DEV SANSKRITI
VISHWAVIDYALAYA

Practical File

Year - 2018-2021

C#.NET

Submitted To:
Mr. Chandrasekhar Patel
Lecturer
Department of Computer Science

Submitted By:
Amit Singh Negi
BCA (5th Semester)

Department of Computer Science,
Dev Sanskriti Vishwavidyalaya
Gayatrikunj-Shantikunj, Haridwar, U.K. -249411,

www.dsvv.ac.in

INDEX

Signature ……

1. Write a program to print Armstrong Numbers

```
using System;
namespace C_{
    class Program{
        static int order(int n){
            int c = 0;
            while(n > 0){
                n /= 10;
                c++;
            }
            return c;
        }
        public static void Main(string[] args){
            Console.WriteLine("Enter the number of iterations:");
            int t = int.Parse(Console.ReadLine());
            while(t-- > 0){
                Console.WriteLine("Enter the number:");
                int n = int.Parse(Console.ReadLine());
                int noOfDigits = order(n);
                int sum = 0, temp = n;
                while(temp > 0){
                    sum += (int)Math.Pow(temp % 10, noOfDigits);
                    temp /= 10;
                }
                if(sum == n)
                    Console.WriteLine(n + " is an Armstrong number
\n");
                else
                    Console.WriteLine(n + " isn't an Armstrong num
ber\n");
            }
        }
    }
}
```

```
Enter the number of iterations:
3
Enter the number:
1
1 is an Armstrong number

Enter the number:
153
153 is an Armstrong number

Enter the number:
123
123 isn't an Armstrong number
```

2. Write a program to print factorial of a number

```
using System;

namespace myproject
{
    public class Factorial
    {
        public static void Main(string[] args)
        {
            Console.WriteLine("A program to print factorial of the
given number n");
            int m, n, fact = 1;

            Console.WriteLine("Enter the value of n: ");

            n = int.Parse(Console.ReadLine());

            for (m = 1; m <= n; m++)
            {
                fact = fact * m;
            }

            Console.WriteLine(fact);
            Console.ReadLine();
        }
    }
}
```

```
A program to print factorial of the given number n
Enter the value of n:
8
40320
```

3. Write a program to find the GCD of two numbers

```
using System;

namespace myproject
{
    public class GCD
    {
        public static void Main(string[] args)
        {
            int n, a, b, gcd = 1, temp;
            Console.WriteLine("A program to find out the GCD of
two numbers");
            Console.WriteLine("Enter the number of iterations: ");
            n = int.Parse(Console.ReadLine());
            for (int i = 0; i < n; i++)
            {
                Console.WriteLine("Enter the no.:");
                a = int.Parse(Console.ReadLine());
                b = int.Parse(Console.ReadLine());

                while (b != 0)
                {
                    temp = b;
                    b = a % b;
                    a = temp;
                }
                gcd = a;
                Console.WriteLine("GCD = " + gcd);
                Console.ReadLine();
            }
        }
    }
}
```

```
A program to find out the GCD of two numbers
Enter the number of iterations:
3
Enter the no.:
14
46
GCD = 2

Enter the no.:
132
432
GCD = 12
```

4. Write a program to check a number if prime or not.

```
using System;

namespace myproject
{
    public class Prime
    {
        public static void Main(string[] args)
        {
            Console.WriteLine("A program to check the number is
prime or not.");
            int m, n, count = 0, i, j;

            Console.WriteLine("Enter the number of iterations: ");
            j = int.Parse(Console.ReadLine());

            for (i = 0; i < j; i++)
            {
                Console.WriteLine("Enter the number: ");
                n = int.Parse(Console.ReadLine());

                for (m = 2; m <= n / 2; m++){
                    if (n % m == 0)
                       count = count + 1;

                if (count == 0)
                {
                    Console.WriteLine("This number " + n + " is a
prime number");
                }
                else
                {
                    Console.WriteLine("This number " + n + " is
NOT a prime number");
                    count = 0;
                }
            }
        }
}
```

```
Enter the number:
32
This number 32 is NOT a prime number
Enter the number:
2
This number 2 is a prime number
Enter the number:
```

5. Write a program to print Fibonacci Series

```
using System;

namespace myproject
{
    public class Fibonacci
    {
        public static void Main(string[] args)
        {
            Console.WriteLine("A program to print Fibonacci series
of n terms");

            int m, n, a = 0, b = 1;

            Console.WriteLine("Enter the value: ");

            n = int.Parse(Console.ReadLine());

            Console.WriteLine("Fibonacci Series: ");

            for (m = 1; m <= n; m++)
            {
                Console.Write(a + " ");
                int next = a + b;
                a = b;
                b = next;

            }
        }
   }
}
```

```
A program to print Fibonacci series of n terms
Enter the value:
8
Fibonacci Series:
0 1 1 2 3 5 8 13
```

6. Write a program to print the half pyramid pattern

```
using System;

namespace myproject
{
    public class Pattern_HalfPyramid
    {
        public static void Main(string[] args)
        {
            int space, rows;
            Console.WriteLine("A program to print half pyramid
pattern");
            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());

            for (int i = 0; i <= rows; i++)
            {

                for (int star = 0; star < i; star++)
                {
                    Console.Write("*");
                }
                for (space = i; space < rows; space++)
                {
                    Console.Write(" ");
                }
                Console.WriteLine();
                Console.ReadLine();
            }

        }
    }
}
```

```
A program to print half pyramid pattern
Enter the number of rows:
6


*

**

***

****

*****

******
```

7. Write a program to print the half pyramid pattern with numbers

```
using System;

namespace myproject
{
    public class Pattern_HalfPyramidNum
    {
        public static void Main(string[] args)
        {
            int space, rows;
            Console.WriteLine("A program to print half pyramid
pattern of numbers:");
            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());
            Console.WriteLine();

            for (int i = 1; i <= rows; i++)
            {

                for (int num = 1; num <= i; num++)
                {
                    Console.Write(num);
                }
                for (space = i; space < rows; space++)
                {
                    Console.Write(" ");
                }
                Console.WriteLine();
                Console.ReadLine();
            }

        }
    }
}
```

```
A program to print half pyramid pattern of numbers:
Enter the number of rows:
5

1

12

123

1234

12345
```

8. Write a program to print the half pyramid inverse pattern

```
using System;

namespace myproject
{
    public class Pattern_HalfInversePyramid
    {
        public static void Main(string[] args)
        {
            int space, rows;
            Console.WriteLine("A program to print half pyramid
inverse pattern");
            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());
            Console.WriteLine();

            for (int i = 0; i <= rows; i++)
            {

                for (int star = rows; star > i; star--)
                {
                    Console.Write("*");
                }
                for (space = i; space < rows; space++)
                {
                    Console.Write(" ");
                }
                Console.WriteLine();
                Console.ReadLine();
            }

        }
    }
}
```

```
A program to print half pyramid inverse pattern
Enter the number of rows:
6

******

*****

****

***

**

*
```

9. Write a program to print the pyramid pattern

```
using System;
namespace myproject
{
    public class Pattern_pyramid
    {
        public static void Main(string[] args)
        {
            int space, rows;

            Console.WriteLine("A program to print pyramid
pattern");

            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());

            for (int i = 1; i <= rows; i++)
            {
                for (space = i; space < rows; space++)
                {
                    Console.Write(" ");
                }
                for (int star = 1; star < (i * 2); star++)
                {
                    Console.Write("*");
                }

                Console.WriteLine();
                Console.ReadLine();
            }
        }
    }
}
```

```
A program to print pyramid pattern
Enter the number of rows:
5
    *
   ***
  *****
 *******
*********
```

10. Write a program to print the inverse pyramid pattern

```
using System;
namespace myproject
{
    public class Pattern_PyramidInverse
    {
        public static void Main(string[] args)
        {
            int space, rows;
            Console.WriteLine("A program to print inverse pyramid
pattern");

            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());
            for (int i = rows; i >= 1; i--)
            {
                for (space = i; space <= rows; space++)
                {
                    Console.Write(" ");
                }
                for (int star = (i * 2); star > 1; star--)
                {
                    Console.Write("*");
                }
                Console.WriteLine();
                Console.ReadLine();
            }
        }
    }
}
}
```

```
A program to print inverse pyramid pattern
Enter the number of rows:
6
 **********
  *********
   *******
    *****
     ***
      *
```

11. Write a program to print the diamond pattern

```
using System;

namespace C_{
    class Program{
        public static void Main(string[] args){
           int rows, i, j, space;

            Console.Write("Enter the number of rows: ");
            rows = int.Parse(Console.ReadLine());

            for (i = 0; i <= rows / 2; i++)
            {
                for (space = i; space < rows / 2; space++)
                {
                    Console.Write("  ");
                }
                for (j = 0; j <= i * 2; j++)
                {
                    Console.Write("* ");
                }
                Console.WriteLine();
            }
            for (i = rows / 2 + 1; i > 0; i--)
            {
                for (space = i; space <= rows / 2 + 1; space++)
                {
                    Console.Write("  ");
                }
                for (j = i * 2 - 4; j >= 0; j--)
                {
                    Console.Write("* ");
                }
                Console.WriteLine();

            }
        }
    }
}
```

```
Enter the number of rows: 7
      *
    * * *
  * * * * *
* * * * * * *
  * * * * *
    * * *
      *
```

12. Write a program to print the Pascal's triangle

```
using System;
namespace myproject{
    public class Pascal_Triangle{
        public static int Factorial(int fact){
            int m, f = 1;
            for (m = 1; m <= fact; m++)
                f = f * m;
            return f;
        }

        public static int Ncr(int a, int b){
            return Factorial(a) / (Factorial(b) * Factorial(a -
b));
        }

        public static void Main(string[] args){
            int space, rows, c;
            Console.WriteLine("A program to print the Pascal
triangle.");
            Console.WriteLine("Enter the number of rows:");
            rows = int.Parse(Console.ReadLine());

            for (int i = 0; i <= rows; i++){
                for (space = i; space < rows; space++)
                    Console.Write(" ");
                for (int j = 0; j <= i; j++){
                    c = Ncr(i, j);
                    Console.Write(c + " ");
                }
                Console.WriteLine();
                Console.ReadLine();
   }}}}
```

```
A program to print the Pascal triangle.
Enter the number of rows:
6
      1

     1 1

    1 2 1

   1 3 3 1

  1 4 6 4 1

 1 5 10 10 5 1
```

13. Write a program to compare two string without using string library functions

```
using System;

namespace C_{
    class Program{
        public static void Main(string[] args){
            string str1, str2;
            int flag = 0;

            Console.WriteLine("A program to compare two strings.")
;
            Console.WriteLine("Enter string 1: ");
            str1 = Console.ReadLine();

            Console.WriteLine("Enter string 2: ");
            str2 = Console.ReadLine();

            for(int i=0; i<str1.Length; i++){
                if(str1[i]!=str2[i]){
                    flag = 0;
                    break;
                }
                else{
                    flag = 1;
                }
            }

            if(flag==0)
                Console.WriteLine(str1 + " and " + str2 + " are NO
T equal");
            else
            Console.WriteLine(str1 + " and " + str2 + " are Equal"
);
        }
    }
}
```

```
A program to compare two strings.
Enter string 1:
amit
Enter string 2:
amit
amit and amit are Equal
```

14. Write a program to count a total number of alphabets, digits and special characters in a string

```
using System;

namespace String
{
    public class StringCount
    {
        public static void Main(string[] args)
        {
            string str;
            int alpha = 0, digit = 0, sym = 0;

            Console.WriteLine("Enter the main string: ");
            str = Console.ReadLine();

            foreach (char s in str)
            {
                if (s >= 65 && s <= 90 || s >= 97 && s <= 122)
                    alpha += 1;
                else if (s >= 48 && s <= 57)
                    digit += 1;
                else
                    sym += 1;
            }
            Console.WriteLine();

            Console.WriteLine("Number of Alphabets: " + alpha);
            Console.WriteLine("Number of Digits: " + digit);
            Console.WriteLine("Number of Special Characters: " +
sym);
            Console.ReadLine();
        }
    }
}
```

```
Enter the main string:
crsgj469bvkmnfdd#$$#^*%*&@

Number of Alphabets: 13
Number of Digits: 3
Number of Special Characters: 10
```

15. Write a program to copy one string to another string

```
using System;

namespace C_
{
    class Program
     {
        public static void Main(string[] args)
          {

            string s1, s2 = "";

            Console.WriteLine("A program to copy one string to ano
ther string");

            Console.WriteLine("Enter the string: ");
            s1 = Console.ReadLine();

            foreach (char c in s1)
            {
                s2 += c;
            }

            Console.WriteLine("\nString Copied\n");
            Console.WriteLine("Copied String is: " + s2);

        }
    }
}
```

```
A program to copy one string to another string
Enter the string:
Hello!!!

String Copied

Copied String is: Hello!!!
```

16. Write a program to find maximum occurring character in a string

```
using System;

namespace Strings
{
    public class StringMax
    {
        public static void Main(string[] args)
        {
            string str;
            int[] count = new int[256];

            Console.WriteLine("Enter the string: ");
            str = Console.ReadLine();

            for (int i = 0; i < str.Length; i++)
            {
                count[str[i]]++;
            }

            int max = -1;
            char result = ' ';

            for (int i = 0; i < str.Length; i++)
            {
                if (max < count[str[i]])
                {
                    max = count[str[i]];
                    result = str[i];
                }
            }
            Console.WriteLine("Maximum occuring character in
string: " + result);
        }
    }
}
```

```
Enter the string:
console
Maximum occuring character in string: o
```

17. Write a program to check whether a given substring is present in the given string

```
using System;

namespace C_{
    class Program{
        public static void Main(string[] args){
          string str, substr;
          Console.WriteLine("\nEnter the string: ");
          str = Console.ReadLine();
          Console.WriteLine("\nEnter the Sub-string");
          substr = Console.ReadLine();

          int flag = 0;
          for (int i = 0; i <= str.Length - substr.Length; i++){
              for (int j = i; j < i + substr.Length; j++){
                  flag = 1;
                  if (str[j] != substr[j - i]) {
                      flag = 0;
                      break;
                  }
              }
              if (flag == 1)
                  break;
           }

           if (flag == 1)
               Console.WriteLine("\nThe substring is present in gi
ven String");
           else
               Console.WriteLine("The substring is NOT present in
given String");
        }
    }
}
```

```
Enter the string:
Hello

Enter the Sub-string
Hell

The substring is present in given String
```

18. Write a program for Encapsulation

```
using System;
namespace test
{
    class Student
    {
        private int roll;
        private string name;
        public int Roll
        {
            get
            {
                return roll;
            }
            set
            {
                roll = value;
            }
        }
        public string Name
        {
            get
            {
                return name;
            }
            set
            {
                name = value;
            }
        }
    }
    class Program
    {
        static void Main(string[] args)
        {
            Student A = new Student();
            A.Roll = 14;
            A.Name = "Pranav";
            Console.WriteLine("\nRoll: " + A.Roll);
            Console.WriteLine("\nName: " + A.Name);
        }
    }
}
```

```
Roll: 1

Name: Abhi
```

19. Write a program for Abstraction

```
using System;

namespace OOPs{
    public class Abstraction{
        abstract class Cs{
            public abstract void Fun();
        }

        private class Good : Cs{
            public override void Fun(){
                Console.WriteLine("C# is Good");
            }
        }

        private class Best : Cs{
            public override void Fun(){
                Console.WriteLine("C# is Best");
            }
        }

        private class Better : Cs{
            public override void Fun(){
                Console.WriteLine("C# is Better");
            }
        }

        public class MyClass
        {
            public static void Main()
            {
                Cs c;
                c = new Good();
                c.Fun();
                c = new Best();
                c.Fun();
                c = new Better();
                c.Fun();
                Console.ReadLine();
            }
        }
    }
}
```

```
C# is Good
C# is Best
C# is Better
```

20. Write a program for single Inheritance

```
using System;

namespace OOPs
{
    public class Inheritence
    {
        class MainClass
        {
            public void Print()
            {
                Console.WriteLine("Print");
            }
        }
        class Subclass : MainClass
        {
            void Print1()
            {
                Console.WriteLine("Print 2");
            }
            static void Main(string[] args)
            {
                Subclass s = new Subclass();

                s.Print();
                s.Print1();
                Console.ReadLine();
            }
        }
    }
}
```

```
Print
Print 2
```

21.Write a program for Multilevel Inheritance

```
using System;
namespace OOPs
{
    public class MultiInheritence
    {
        class MainClass
        {
            public void Print()
            {
                Console.WriteLine("Print");
            }
        }
        class Subclass : MainClass
        {
            public void Print1()
            {
                Console.WriteLine("Print 1");
            }

        }
        class Subclass2 : Subclass
        {

            public void Print2()
            {
                Console.WriteLine("Print 2");
            }
            static void Main(string[] args)
            {
                Subclass2 s = new Subclass2();
                s.Print();
                s.Print1();
                s.Print2();
                Console.ReadLine();
            }

        }
    }
}
```

```
Print
Print 1
Print 2
```

22. Write a program for Multiple Inheritance

```
using System;

namespace OOPs
{
    public class MultipleInheritence
    {
        class MainClass
        {
            public void Print()
            {
                Console.WriteLine("Print");
            }
        }
        interface MainClass1
        {
            void Print1();
        }
        class Subclass : MainClass, MainClass1
        {
            void Print2()
            {
                Console.WriteLine("Print 2");
            }
            public void Print1()
            {
                Console.WriteLine("Print 1");
            }
            static void Main(string[] args)
            {
                Subclass s = new Subclass();
                s.Print();
                s.Print1();
                s.Print2();
                Console.ReadLine();
            }
        }
    }
}
```

```
Print
Print 1
Print 2
```

23. Write a program for Method Overloading

```
using System;

namespace OOPs
{
    public class MethodOverloading
    {
        static int Sum(int a, int b)
        {
            return a + b;
        }
        static double Sum(double a, double b)
        {
            return a + b;
        }

        public static void Main()
        {
            int sum1 = Sum(54, 74);

            double sum2 = Sum(34.84, 65.16);

            Console.WriteLine(sum1);
            Console.WriteLine(sum2);
        }
    }
}
```

```
128
100
```

24. Write a program for Method Overriding

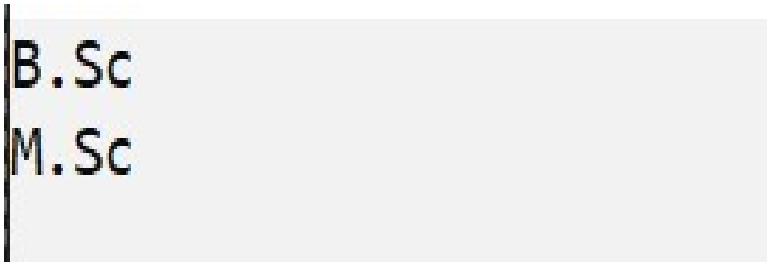
```
using System;

namespace OOPs
{
    public class MethodOverriding
    {
        public class Cs
        {
            public virtual void Fun()
            {
                Console.WriteLine("B.Sc");
            }
        }

        public class MCA : Cs
        {
            public override void Fun()
            {
                Console.WriteLine("M.Sc");
            }
        }

        private class BCA : Cs
        {
            public override void Fun()
            {
                Console.WriteLine("B.Sc");
            }
        }

        public static void Main()
        {
            Cs c;
            c = new BCA();
            c.Fun();
            c = new MCA();
            c.Fun();
        }
    }
}
```



25. Write a program for Interface

```
using System;

namespace OOPs
{
    public class Interface
    {
        public interface Cs
        {
            void Fun();
        }

        private class Bca : Cs
        {
            public void Fun()
            {
                Console.WriteLine("BCA");
            }
        }
        private class Mca : Cs
        {
            public void Fun()
            {
                Console.WriteLine("MCA");
            }
        }

        public class MyClass
        {
            public static void Main(string[] args)
            {
                Cs c;
                c = new Bca();
                c.Fun();
                c = new Mca();
                c.Fun();
                Console.ReadLine();
            }
        }
    }
}
```

```
BCA
MCA
```

26. Write a program to demonstrate namespace

```
using System;
namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            A.B test = new A.B();
            test.C();
        }
    }
}
namespace A
{
    public class B {
        public void C(){
            Console.WriteLine("Hi");
        }
    }
}
```

```
Hi
```

27. Write a program for exception handling through try and catch

```
using System;
namespace ConsoleApp{
    class Program{
        static void Main(string[] args){
            int[] myNumbers = { 1, 2, 3,4 };
            try
            {
                Console.WriteLine(myNumbers[5]);
            }
            catch (Exception c)
            {
                Console.WriteLine(c.Message);
            }
        }
    }
}
```

```
Index was outside the bounds of the array.
```

28. Write a program to demonstrate properties

```
using System;
namespace OOPBasics
{
    public class Properties
    {
        public class CSharp
        {
            public int roll;
            public string name;
            private string Result;
            public CSharp(int a, string b, string c){
                roll = a;
                name = b;
                Result1 = c;
            }
            public string Result1{
                get{
                    return Result;
                }
                set{
                    if (value == "good" || value == "average" ||
value == "bad"){
                        Result = value;
                    }else{
                        Result = "Not Valid";
                    }
                }
            }
        }
    }
    class Program{
        static void Main(string[] args){
            Properties.CSharp c1 = new Properties.CSharp(1, "Xyz",
"good");
            Properties.CSharp c2 = new Properties.CSharp(2, "Ijk",
"5");
            Console.WriteLine(c1.roll + " " + c1.name + " " +
c1.Result1);
            Console.WriteLine(c2.roll + " " + c2.name + " " +
c2.Result1);
            Console.ReadLine();
        }
    }
}
```

```
1 Xyz good
2 Ijk Not Valid
```

29. Write a program to demonstrate constructor

```
using System;
using OOPBasics;

namespace OOPBasics
{
    class Constructor
    {
        public class CSharp
        {
            public int roll;
            public string name;
            public int marks;

            public CSharp(int a, string b, int c)
            {
                roll = a;
                name = b;
                marks = c;
            }
        }
    }
    public class MyClass
    {
        public static void Main(string[] args)
        {
            Constructor.CSharp a = new Constructor.CSharp(14,
"Pranav", 7);
            Console.WriteLine("Roll: " + a.roll + "\nName: " +
a.name + "\nMarks: " + a.marks);
            Console.ReadLine();
        }
    }
}
```

```
Enter your details:

Roll: 1

Name: XYZ

Marks: 70
```

30. Write a program to demonstrate threading.

```
using System;
using System.Threading;

namespace Threading
{
    class Program
    {
        static void T1()
        {
            Console.WriteLine("Thread1 Started");
            Thread.Sleep(5000);
            Console.WriteLine("Thread1 executing");
            Thread.Sleep(5000);
            Console.WriteLine("Thread1 executing");
        }

        static void T2()
        {
            Console.WriteLine("Thread2 Started");
            Thread.Sleep(5000);
            Console.WriteLine("Thread2 executing");
            Thread.Sleep(5000);
            Console.WriteLine("Thread2 executing");
        }

        public static void Main()
        {
            Thread t1 = new Thread(T1);
            Thread t2 = new Thread(T2);

            t1.Start();
            t2.Start();
        }
    }
}
```

```
Thread2 Started
Thread1 Started
Thread1 executing
Thread2 executing
Thread1 executing
Thread2 executing
```

31. Write a program to demonstrate indexer

```
using System;
class IndexerCreation
{
    private string[] val = new string[4];
    public string this[int index]
    {
        get
        {
            return val[index];
        }
        set
        {
            val[index] = value;
        }
    }
}

class MyClass
{

    public static void Main()
    {
        IndexerCreation ic = new IndexerCreation();
        ic[0] = "Hardik";
        ic[1] = "Harsh";
        ic[2] = "Kshitiz";
        ic[3] = "Neeraj";

        Console.Write("Printing values stored in objects used as
arrays\n");
        Console.WriteLine(ic[0] + "\n" + ic[1] + "\n" + ic[2] +
"\n" + ic[3]);
        Console.ReadLine();

    }
}
```

```
Printing values stored in objects used as arrays
Hardik
Harsh
Kshitiz
Neeraj
```

32. Write a program to access data from database using ADO.NET

```
using System;
using MySqlConnector;

namespace test {
     class Program    {
           public static void Main(string[] args)        {
               string connectionString;
               MySqlConnection conn;
               connectionString = @"Data Source=localhost;Initial
               Catalog=test;User ID=myuser;Password=password";
               conn = new MySqlConnection(connectionString);
          conn.Open();
               Console.WriteLine("Connected to Database!");
               string query = "select * from student";

               MySqlCommand cmd = new MySqlCommand(query, conn);

               MySqlDataReader dataReader = cmd.ExecuteReader();

               Console.WriteLine(dataReader.GetName(0)+"
               "+dataReader.GetName(1)+"
               "+dataReader.GetName(2));

               while(dataReader.Read()){
                         Console.WriteLine(dataReader.GetValue(0)+"
                         "+ dataReader.GetValue(1)+"
                         "+dataReader.GetValue(2));
               }
               conn.Close();
               }
     }
}
```

```
Connected to Database!
Roll    Name    Marks
1824001 Abhijeet 7
1824003 Aman 7
1824004 Amisha 7
1824005 Amit 8
1824006 Aniket 8
```