



DEV SANSKRITI  
VISHWAVIDYALAYA



# Practical File



Year - 2018-2021

## C#.NET

### Submitted To:

Mr. Chandrasekhar Patel  
Lecturer  
Department of Computer Science

### Submitted By:

Aniket Kumar  
BCA (5<sup>th</sup> Semester)  
Department of Computer Science

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

[www.dsvv.ac.in](http://www.dsvv.ac.in)

---

# INDEX

S. No.	Task	Page No
1	Write a program for Armstrong Numbers	1
2	Write a program to print factorial of a number	3
3	Write a program to find the GCD of two numbers	4
4	Write a program to check if a number is prime number	5
5	Write a program to print the fibonacci series	7
6	Write a program to print the half pyramid pattern	8
7	Write a program to print the half pyramid pattern with numbers	10
8	Write a program to print the half pyramid inverse pattern	12
9	Write a program to print the pyramid pattern	14
10	Write a program to print the inverse pyramid pattern	15
11	Write a program to print the diamond pattern	16
12	Write a program to print the Pascal's triangle	18
13	Write a program to compare two string without using string library functions	20
14	Write a program to count a total number of alphabets, digits and special characters in a string	22
15	Write a program to copy one string to another string	24
16	Write a program to find maximum occurring character in a string	25
17	Write a program to check whether a given substring is present in the given string	26
18	Write a program for Encapsulation	28
19	Write a program for Abstraction	30
20	Write a program for single Inheritance	32
21	Write a program for Multilevel Inheritance	33
22	Write a program for multiple Inheritance	34
23	Write a program for method overloading	35
24	Write a program for method overriding	36

25	Write a program for Interface	37
26	Write a program for Namespace	38
27	Write a program for exception handling through try and catch	39
28	Write a program for constructor	40
29	Write a program for Properties	42
30	Write a program for Threading	43
31	Write a program for Indexer	44
32	Write a program to access data from database using ADO.NET	45

.....  
**Signature**

## 1. Write a program for Armstrong Numbers

```
//A program to check if the given number is Armstrong Number or
not

using System;
namespace myproject
{
    public class Armstrong
    {
        public static void main(string[] args)
        {
            Console.WriteLine("A program to check if the given
number is Armstrong Number or not");
            int n, m, num ,d;
            double result = 0, number;

            Console.WriteLine("Enter the number of iterations: ");
            n = int.Parse(Console.ReadLine());
            for (m = 0; m < n; m++)
            {
                Console.WriteLine("Enter the number of digits: ");
                d = int.Parse(Console.ReadLine());
                Console.WriteLine("Enter the number: ");
                num = int.Parse(Console.ReadLine());
                number = num;
                for(int i=0;i<d;i++)
                {
                    int rem = num % 10;
                    //split last digit from number
                    double power = Math.Pow(rem, d);
                    //Console.WriteLine(power);
                    result = result + power;
                    num = num / 10;
                }
                if(number == result)
                {
                    Console.WriteLine(number + " is an Armstrong
number.");
                    result = 0;
                }
                else
                {
                    Console.WriteLine(number + " is NOT an
Armstrong number");
                    result = 0;
                }
            }
        }
    }
}
```

```
}  
    }  
    }  
}
```

### Output:

```
A program to check if the given number is Armstrong Number or not  
Enter the number of iterations:  
3  
Enter the number of digits:  
3  
Enter the number:  
153  
153 is an Armstrong number.  
Enter the number of digits:  
3  
Enter the number:  
123  
123 is NOT an Armstrong number  
Enter the number of digits:  
4  
Enter the number:  
2344  
2344 is NOT an Armstrong number  
  
Process finished with exit code 0.
```

## 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);
        }
    }
}
```

### Output:

```
A program to print factorial of the given number n
Enter the value of n:
7
5040

Process finished with exit code 0.
```

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

```
//A program to find out 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 numbers:");
                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);
            }
        }
    }
}
```

#### Output:

```
A program to find out the GCD of two numbers
Enter the number of iterations:
2
Enter the numbers:
12
48
GCD = 12
Enter the numbers:
432
134
GCD = 2

Process finished with exit code 0.
```

#### 4. Write a program to check if a number is prime number

```
// A program to check whether the number is prime or not.

using System;
namespace myproject
{
    public class Prime
    {
        public static void Main(string[] args)
        {
            Console.WriteLine("A program to check whether 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());
                if (n == 0)
                {
                    Console.WriteLine("The number " + n + " is NOT
a prime number");
                }
                else
                {
                    for (m = 2; m <= n / 2; m++)
                    {
                        if (n % m == 0)
                        {
                            count = count + 1;
                        }
                    }
                    if (count == 0)
                    {
                        Console.WriteLine("The number " + n + " is
a prime number");
                    }
                    else
                    {
                        Console.WriteLine("The number " + n + " is
NOT a prime number");
                    }
                }
            }
        }
    }
}
```



```

        }
        count = 0;
    }
}

```

### Output:

```

A program to check whether the number is prime or not.
Enter the number of iterations:
5
Enter the number:
0
The number 0 is NOT a prime number
Enter the number:
6
The number 6 is NOT a prime number
Enter the number:
34
The number 34 is NOT a prime number
Enter the number:
2
The number 2 is a prime number
Enter the number:
785
The number 785 is NOT a prime number

Process finished with exit code 0.

```

## 5. Write a program to print the fibonacci series

```
//A program to print fibonacci series of n terms
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 of n: ");
            n = int.Parse(Console.ReadLine());
            Console.WriteLine("Fibonacci Series: ");
            for (m = 1; m <= n; m++)
            {
                Console.WriteLine(a);
                int next = a + b;
                a = b;
                b = next;
            }
        }
    }
}
```

**Output:**

```
A program to print fibonacci series of n terms
Enter the value of n:
15
Fibonacci Series:
0
1
1
2
3
5
8
13
21
34
55
89
144
233
377

Process finished with exit code 0.
```

## 6. Write a program to print the half pyramid pattern

```
/*A program to print the following 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 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();
            }
        }
    }
}
```

## Output:

```
A program to print pyramid pattern
Enter the number of rows:
7

*
**
***
****
*****
*****
*****
*****

Process finished with exit code 0.
```

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

/\*A program to print the following pyramid pattern:

```
1
12
123
1234
12345
123456
*/
```

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

## Output:

```
A program to print pyramid pattern of numbers:  
Enter the number of rows:  
7  
  
1  
12  
123  
1234  
12345  
123456  
1234567  
  
Process finished with exit code 0.
```

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

```
/*A program to print the following pyramid pattern:
*****
*****
****
***
**
*
*/

using System;
namespace myproject
{
    public class Pattern_HalfPyramidInverse
    {
        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());
            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();
            }
        }
    }
}
```

## Output:

```
A program to print pyramid pattern
Enter the number of rows:
7

*****
*****
*****
****
***
**
*

Process finished with exit code 0.
```



## 9. Write a program to print the pyramid pattern

/\*A program to print the following 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();

}

}

}

}

**Output:**

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

    *
  ***
*****
*****
*****

Process finished with exit code 0.
```

## 10. Write a program to print the inverse pyramid pattern

/\*A program to print the following 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 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();

}

}

}

}

**Output:**

A program to print pyramid pattern

Enter the number of rows:

5

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

Process finished with exit code 0.

## 11. Write a program to print the diamond pattern

```
/*
A program to print the following diamond like pattern:
    *
   ***
  *****
 *****
  *****
   ***
    *
*/

using System;
namespace myproject
{
    public class Pattern_diamond
    {
        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 >= 1; i--)
            {
                for (space = i; space <= rows / 2 + 1; space++)
                {
                    Console.Write(" ");
                }
                for (j = i * 2 - 4; j >= 0; j--)
                {
                    Console.Write("*");
                }
            }
        }
    }
}
```

```

    }
    Console.WriteLine();
}
}
}
}
}

```

## Output:

```

Enter the number of rows: 10

      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
 *****
  *****
   ***
    *

Process finished with exit code 0.

```

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

```
/*
A program to print the Pascal's triangle.
    1
   1 1
  1 2 1
 1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
*/
using System;
namespace myproject
{
    public class Pascal
    {
        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's
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();
    }
}

```

## Output:

```

A program to print the Pascal's triangle.
Enter the number of rows:
7
    1
  1 1
 1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1

Process finished with exit code 0.

```

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

```
/* A program in C# to compare two strings. */

using System;
namespace String
{
    public class StringCompare
    {
        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
NOT equal");
            }
            else if(flag==1)
            {
                Console.WriteLine(str1 + " and " + str2 + " are
Equal");
            }
        }
    }
}
```

```
}  
}
```

### Output:

```
A program to compare two strings.  
Enter string 1:  
Aniket  
Enter string 2:  
niket  
Aniket and niket are NOT equal  
  
Process finished with exit code 0.
```



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

/\* A program in C# to count the total number of alphabets, digits and special characters. \*/

```
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) //Alphabet
                {
                    alpha += 1;
                }
                else if(s>=48&&s<=57) //Digits
                {
                    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);
        }
    }
}
```

## Output:

```
Enter the main string:  
nxheirly43679778%^&%&^nhjk  
  
Number of Alphabets: 12  
Number of Digits: 8  
Number of Special Characters: 6  
  
Process finished with exit code 0.
```

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

/\* A program in C# to copy one string to another without using any library function. \*/

```
using System;
namespace String
{
    public class StringCopy
    {
        public static void Main(string[] args)
        {
            string s1, s2="Not Copied";

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

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

            foreach(char a in s1)
            {
                s2 = s1;
            }
            Console.WriteLine("String Copied");
            Console.WriteLine("Copied String is: " + s2);
        }
    }
}
```

### Output:

```
A program to copy string
Enter the string:
Aniket
String Copied
Copied String is: Aniket

Process finished with exit code 0.
```

## 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 occurring character: "+
result);
            Console.WriteLine("Frequency of "+result+" is "+max);
        }
    }
}
```

### Output:

```
Enter the string:
dotnet
Maximum occurring character: t
Frequency of t is 2

Process finished with exit code 0.
```

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

```
using System;

namespace Strings
{
    public class StringSubString
    {
        public static void Main(string[] args)
        {
            string str, substr;
            Console.WriteLine("A program to check if the given
substring is present in string.");
            Console.WriteLine("Enter the string: ");
            str = Console.ReadLine();
            Console.WriteLine("Enter 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("The substring is present in
String");
            }
            else
            {
                Console.WriteLine("The substring is NOT present in
String");
            }
        }
    }
}
```

```
}  
}
```

## Output:

```
A program to check if the given substring is present in string.  
Enter the string:  
Aniket  
Enter the Sub-string  
kr  
The substring is NOT present in String  
  
Process finished with exit code 0.
```

```
A program to check if the given substring is present in string.  
Enter the string:  
Aniket  
Enter the Sub-string  
ket  
The substring is present in String  
  
Process finished with exit code 0.
```

## 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 = 6;
            A.Name = "Aniket";
            Console.WriteLine("Roll: " + A.Roll);
            Console.WriteLine("Name: " + A.Name);
        }
    }
}
```

**Output:**

```
Roll: 6  
Name: Aniket  
  
Process finished with exit code 0.
```



## 19. Write a program for Abstraction

// A program in C# to understand the concepts of Abstraction and abstract classes

```
using System;
```

```
namespace OOPs
```

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

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

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

        private class BSc:Cs
        {
            public override void Fun()
            {
                Console.WriteLine("BSc IT");
            }
        }

        public class MyClass
        {
            public static void Main()
            {
                Cs c;
                c = new Bca();
            }
        }
    }
}
```

```

        c.Fun();
        c = new Mca();
        c.Fun();
        c = new BSc();
        c.Fun();
    }
}
}

```

## Output:

```

Bca
MCA
BSc IT

Process finished with exit code 0.

```

## 20. Write a program for single Inheritance

// A program to understand the concepts of multiple inheritance

```
using System;
namespace OOPs
{
    public class Inheritance
    {
        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(); //creating object
                //accessing methods of parent classes
                s.Print();
                s.Print1();
            }
        }
    }
}
```

**Output:**

```
Print
Print 2

Process finished with exit code 0.
```

## 21. Write a program for Multilevel Inheritance

```
// A program to understand the concepts of multiple inheritance
using System;
namespace OOPs
{
    public class MultiInheritance
    {
        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(); //creating object
                //accessing methods of parent classes
                s.Print();
                s.Print1();
                s.Print2();
            }
        }
    }
}
```

**Output:**

```
Print
Print 1
Print 2

Process finished with exit code 0.
```

## 22. Write a program for multiple Inheritance

```
// A program to understand the concepts of multiple inheritance
using System;
namespace OOPs
{
    public class MultipleInheritance
    {
        class MainClass
        {
            public void Print()
            {
                Console.WriteLine("Print");
            }
        }
        interface MainClass1
        {
            public 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(); //creating object
                //accessing methods of parent classes
                s.Print();
                s.Print1();
                s.Print2();
            }
        }
    }
}
```

**Output:**

```
Print
Print 1
Print 2

Process finished with exit code 0.
```

## 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(23, 34);
            double sum2 = Sum(43.54, 43.6);
            Console.WriteLine(sum1);
            Console.WriteLine(sum2);
        }
    }
}
```

### Output:

```
57
87.14

Process finished with exit code 0.
```

## 24. Write a program for method overriding

```
using System;
namespace OOPs
{
    public class MethodOverriding
    {
        public class Cs
        {
            public virtual void Fun()
            {
                Console.WriteLine("Bca");
            }
        }
        public class Mca:Cs
        {
            public override void Fun()
            {
                Console.WriteLine("MCA");
            }
        }
        private class Bca:Cs
        {
            public override void Fun()
            {
                Console.WriteLine("BCA");
            }
        }
        public static void Main()
        {
            Cs c;
            c = new Bca();
            c.Fun();
            c = new Mca();
            c.Fun();
        }
    }
}
```

**Output:**

```
BCA
MCA
```

```
Process finished with exit code 0.
```

## 25. Write a program for Interface

// A program in C# to understand the concepts of 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();
            }
        }
    }
}
```

**Output:**

```
BCA
MCA
```

```
Process finished with exit code 0.
```



## 26. Write a program for Namespace

//A program to understand the concept of namespace in C#

```
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("Hello");
        }
    }
}
```

### Output:

A screenshot of a console window with a light blue header bar. The text "Hello" is displayed in black font on a white background.A screenshot of a console window showing the message "Process finished with exit code 0." in blue font on a white background.

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

```
using System;
namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] myNumbers = {1, 2, 3};
            try
            {
                Console.WriteLine(myNumbers[5]);
            }
            catch (Exception e)
            {
                Console.WriteLine(e.Message);
            }
        }
    }
}
```

### Output:

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

Process finished with exit code 0.
```

## 28. Write a program for Properties

// A program to understand the properties in C# using get and set.

```
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,
"Abcd", "good");
        Properties.CSharp c2 = new Properties.CSharp(2,
"Efgh", "2");
        Console.WriteLine(c1.roll+" " + c1.name+ " " +
c1.Result1);
        Console.WriteLine(c2.roll+" " + c2.name+ " " +
c2.Result1);
    }
}
```

**Output:**

```
1 Abcd good
2 Efgh Not Valid

Process finished with exit code 0.
```

## 29. Write a program for Constructors

```
// A program in C# to understand the concepts of interface

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(6,
            "Aniket", 8);
            Console.WriteLine("Roll: "+ a.roll+"\nName:
            "+a.name+"\nMarks: "+a.marks);
        }
    }
}
```

### Output:

```
Roll: 6
Name: Aniket
Marks: 8

Process finished with exit code 0.
```

### 30. Write a program for Threading

```
using System;
using System.Threading;

namespace Threading
{
    public 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();
        }
    }
}
```

#### Output:

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

Process finished with exit code 0.
```

### 31. Write a program for 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] = "Abhijeet";
        ic[1] = "Aman";
        ic[2] = "Amit";
        ic[3] = "Aniket";

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

#### Output:

```
Printing values stored in objects used as arrays
Abhijeet
Aman
Amit
Aniket

Process finished with exit code 0.
```

### 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();
        }
    }
}
```

#### Output:

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

Process finished with exit code 0.
```