***C# Assignment***

1. namespace basicprgm1

{

internal abstract class nothing

{

public abstract void Name(string name,int roll);

}

}

namespace basicprgm1

{

internal class new\_nothing : nothing

{

public override void Name(string name,int roll)

{

string sname,address;

int rollno;

sname= name;

rollno = roll;

address = "Mettupalayam";

Console.WriteLine("Enter Student Name "+sname);

Console.WriteLine("Rollno " + rollno);

Console.WriteLine("Address " + address);

}

}

}

new\_nothing n= new new\_nothing();

n.Name("Boo", 9);

1. namespace basicprgm1

{

internal interface IProject

{

public void Progress();

}

}

namespace basicprgm1

{

internal class UIProject : IProject

{

public void Progress()

{

Console.WriteLine("Completed UI");

}

}

}

namespace basicprgm1

{

internal class DBIProject : IProject

{

public void Progress()

{

Console.WriteLine("Completed DB Design");

}

}

}

int choice;

Console.WriteLine("Enter your choice");

Console.WriteLine("1.UIProject");

Console.WriteLine("2.DBProject");

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

switch (choice) {

case 1:

UIProject ui = new UIProject();

ui.Progress();

break;

case 2:

DBIProject dbi = new DBIProject();

dbi.Progress();

break;

default:

Console.WriteLine("Enter your correct choice");

break;

}

1. namespace basicprgm1

{

internal class multipleclass

{

public void display()

{

Console.WriteLine("Base Class 1");

}

}

}

namespace basicprgm1

{

internal interface multipleinterface

{

public void display1();

}

}

namespace basicprgm1

{

internal class derivedclass : multipleclass,multipleinterface

{

public derivedclass() : base()

{ }

public void derived() {

Console.WriteLine("Derived Class");

}

public void display1()

{

Console.WriteLine("Base class 2");

}

}

}

derivedclass d=new derivedclass();

d.display();

1. Implementation of Hierarchical Inheritance with Loan Acc and Save Acc

namespace Assign3

{

internal class Program

{

public static void Main(string[] args)

{

Console.Write("Enter Name: ");

string name=Console.ReadLine();

Console.Write("Enter Email: ");

string email=Console.ReadLine();

Console.Write("Enter Phone Number: ");

long phone=Convert.ToInt64(Console.ReadLine());

Console.Write("Enter City Name: ");

string city=Console.ReadLine();

Console.Write("Enter Loan Acc Number: ");

int LoanAcc=Convert.ToInt32(Console.ReadLine());

Console.Write("Enter LoanAmt: ");

int LoanAmt=Convert.ToInt32(Console.ReadLine());

Console.Write("Enter LoanTen: ");

int LoanTen=Convert.ToInt32(Console.ReadLine());

Console.Write("Enter AccNo: ");

int Acc = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Amt: ");

int Amt = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Branch Name: ");

string BranchName = Console.ReadLine();

Console.Write("Enter IFSC Code: ");

string IFSC = Console.ReadLine();

Console.WriteLine("Select 1 => LoanAcc and 2 => SaveAcc");

int choice=Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 1:

LoanAcc loanAcc = new LoanAcc(name, email, phone, city, LoanAcc, LoanAmt, LoanTen);

loanAcc.display();

break;

case 2:

SaveAcc saveAcc = new SaveAcc(name, email, phone, city, Acc, Amt, BranchName, IFSC);

saveAcc.display();

break;

}

}

}

}

namespace Pract

{

internal class Customer

{

private string name;

private string email;

private long phone;

private string city;

public Customer(string name, string email, long phone, string city)

{

this.Name = name;

this.Email = email;

this.Phone = phone;

this.City = city;

}

public string Name { get => name; set => name = value; }

public string Email { get => email; set => email = value; }

public long Phone { get => phone; set => phone = value; }

public string City { get => city; set => city = value; }

public void display()

{

Console.WriteLine("Name: "+this.Name);

Console.WriteLine("Email: "+this.Email);

Console.WriteLine("Phone: " + this.Phone);

Console.WriteLine("City: " + this.City);

}

}

internal class LoanAcc:Customer

{

private int LoanAccNo;

private int LoanAmt;

private int LoanTenury;

public LoanAcc(string name,string email,long phone, string city,int loanAccNo, int loanAmt, int loanTenury):base( name,email, phone, city)

{

LoanAccNo = loanAccNo;

LoanAmt = loanAmt;

LoanTenury = loanTenury;

}

public int LoanAccNo1 { get => LoanAccNo; set => LoanAccNo = value; }

public int LoanAmt1 { get => LoanAmt; set => LoanAmt = value; }

public int LoanTenury1 { get => LoanTenury; set => LoanTenury = value; }

public void display()

{

base.display() ;

Console.WriteLine("Loanacc:" + LoanAccNo);

Console.WriteLine("LoanAmt:"+LoanAmt);

Console.WriteLine("LoanTen: " + LoanTenury);

}

}

internal class SaveAcc:Customer

{

private int AccNo;

private int Amt;

private string BranchName;

private string IFSC;

public SaveAcc(string name, string email, long phone, string city, int accNo, int amt, string branchName, string iFSC) : base(name, email, phone, city)

{

AccNo1 = accNo;

Amt1 = amt;

BranchName1 = branchName;

IFSC1 = iFSC;

}

public int AccNo1 { get => AccNo; set => AccNo = value; }

public int Amt1 { get => Amt; set => Amt = value; }

public string BranchName1 { get => BranchName; set => BranchName = value; }

public string IFSC1 { get => IFSC; set => IFSC = value; }

public void display()

{

base.display() ;

Console.WriteLine("Acc:" + AccNo);

Console.WriteLine("Amt:" + Amt);

Console.WriteLine("BranchName: " + BranchName);

Console.WriteLine("IFSC: " + IFSC);

}}

}

1. Jagged Array :

public void jagged()

{

int i, j;

int[][] jaggedArray = new int[4][];

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

jaggedArray[1] = new int[] { 40, 50, 11, 4 };

jaggedArray[2] = new int[] { 55, 17 };

jaggedArray[3] = new int[4];

Console.WriteLine(jaggedArray[0][2]);

for(i = 0; i < jaggedArray[3].Length; i++)

{

jaggedArray[3][i]=Convert.ToInt32( Console.ReadLine());

}

int[][,] jaggy=new int[3][,];

jaggy[0] = new int[5,4];

jaggy[1]= new int[6,5];

jaggy[2]= new int[7,6];

for( i = 0; i < 5;i++)

{

for( j = 0; j < 4; j++)

{

jaggy[0][i,j] =Convert.ToInt32(Console.ReadLine());

}

}

for (i = 0; i < 5; i++)

{

for (j = 0; j < 4; j++)

{

Console.Write(jaggy[0][i, j]);

}

Console.WriteLine();

}

}

1. namespace Assignment

{

internal class Program

{

public static void Main(string[] args)

{

Console.Write("Enter limit: ");

int limit=Convert.ToInt32(Console.ReadLine());

int[] array1 = new int[limit];

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

{

array1[i] = Convert.ToInt32(Console.ReadLine());

}

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

{

for(int j = i+1; j < limit; j++)

{

if (array1[i] > array1[j])

{

int temp = array1[i];

array1[i] = array1[j];

array1[j] = temp;

}

}

}

Console.WriteLine("--------------------------------");

Console.WriteLine("Ascending Order Sort is displayed Below");

Console.Write(String.Join(",", array1));

}

}

}

1. namespace Assignment

{

internal class Array

{

private int id;

private string name;

private int age;

private string addr;

public Array(int id,string name, int age, string addr)

{

this.id = id;

this.name = name;

this.age = age;

this.addr = addr;

}

public int Id { get => id; set => id = value; }

public string Name { get => name; set => name = value; }

public int Age { get => age; set => age = value; }

public string Addr { get => addr; set => addr = value; }

public void display()

{

Console.WriteLine(this.id);

Console.WriteLine(this.name);

Console.WriteLine(this.age);

Console.WriteLine(this.addr);

}

}

internal class Program

{

public static void Main(string[] args)

{

Console.Write("Enter id: ");

int id=Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Name: ");

string name=Console.ReadLine();

Console.Write("Enter Age: ");

int age=Convert.ToInt32(Console.ReadLine());

Console.Write("Enter addr: ");

string addr=Console.ReadLine();

Console.WriteLine("------------------------------------------------");

Array array = new Array(id, name, age, addr);

array.display();

}

}

}

1. internal class Setter

{

private int id;

private string name;

private int age;

private string addr;

public Setter(int id,string name, int age, string addr)

{

this.id = id;

this.name = name;

this.age = age;

this.addr = addr;

}

public int Id { get => id; set => id = value; }

public string Name { get => name; set => name = value; }

public int Age { get => age; set => age = value; }

public string Addr { get => addr; set => addr = value; }

public void display()

{

Console.WriteLine(this.id);

Console.WriteLine(this.name);

Console.WriteLine(this.age);

Console.WriteLine(this.addr);

}

}