**PRACTICAL NO.11**

**Write the program for the following:**

**A)Create a simple application to demonstrate the concepts boxing and unboxing.**

**Code:**

using System;

namespace BoxingUnboxingDemo

{

class Program

{

static void Main(string[] args)

{

int valueType = 123;

object boxedValue = valueType;

Console.WriteLine($"Boxed Value: {boxedValue}");

int unboxedValue = (int)boxedValue;

Console.WriteLine($"Unboxed Value: {unboxedValue}");

valueType = 456;

Console.WriteLine($"Original Value Type after modification: {valueType}");

Console.WriteLine($"Boxed Value after valueType modification: {boxedValue}");

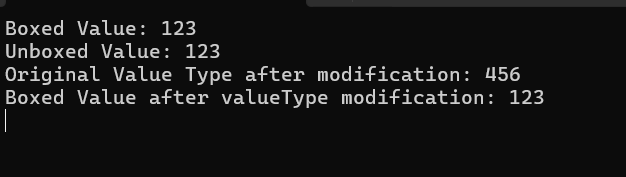
Console.ReadLine();

}

}

}

Output:



**B)Create a simple application to perform addition and subtraction using delegate.**

**Code:**

using System;

namespace DelegateDemo

{

delegate int MathOperation(int x, int y);

class Program

{

static void Main(string[] args)

{

int Add(int x, int y)

{

return x + y;

}

int Subtract(int x, int y)

{

return x - y;

}

MathOperation addOperation = new MathOperation(Add);

MathOperation subtractOperation = new MathOperation(Subtract);

int a = 10;

int b = 5;

int additionResult = addOperation(a, b);

int subtractionResult = subtractOperation(a, b);

Console.WriteLine($"Addition of {a} and {b} is: {additionResult}");

Console.WriteLine($"Subtraction of {a} and {b} is: {subtractionResult}");

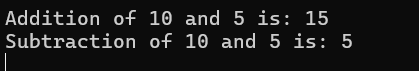
Console.ReadLine();

}

}

}

Output:



**C) Create a simple application to demonstrate use of the concepts of interfaces.**

**Code:**

using System;

namespace InterfaceDemo

{

public interface Shape

{

double GetArea();

double GetPerimeter();

}

public class Rectangle : Shape

{

public double Width;

public double Height;

public Rectangle(double width, double height)

{

Width = width;

Height = height;

}

public double GetArea()

{

return Width \* Height;

}

public double GetPerimeter()

{

return 2 \* (Width + Height);

}

}

public class Circle : Shape

{

public double Radius;

public Circle(double radius)

{

Radius = radius;

}

public double GetArea()

{

return Math.PI \* Radius \* Radius;

}

public double GetPerimeter()

{

return 2 \* Math.PI \* Radius;

}

}

class Program

{

static void Main(string[] args)

{

Shape rectangle = new Rectangle(5.0, 4.0);

Shape circle = new Circle(3.0);

Console.WriteLine("Rectangle:");

Console.WriteLine($"Area: {rectangle.GetArea()}");

Console.WriteLine($"Perimeter: {rectangle.GetPerimeter()}");

Console.WriteLine("\nCircle:");

Console.WriteLine($"Area: {circle.GetArea()}");

Console.WriteLine($"Perimeter: {circle.GetPerimeter()}");

Console.ReadLine();

}

}

}

**Output:**

