**C # Programs.**

**Name:-** Bhupesh Pravin Patil.

**Roll:-** 68. **Div:-** B.

1. **Write a program to print “Teach One, Each One, Tree One” given number of times.**

=>

// Bhupesh Pravin Patil.

Using System;

Namespace HelloWorld

{

Class Program

{

Static void Main(string[] args)

{

Console.WriteLine(“Enter The Word “);

String name=Console.ReadLine();

Console.WriteLine(“How Many Times”);

Int n=Convert.ToInt32(Console.ReadLine());

For(int i=1;i<=n;i++)

{

Console.WriteLine(“\n”+name);

}

}

}

}

**Output:-**

Enter The Word

Teach one, each one, Tree one

How Many Times

4

Teach one, each one, Tree one

Teach one, each one, Tree one

Teach one, each one, Tree one

Teach one, each one, Tree one

**2. Write a program to show use of different operators.**

=>

// Bhupesh Pravin Patil

Using System;

Namespace HelloWorld

{

Class Program

{

Static void Main(string[] args)

{

Console.WriteLine(“Enter 1st no”);

Int n1=int.Parse(Console.ReadLine());

Console.WriteLine(“Enter 2nd no”);

Int n2=int.Parse(Console.ReadLine());

Int sum=n1+n2;

Console.WriteLine(“The Arithmetic operator The Sum is :- “+sum);

// Comparison Operator

Console.WriteLine(“n1 == n2 “+(n1==n2));

Console.WriteLine(“n1 != n2 “+(n1!=n2));

// Assignment Operator

Console.WriteLine(“n1 /= n2 “+(n1+=n2));

// Logical operator

Console.WriteLine(“n1 != n2 && n1 == 2 “+(n1 != n2 && n1 == 2));

Console.ReadLine();

}

}

}

**Output:-**

Enter 1st no

3

Enter 2nd no

5

The Arithmetic operator The Sum is :- 8

N1 == n2 False

N1 != n2 True

N1 /= n2 8

N1 != n2 && n1 == 2 False

**3. Write a program to show use of Looping Constructs.**

=>

**1. For loop**

**:->**

// Bhupesh Pravin Patil

using System;

namespace HelloWorld

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter The Number");

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

Console.WriteLine("The Given Number of Table");

for (int i=1;i<=10;i++)

{

Console.WriteLine(" "+(n\*i));

}

Console.ReadLine();

}

}

}

**Output:-**

Enter The Number

3

The Given Number of Table

3

6

9

12

15

18

21

24

27

30

1. **While loop**

**:->**

// Bhupesh Pravin Patil

Using System;

Namespace HelloWorld

{

Class Program

{

Static void Main(string[] args)

{

Int i=1;

Int fact =1;

Console.WriteLine(“Enter The Number”);

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

Console.WriteLine(“The Factorial of Given Number”);

While(i<=n)

{

Fact=fact\*i;

I++;

}

Console.WriteLine(“ “+fact);

Console.ReadLine();

}

}

}

**Output:-**

Enter The Number

5

The Factorial of Given Number

120

1. **Do-While loop**

**:->**

// Bhupesh Pravin Patil

using System;

namespace CSharp\_Shell

{

class Program

{

public static void Main()

{

int i = 1; int n;

Console.WriteLine("Enter The Number:- ");

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

Console.WriteLine("The Table of Given Number is: ");

do

{

Console.WriteLine(" " + (n \* i));

i++;

}

while (i <= 10);

Console.ReadLine();

}

}

}

**Output:-**

Enter The Number

3

The Given Number of Table

3

6

9

12

15

18

21

24

27

30

**4. Write a program to show use of Constructor.**

=>

// Bhupesh Pravin Patil

Using System;

Namespace HelloWorld

{

Class Program

{

Int Roll;

String name;

Program (int r,String n)

{

Roll=r;

Name=n;

Console.WriteLine(“Roll No. Is :- “+r+”\nName is :-“+n);

}

Static void Main(string[] args)

{

Program p=new Program (68,”Bhupesh”);

}

}

}

**Output:-**

Roll No. Is :- 68

Name is :-Bhupesh

**5. Write a program to demonstrate Inheritance.**

=>

**1. Single inheritance.**

=>

// Bhupesh Pravin Patil

Using System;

Namespace CSharp\_Shell

{

Class Baseclass

{

Public void show1()

{

Console.WriteLine(“This Is Base class Method”);

}

}

Class Derivedclass : Baseclass

{

Public void show2()

{

Console.WriteLine(“This Is Derived class Method”);

}

}

Class Program

{

Public static void Main()

{

Derivedclass d = new Derivedclass();

d.show1();

d.show2();

Console.ReadLine();

}

}

}

**Output:-**

This Is Base class Method.

This Is Derived class Method.

**2. Multilevel inheritance.**

=>

// Bhupesh Pravin Patil

Using System;

Namespace CSharp\_Shell

{

Class Baseclass

{

Public void baseclass()

{

Console.WriteLine(“This is Base class”);

}

}

Class Derivedclass1 : Baseclass

{

Public void derivedclass1()

{

Console.WriteLine(“This is Derived from Base class”);

}

}

Class Derivedclass2 : Derivedclass1

{

Public void derivedclass2()

{

Console.WriteLine(“This is Derived from Derived class1”);

}

}

Class Program

{

Public static void Main()

{

Derivedclass2 d = new Derivedclass2();

d.baseclass();

d.derivedclass1();

d.derivedclass2();

Console.ReadLine();

}

}

}

**Output:-**

This is Base class.

This is Derived from Base class.

“This is Derived from Derived class1.

**3. Hierarchical inheritance.**

=>

Using System;

Namespace CSharp\_Shell

{

Class Baseclass

{

Public void baseclass()

{

Console.WriteLine(“This is Method of Base class”);

}

}

Class Derivedclass1 : Baseclass

{

Public void derivedclass1()

{

Console.WriteLine(“This is The Method of 1st Derived class”);

}

}

Class Derivedclass2 : Baseclass

{

Public void derivedclass2()

{

Console.WriteLine(“This is The Method of 2nd Derived class”);

}

}

Class Program

{

Public static void Main()

{

/\*Baseclass b = new Baseclass();

b.baseclass();\*/

Derivedclass1 d = new Derivedclass1();

d.baseclass();

d.derivedclass1();

Derivedclass2 d2 = new Derivedclass2();

D2.derivedclass2();

D2.baseclass();

Console.ReadLine();

}

}

}

**Output:-**

This is Method of Base class.

This is The Method of 1st Derived class.

This is The Method of 2nd Derived class.

**4. Multiple inheritance.**

=>

Using System;

Namespace CSharp\_Shell

{

Class Person

{

Public void show1()

{

Console.WriteLine(“This is Method of Person class”);

}

}

Interface Employee

{

Void show2();

}

Class Teacher : Person, Employee

{

Public void show2()

{

Console.WriteLine(“This is Method of Employee interface”);

}

}

Class Program

{

Public static void Main()

{

Teacher t = new Teacher();

t.show1();

t.show2();

Console.WriteLine();

Console.ReadLine();

}

}

}

**Output:-**

This is Method of Person class.

This is Method of Employee interface.