read();
        }
        }
}
```

Output :



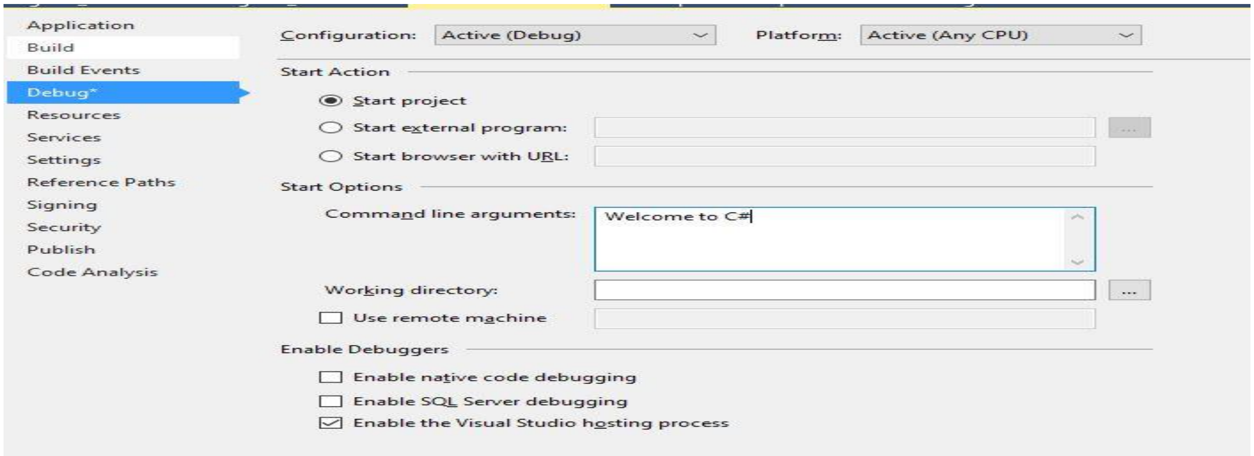
file:///C:/Users/user/Documents/Visual Studio 2015/Projects/B.TechCSE-I/B.TechCSE-I/bin/Debug/B.TechCSE-I.exe

```
Enter the Length of rectangle:
10
Enter the Breadth of rectangle:
20
The area of Rectangle having length =10 and breadth =20 is 200
```

3. Write a program in C# to demonstrate Command Line arguments processing.

```
using System;
namespace B.TechCSE_I
{
   class Program_3
   {
      static void Main(string[] args)
      {
         Console.WriteLine("Argument length: " + args.Length);
         Console.WriteLine("Supplied Arguments are:");
         foreach (Object obj in args)
         {
            Console.WriteLine(obj);
         }
         Console.Read();
      }
   }
}
```

Output :



Configuration: Active (Debug) Platform: Active (Any CPU)

Start Action

● Start project

○ Start external program:

○ Start browser with URL:

Start Options

Command line arguments: Welcome to C#

Working directory:

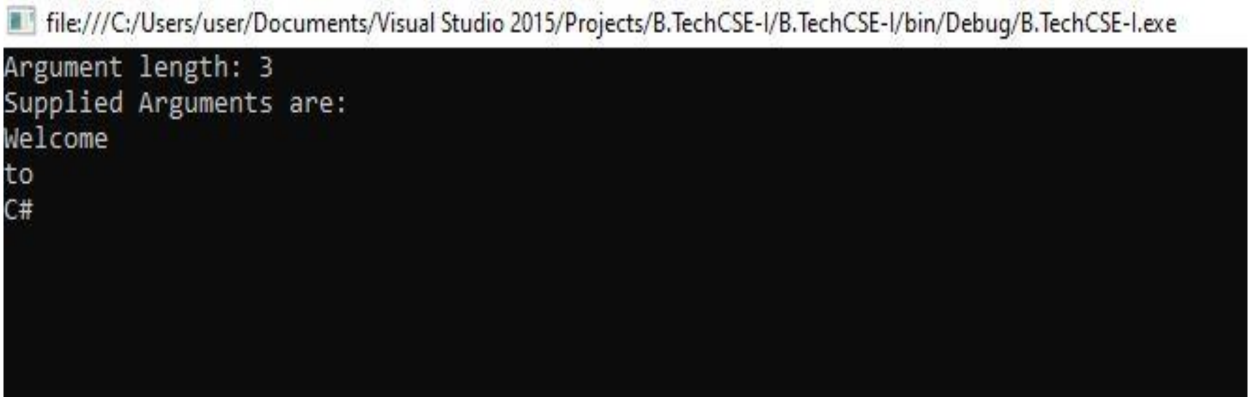
□ Use remote machine

Enable Debuggers

□ Enable native code debugging

□ Enable SQL Server debugging

☑ Enable the Visual Studio hosting process



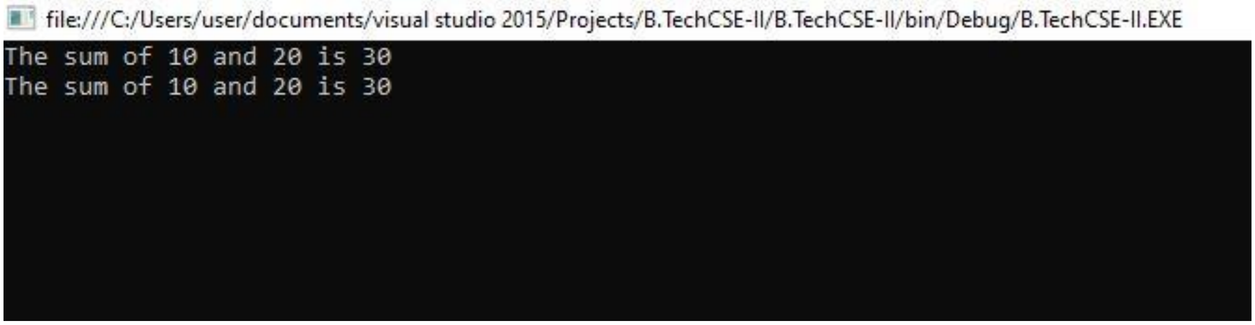
file:///C:/Users/user/Documents/Visual Studio 2015/Projects/B.TechCSE-I/B.TechCSE-I/bin/Debug/B.TechCSE-I.exe

```
Argument length: 3
Supplied Arguments are:
Welcome
to
C#
```

4. Write a program to show the use of Constructor Overloading.

```
using System;
namespace B.TechCSE_II
{
    class ParamConstructor
    {
        public ParamConstructor(int a, int b)
        {
            Console.WriteLine("The sum of {0} and {1} is {2}", a, b, a + b);
        }
        public ParamConstructor(double a, int b)
        {
            Console.WriteLine("The multiplication of {0} and {1} is {2}", a, b, a * b);
        }
    }
    class ParameterizedConstructor
    {
        static void Main(string[] args)
        {
            ParamConstructor Obj_Param1 = new ParamConstructor(10, 20);
            ParamConstructor Obj_Param2 = new ParamConstructor(10, 20);
            Console.Read();
        }
    }
}
```

Output :



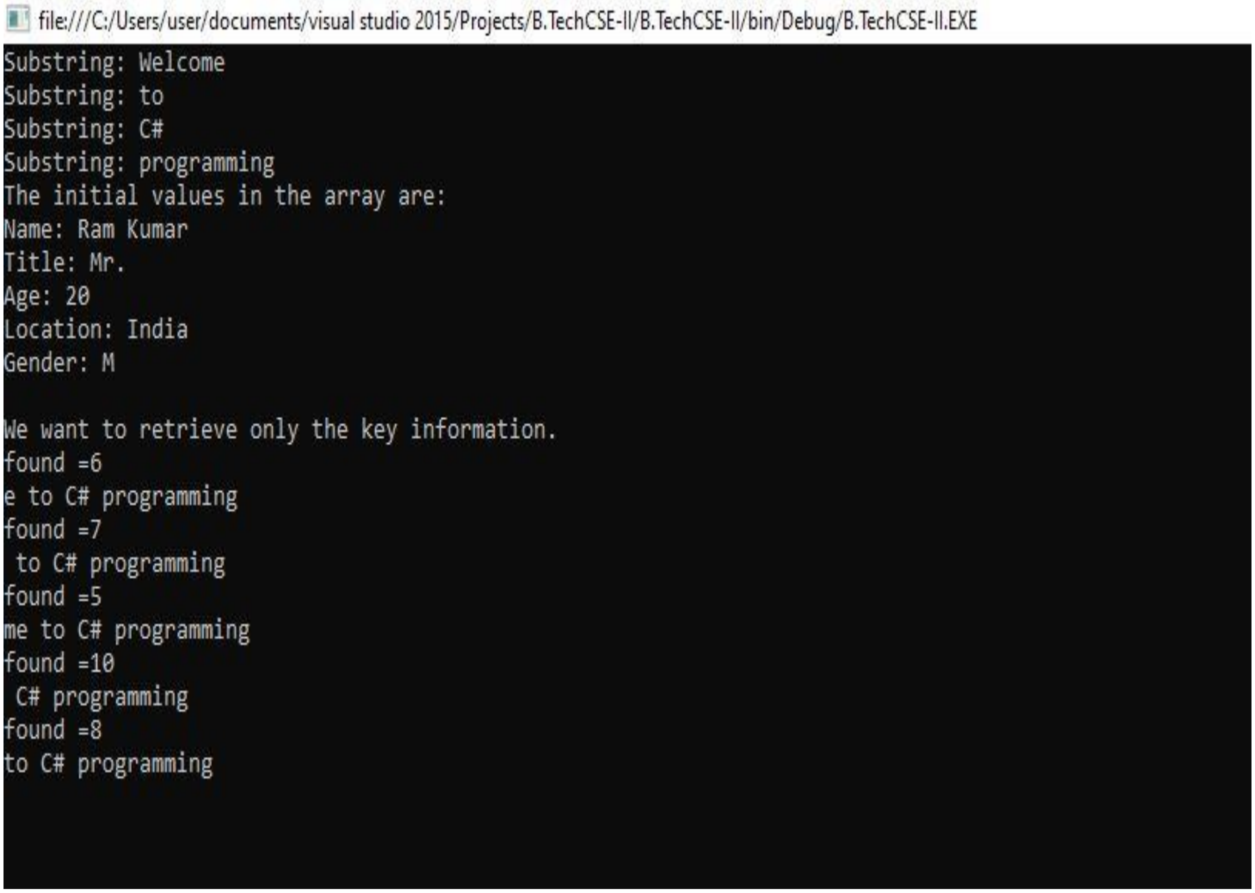
file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

```
The sum of 10 and 20 is 30
The sum of 10 and 20 is 30
```

5. **Write a program to show the string manipulation using string methods and properties.**

```
using System;
namespace B.TechCSE_II
{
    class StringManipulation
    {
        static void Main(string[] args)
        {
            string s = "Welcome to C# programming";
            string[] subs = s.Split(' ');
            foreach (string sub in subs)
            {
                Console.WriteLine("Substring: {0}", sub);
            }
            string[] info = { "Name: Ram Kumar", "Title: Mr.", "Age: 20", "Location:
India", "Gender: M" };
            int found = 0;
            Console.WriteLine("The initial values in the array are:");
            foreach (string sr in info)
            Console.WriteLine(sr);
            Console.WriteLine("\nWe want to retrieve only the key information:");
            foreach (string st in info)
            {
                found = st.IndexOf(": ");
                found = found + 2;
                Console.WriteLine("found =" + found);
                Console.WriteLine(s.Substring(found));
            }
            Console.Read();
        }
    }
}
```

Output :


file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

```
Substring: Welcome
Substring: to
Substring: C#
Substring: programming
The initial values in the array are:
Name: Ram Kumar
Title: Mr.
Age: 20
Location: India
Gender: M

We want to retrieve only the key information.
found =6
e to C# programming
found =7
 to C# programming
found =5
me to C# programming
found =10
 C# programming
found =8
to C# programming
```

6. Write a program for Encapsulation using properties.

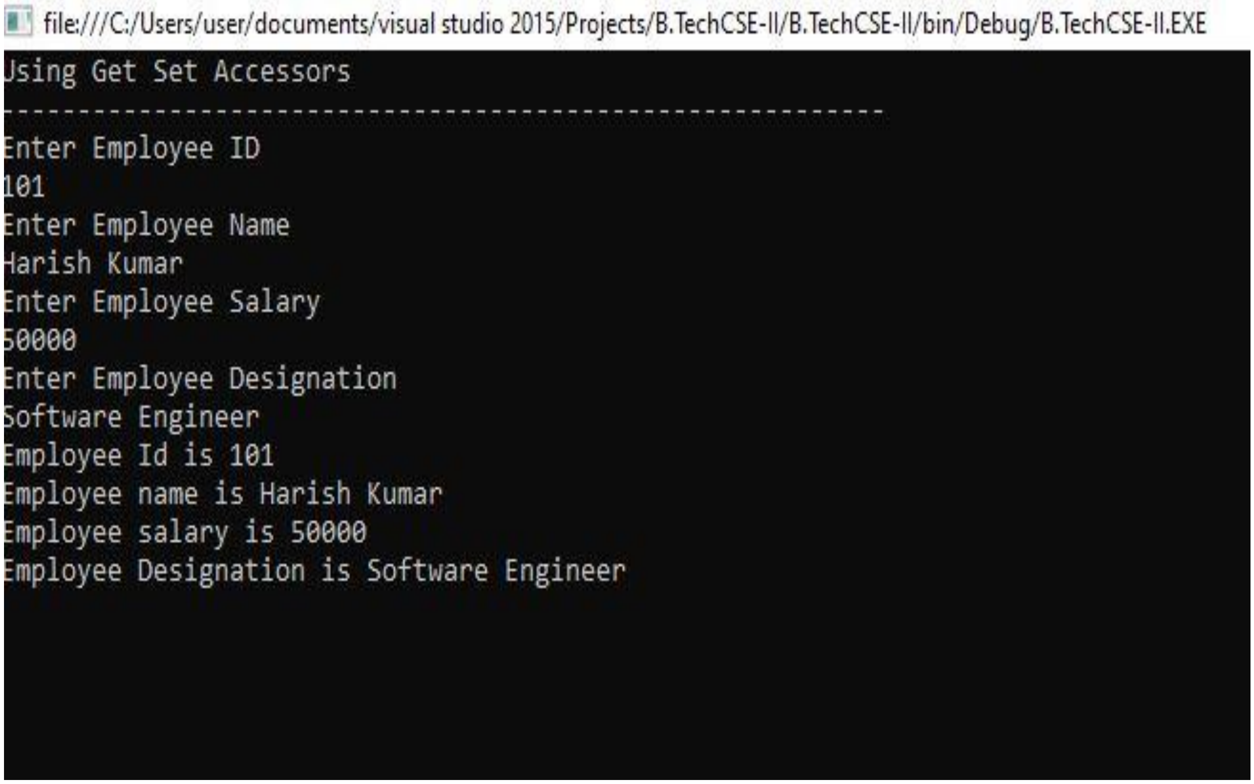
```
using System;
namespace B.TechCSE_II
{
    class EmployeeInfo
    {
        private int EmpID;
        private string Name;
        private int Sal;
        private string Designation;
        public int getEmpId
        {
            set { EmpID = value; }
             get { return EmpID; }

        }
        public string getEmpName
        {
            set
            {
                Name = value;
            }
            get
            {
                return Name;
            }

        }
        public int getEmpSalary
        {
            get { return Sal; }
            set { Sal = value; }
        }
        public string getEmpDesignation
        {
            get { return Designation; }
            set { Designation = value; }
        }

    }
    class Encapsulation_CSharp_UsingGetSet
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Using Get Set Accessors\n--------------------------");
            EmployeeInfo em = new EmployeeInfo();
            Console.WriteLine("Enter Employee ID");
            em.getEmpId = Convert.ToInt16(Console.ReadLine());
            Console.WriteLine("Enter Employee Name");
            em.getEmpName = Console.ReadLine();
            Console.WriteLine("Enter Employee Salary");
            em.getEmpSalary = int.Parse(Console.ReadLine());
            Console.WriteLine("Enter Employee Designation");
            em.getEmpDesignation = Console.ReadLine();
            Console.WriteLine("Employee Id is {0}", em.getEmpId);
            Console.WriteLine("Employee name is {0}",em.getEmpName);
            Console.WriteLine("Employee salary is {0}", em.getEmpSalary);
            Console.WriteLine("Employee Designation is {0}",em.getEmpDesignation);
            Console.Read();
        }
    }
}
```

Output :



file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

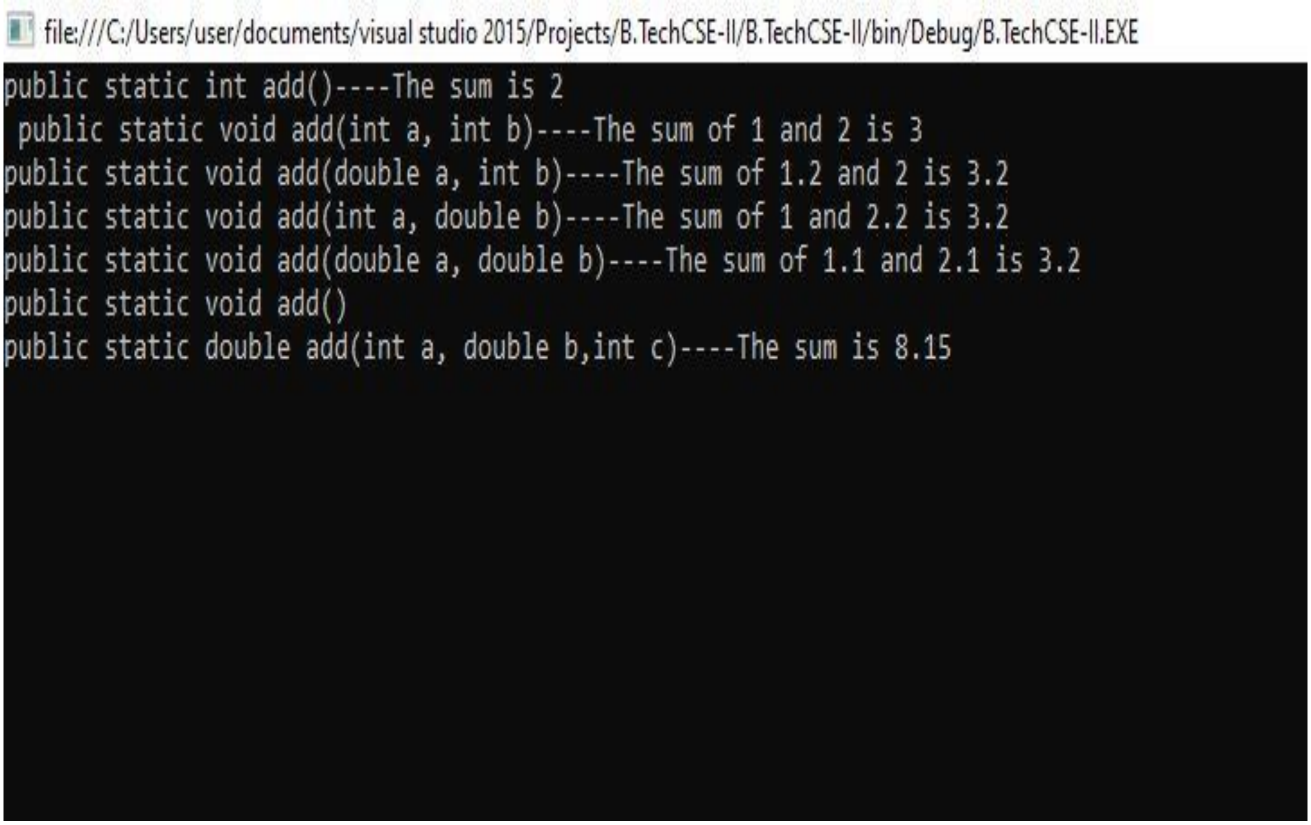
```
Using Get Set Accessors
----------------------------------------------------------
Enter Employee ID
101
Enter Employee Name
Harish Kumar
Enter Employee Salary
50000
Enter Employee Designation
Software Engineer
Employee Id is 101
Employee name is Harish Kumar
Employee salary is 50000
Employee Designation is Software Engineer
```

7. Write a program for polymorphism using overloading and Overriding.

i. Polymorphism overloading

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

namespace B.TechCSE_II
{
    class Add2Num
    {
         public static void add()
        {
           Console.WriteLine("public static void add()");
        }
        public static int add(int a)
        {
           a = 2;
           return a;
        }
        public static void add(double a, int b)
        {
           Console.WriteLine("public static void add(double a, int b)----The sum of {0}
and {1} is {2}", a, b, a + b);
        }
        public static void add(double a, double b)
        {
           Console.WriteLine("public static void add(double a, double b)----The sum of
{0} and {1} is {2}", a, b, a + b);
        }
        public static void add(int a, int b)
        {
           Console.WriteLine(" public static void add(int a, int b)----The sum of {0} and
{1} is {2}", a, b, a + b);
        }
        public static void add(int a, double b)
        {
           Console.WriteLine("public static void add(int a, double b)----The sum of {0}
and {1} is {2}", a, b, a + b);
        }
        public static double add(int a, double b,int c)
        {
            return (a + b + c);
        }
    }
    class PolymorphismConcept
    {
        static void Main(string[] args)
        {
            Console.WriteLine("public static int add()----The sum is {0}",
Add2Num.add(2));
            Add2Num.add(1, 2);
            Add2Num.add(1.2, 2);
            Add2Num.add(1, 2.2);
            Add2Num.add(1.1, 2.1);
            Add2Num.add();
            Console.WriteLine("public static double add(int a, double b,int c)----The sum
is {0}", Add2Num.add(1, 2.15,5));
            Console.Read();
        }
    }
}
```



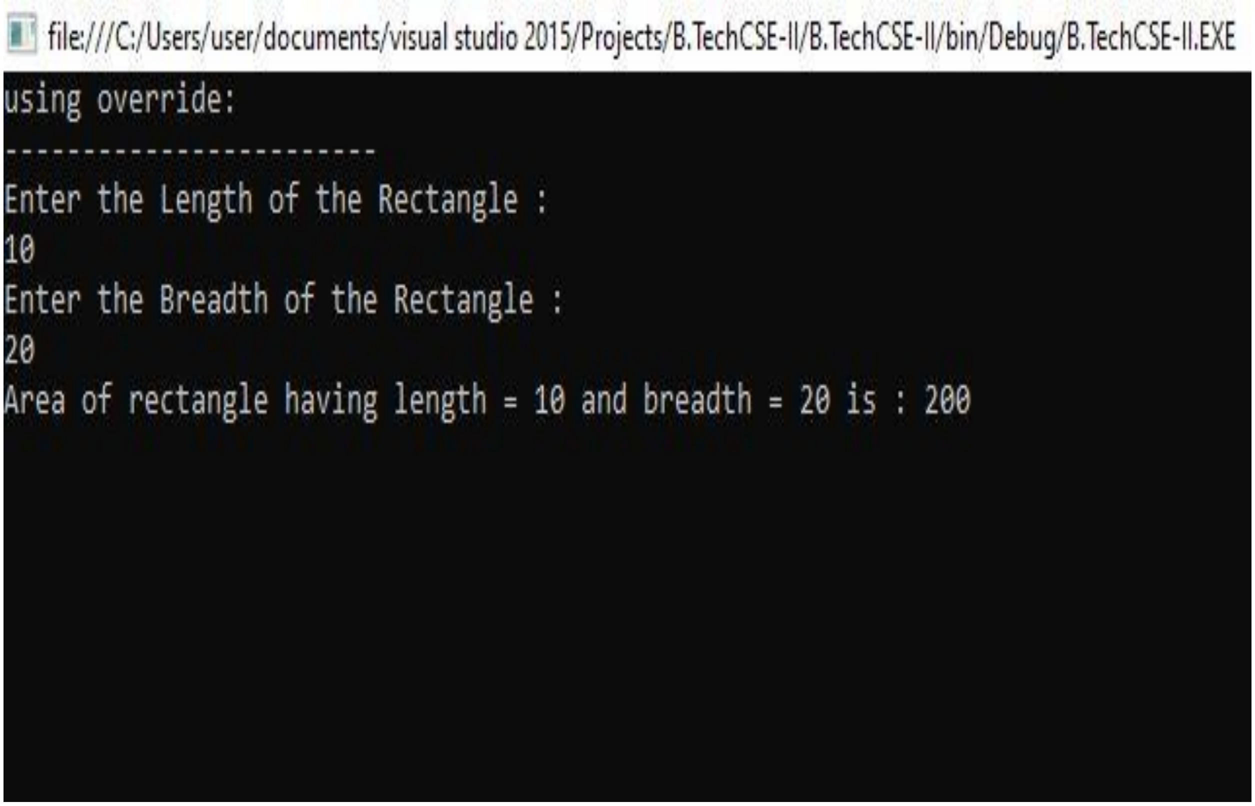
file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

```
public static int add()----The sum is 2
 public static void add(int a, int b)----The sum of 1 and 2 is 3
public static void add(double a, int b)----The sum of 1.2 and 2 is 3.2
public static void add(int a, double b)----The sum of 1 and 2.2 is 3.2
public static void add(double a, double b)----The sum of 1.1 and 2.1 is 3.2
public static void add()
public static double add(int a, double b,int c)----The sum is 8.15
```

ii. Polymorphism Overriding

```
using System;
namespace B.TechCSE_II
{
    abstract class Shape
    {
        public abstract int area();
    }
    class Rectangle1 : Shape
    {
        private int length;
        private int width;
        public Rectangle1(int a , int b )
        {
            length = a; width = b;
        }
        public override int area()
        {
            Console.Write("Area of rectangle having length = {0} and breadth = {1} is : ",length,width);
            return (width * length);
        }
    }
    class RunTimePoly
    {
        static void Main(string[] args)
        {
            Console.WriteLine("using override:\n------------------------");
           int len, wid;
            Console.WriteLine("Enter the Length of the Rectangle :");
            len = Convert.ToInt16(Console.ReadLine());
            Console.WriteLine("Enter the Breadth of the Rectangle :");
            wid = Convert.ToInt16(Console.ReadLine());
            Rectangle1 r = new Rectangle1(len,wid);
            double a = r.area();
            Console.WriteLine(a);
            Console.ReadKey();
        }
    }
}
```

Output :

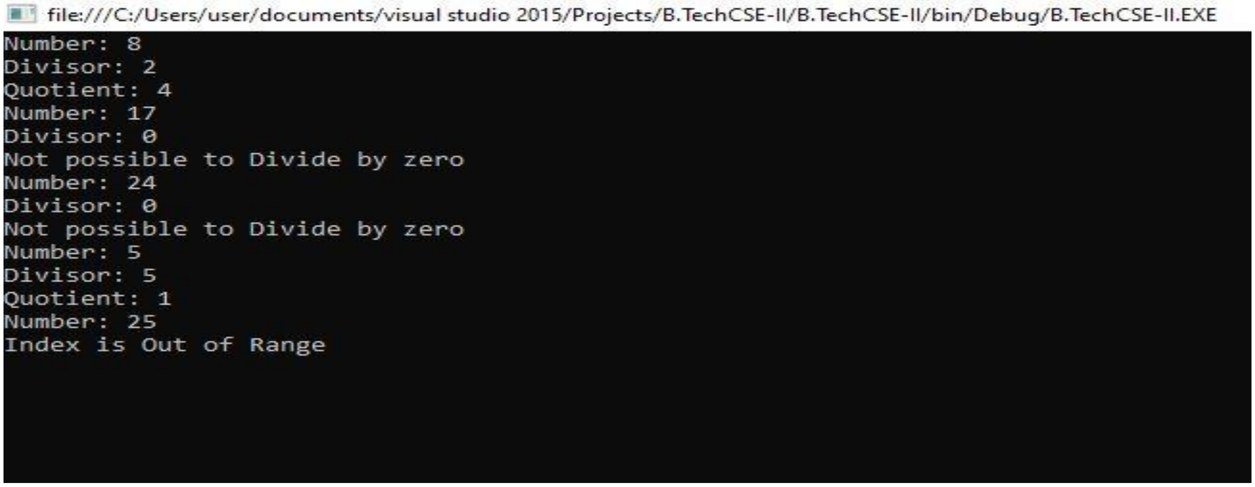

file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

```
using override:
------------------------
Enter the Length of the Rectangle :
10
Enter the Breadth of the Rectangle :
20
Area of rectangle having length = 10 and breadth = 20 is : 200
```

8. Write a program to show handling of multiple Exceptions.

```
using System;
namespace B.TechCSE_II
{
    class MultipleExceptions
    {
        static void Main(string[] args)
        {
            int[] number = { 8, 17, 24, 5, 25 };
            int[] divisor = { 2, 0, 0, 5 };
            for (int j = 0; j < number.Length; j++)
                try
                {
                    Console.WriteLine("Number: " + number[j]);
                    Console.WriteLine("Divisor: " + divisor[j]);
                    Console.WriteLine("Quotient: " + number[j] / divisor[j]);
                }
                catch (DivideByZeroException)
                {
                    Console.WriteLine("Not possible to Divide by zero");
                }
                catch (IndexOutOfRangeException)
                {
                    Console.WriteLine("Index is Out of Range");
                    Console.Read();
                }
        }
    }
}
```

Output :



file:///C:/Users/user/documents/visual studio 2015/Projects/B.TechCSE-II/B.TechCSE-II/bin/Debug/B.TechCSE-II.EXE

```
Number: 8
Divisor: 2
Quotient: 4
Number: 17
Divisor: 0
Not possible to Divide by zero
Number: 24
Divisor: 0
Not possible to Divide by zero
Number: 5
Divisor: 5
Quotient: 1
Number: 25
Index is Out of Range
```