Implementation of Inheritance in C#

using System;

namespace Tutlane

{

public class User

{

public string Name;

private string Location;

public User()

{

Console.WriteLine("Base Class Constructor");

}

public void GetUserInfo(string loc)

{

Location = loc;

Console.WriteLine("Name: {0}", Name);

Console.WriteLine("Location: {0}", Location);

}

}

public class Details: User

{

public int Age;

public Details()

{

Console.WriteLine("Child Class Constructor");

}

public void GetAge()

{

Console.WriteLine("Age: {0}", Age);

}

}

class Program

{

static void Main(string[] args)

{

Details d = new Details();

d.Name = "Suresh Dasari";

// Compile Time Error

//d.Location = "Hyderabad";

d.Age = 32;

d.GetUserInfo("Hyderabad");

d.GetAge();

Console.WriteLine("\nPress Any Key to Exit..");

Console.ReadLine();

} }}

# // C# program to find largest of three numbers

using System;

using System.IO;

using System.Text;

namespace IncludeHelp

{

class Test

{

// Main Method

static void Main(string[] args)

{

int a;

int b;

int c;

int large;

//input the numbers

Console.Write("Enter first number : ");

a = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter second number: ");

b = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter third number : ");

c = Convert.ToInt32(Console.ReadLine());

//finding largest number using if-else

if (a > b && a > c)

large = a;

else if (b > a && b > c)

large = b;

else large = c;

//printing

Console.WriteLine("Using if-else...");

Console.WriteLine("Largest number is {0}", large);

//finding largest number using ternary operator

large = (a > b && a > c) ? a : (b > a && b > c) ? b : c;

//printing

Console.WriteLine("Using ternary operator...");

Console.WriteLine("Largest number is {0}", large);

//hit ENTER to exit the program

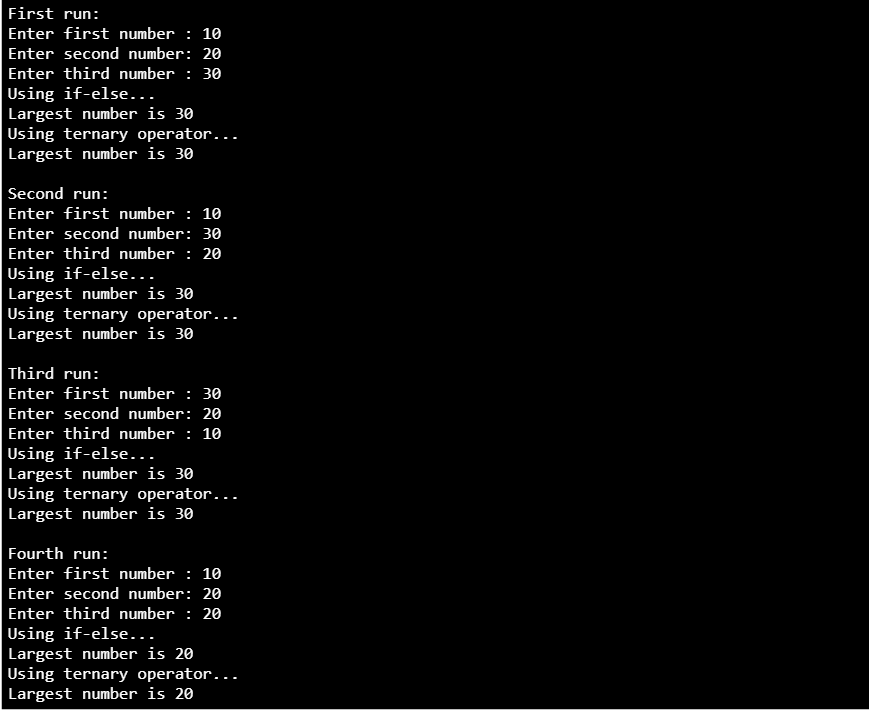
Console.ReadLine();

}

}

}

**OUTPUT**



# Implementation of Multidimensional Arrays in C#

using System;

namespace Vedas

{

class Program

{

static void Main(string[] args)

{

// Two Dimensional Array

int[,] array2D = new int[3, 2] { { 4, 5 }, { 5, 0 }, { 3, 1 } };

// Three Dimensional Array

int[, ,] array3D = new int[2, 2, 3] { { { 1, 2, 3 }, { 4, 5, 6 } }, { { 7, 8, 9 }, { 10, 11, 12 } } };

Console.WriteLine("---Two Dimensional Array Elements---");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 2; j++)

{

Console.WriteLine("a[{0},{1}] = {2}", i, j, array2D[i, j]);

}

}

Console.WriteLine("---Three Dimensional Array Elements---");

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 2; j++)

{

for (int k = 0; k < 3; k++)

{

Console.WriteLine("a[{0},{1},{2}] = {3}", i, j, k, array3D[i, j, k]);

}

}

}

Console.WriteLine("Press Enter Key to Exit..");

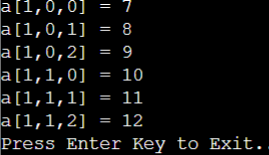
Console.ReadLine();

}

}

}

**OUTPUT**



# Implementation of LINQ in C#

# using System;

using System.Linq;

using System.Collections.Generic;

public class Program

{

public static void Main()

{

// Student collection

IList<Student> studentList = new List<Student>() {

new Student() { StudentID = 1, StudentName = "John", Age = 13} ,

new Student() { StudentID = 2, StudentName = "Moin", Age = 21 } ,

new Student() { StudentID = 3, StudentName = "Bill", Age = 18 } ,

new Student() { StudentID = 4, StudentName = "Ram" , Age = 20} ,

new Student() { StudentID = 5, StudentName = "Ron" , Age = 15 }

};

// LINQ Query Syntax to find out teenager students

var teenAgerStudent = from s in studentList

where s.Age > 12 && s.Age < 20

select s;

Console.WriteLine("Teen age Students:");

foreach(Student std in teenAgerStudent){

Console.WriteLine(std.StudentName);

}

}

}

public class Student{

public int StudentID { get; set; }

public string StudentName { get; set; }

public int Age { get; set; }

}

# Implementation of File Handling in C#

using System;

using System.Text;

using System.IO;

namespace FileWriting\_SW

{

class Program

{

class FileWrite

{

public void WriteData ()

{

FileStream fs =

new FileStream ("c:\\test.txt", FileMode.Append, FileAccess.Write);

StreamWriter sw = new StreamWriter (fs);

Console.WriteLine

("Enter the text which you want to write to the file");

string str = Console.ReadLine ();

sw.WriteLine (str);

sw.Flush ();

sw.Close ();

fs.Close ();

}

}

static void Main (string[]args)

{

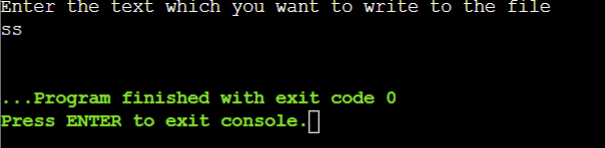
FileWrite wr = new FileWrite ();

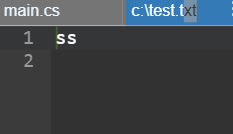
wr.WriteData ();

}

}

}

**OUTPUT**



# Implementation of Collection in C#

// C# program to illustrate the concept

// of generic collection using List<T>

using System;

using System.Collections.Generic;

class Vedas {

// Main Method

public static void Main(String[] args)

{

// Creating a List of integers

List<int> mylist = new List<int>();

// adding items in mylist

for (int j = 5; j < 10; j++) {

mylist.Add(j \* 3);

}

// Displaying items of mylist

// by using foreach loop

foreach(int items in mylist)

{

Console.WriteLine(items);

}

}

}

# Implementation of Polymorphism in C#

using System;

public class Shape{

public virtual void draw(){

Console.WriteLine("drawing...");

}

}

public class Rectangle: Shape

{

public override void draw()

{

Console.WriteLine("drawing rectangle...");

}

}

public class Circle : Shape

{

public override void draw()

{

Console.WriteLine("drawing circle...");

}

}

public class TestPolymorphism

{

public static void Main()

{

Shape s;

s = new Shape();

s.draw();

s = new Rectangle();

s.draw();

s = new Circle();

s.draw();

}

}

**OUTPUT**

