TAAZAA TRAINING – ASSIGNMENT(3)

BY DAS SUKHDEV

**INTERFACE CONCEPT**

using System;

namespace interfaceconcept

{

interface Ifirst

{

void fun1();

}

class first:Ifirst

{

public void fun1()

{

System.Console.WriteLine("Hey I'm from interface1!!");

}

}

class Program

{

static void Main(string[] args)

{

var obj=new first();

Ifirst obj1=new first();

obj.fun1();

obj1.fun1();

}}}

**BOXING AND UNBOXING CONCEPT**

using System;

namespace boxing\_unboxing

{

class Program

{

//process of converting value type to reference type is called boxing

//process of converting reference type to value type is called unboxing

static void Main(string[] args)

{

int i=100;

object obj=i;//boxing

int x=(int)obj;//unboxing

System.Console.WriteLine(i);

System.Console.WriteLine(obj);

System.Console.WriteLine(x);

}

}

}

**VALUE AND REFERENCE TYPE PARAMETERS**

using System;

namespace value\_reference

{

class Program

{

public string name;

public int id;

public void change(int i)

{

i=200;

}

static void SomeFunction(ref int x)

{

x=x+50;

}

static void Main(string[] args)

{

int i=100;

var obj=new Program();

obj.change(i);//value type

System.Console.WriteLine(i);

//int j; we must initialize the value otherwise it will create an error

int j=20;

SomeFunction(ref j);

System.Console.WriteLine(j);

}

}}

**TWO-DIMENSION ARRAY**

using System;

namespace \_2\_d

{

class Program

{

static void Main(string[] args)

{

int[,] twodim={

{1,2},

{3,4},

{5,6}

};

System.Console.WriteLine(twodim[0,1]);

System.Console.WriteLine(twodim[1,1]);

}

}

}

**JAGGED ARRAY**

using System;

namespace jagged

{

class Program

{

static void Main(string[] args)

{

int [][] jarray=new int[2][];

jarray[0]=new int[3];

jarray[1]=new int[4];

jarray[0][0]=10;

jarray[0][1]=20;

jarray[0][2]=30;

jarray[1][0]=40;

jarray[1][1]=50;

jarray[1][2]=60;

jarray[1][3]=70;

for(int i=0;i<jarray.GetLength(0);i++)

{

Console.Write("\n");

for(int j=0;j<jarray[i].GetLength(0);j++)

{

System.Console.Write("{0} ",jarray[i][j]);

}

}

Console.ReadKey();

}

}

}

**ABSTRACT CLASS IMPLEMENTATION**

using System;

namespace abstractimplementation

{

public abstract class Figure

{

public double width,height,radius;

public const float pi=3.14f;

public abstract double getArea();

}

public class Rectangle:Figure{

public Rectangle(double width,double height)

{

this.width=width;

this.height=height;

}

public override double getArea()

{

return width\*height;

}

}

public class Circle:Figure{

public Circle(double radius)

{

this.radius=radius;

}

public override double getArea()

{

return pi\*radius\*radius;

}

}

class TestFigure

{

static void Main()

{

var obj=new Rectangle(10.45,20.54);

var obj1=new Circle(12.5);

System.Console.WriteLine("Area of Rectangle is"+obj.getArea());

System.Console.WriteLine("Area of Circle is"+obj1.getArea());

}

}

}