Lab Assignment 4 in C#

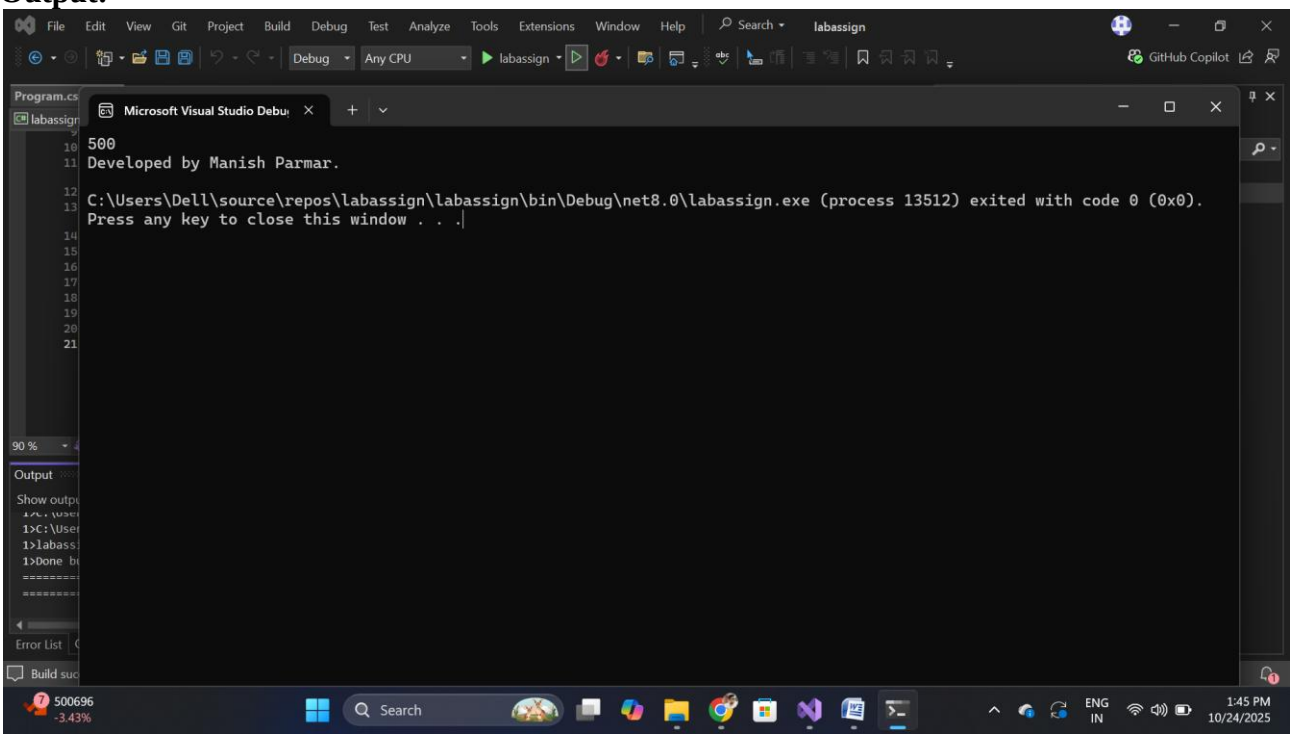
Q 1. Create a class BankAccount with a property Balance.
The property should allow deposit but not allow direct withdrawal (only decrease balance via a method).

Demonstrate depositing 1000 and withdrawing 500 using the property and method.

Ans 1.

```
using System;

class LabAssignment5
{
    public decimal Balance { get; private set; }
    public void Deposit(decimal amount) { Balance += amount; }
    public void Withdraw(decimal amount) { if (amount <= Balance) Balance -= amount;
}
}

class  flrogram
{
    static void Main()
    {
        LabAssignment5 account = new LabAssignment5();
        account.Deposit(1000);
        account.Withdraw(500);
        Console.WriteLine(account.Balance);
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



Q 2. Design a class Student with a property Age.

Ensure that only values between 5 and 25 are allowed.

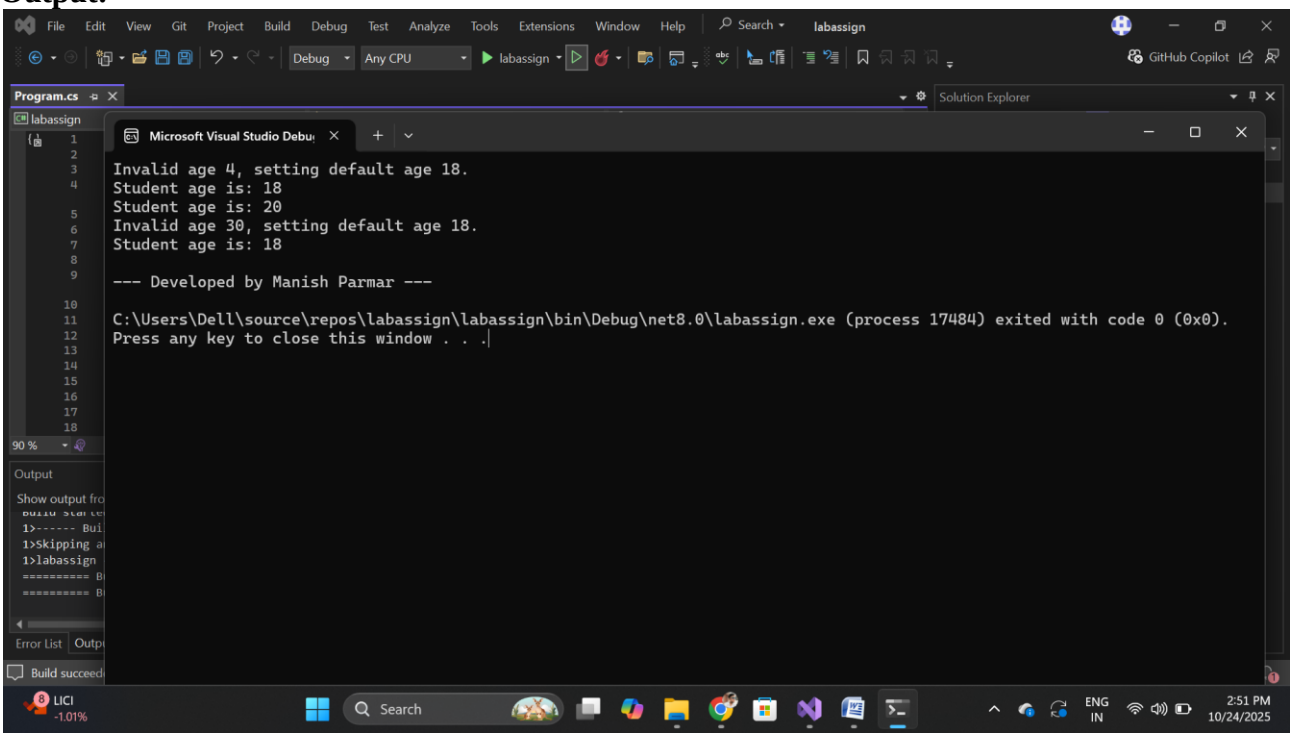
If invalid age is set, default to 18.

Show the behavior for age 4, 20, and 30.

Ans 2.

```
using System;

class LabAssignment5
{
    private int age;
    public int Age
    {
        get { return age; }
        set { age = (value >= 5 fifi value <= 25) o value : 18; }
    }
}

class  flrogram
{
    static void Main()
    {
        LabAssignment5 s1 = new LabAssignment5 { Age = 4 };
        LabAssignment5 s2 = new LabAssignment5 { Age = 20 };
        LabAssignment5 s3 = new LabAssignment5 { Age = 30 };
        Console.WriteLine(s1.Age);
        Console.WriteLine(s2.Age);
        Console.WriteLine(s3.Age);
        Console.WriteLine("Developed by Manish
        Parmar.");
    }
}
```

Output:



Q 3. Create a class Employee with:
A private field basicSalary.

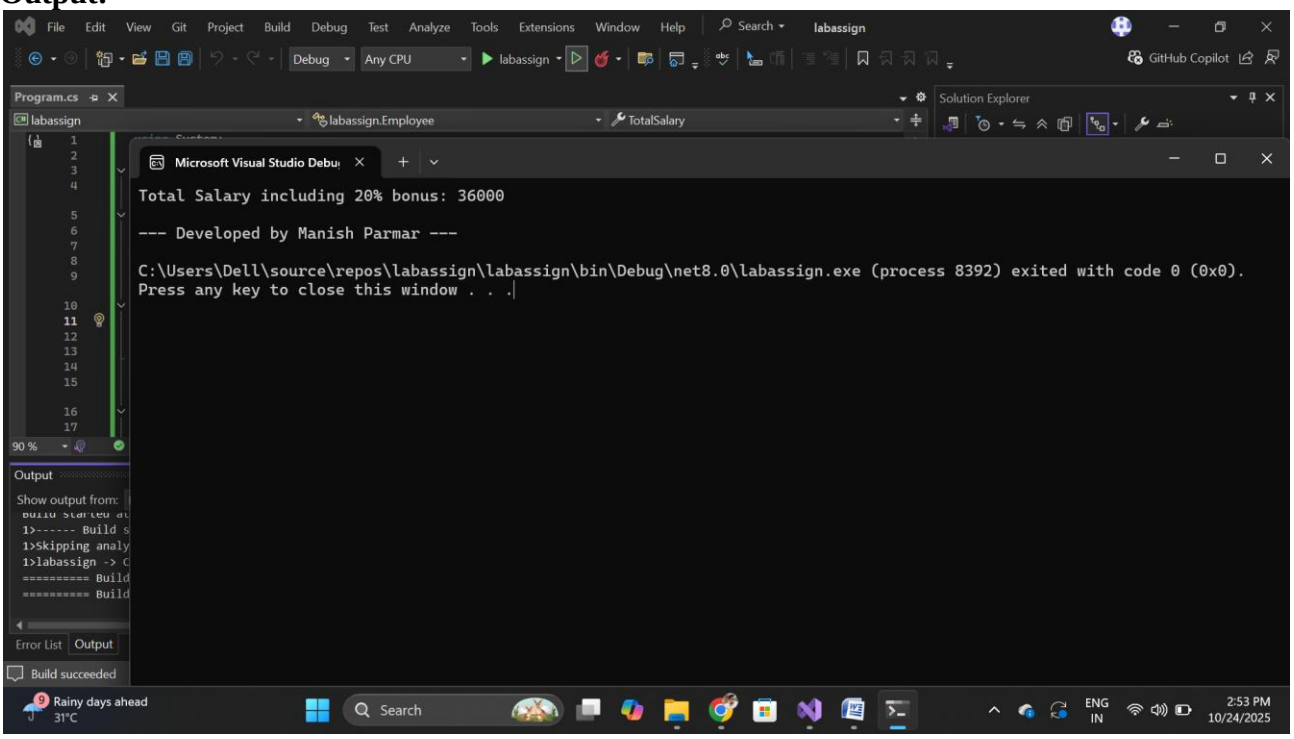
A read-only property TotalSalary that calculates salary with 20% bonus.

Demonstrate setting basicSalary = 30000 and display TotalSalary.

Ans 3.

```
using System;

class LabAssignment5
{
    private decimal basicSalary;
    public decimal TotalSalary { get { return basicSalary + (basicSalary * 0.2m); } }
    public void SetBasicSalary(decimal salary) { basicSalary = salary; }
}

class Program
{
    static void Main()
    {
        LabAssignment5 emp = new LabAssignment5();
        emp.SetBasicSalary(30000);
        Console.WriteLine(emp.TotalSalary);
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



Q 4. Build a class Product with two auto-properties: Price and Discount.
Add a method to calculate the final price = Price – (Price * Discount/100).

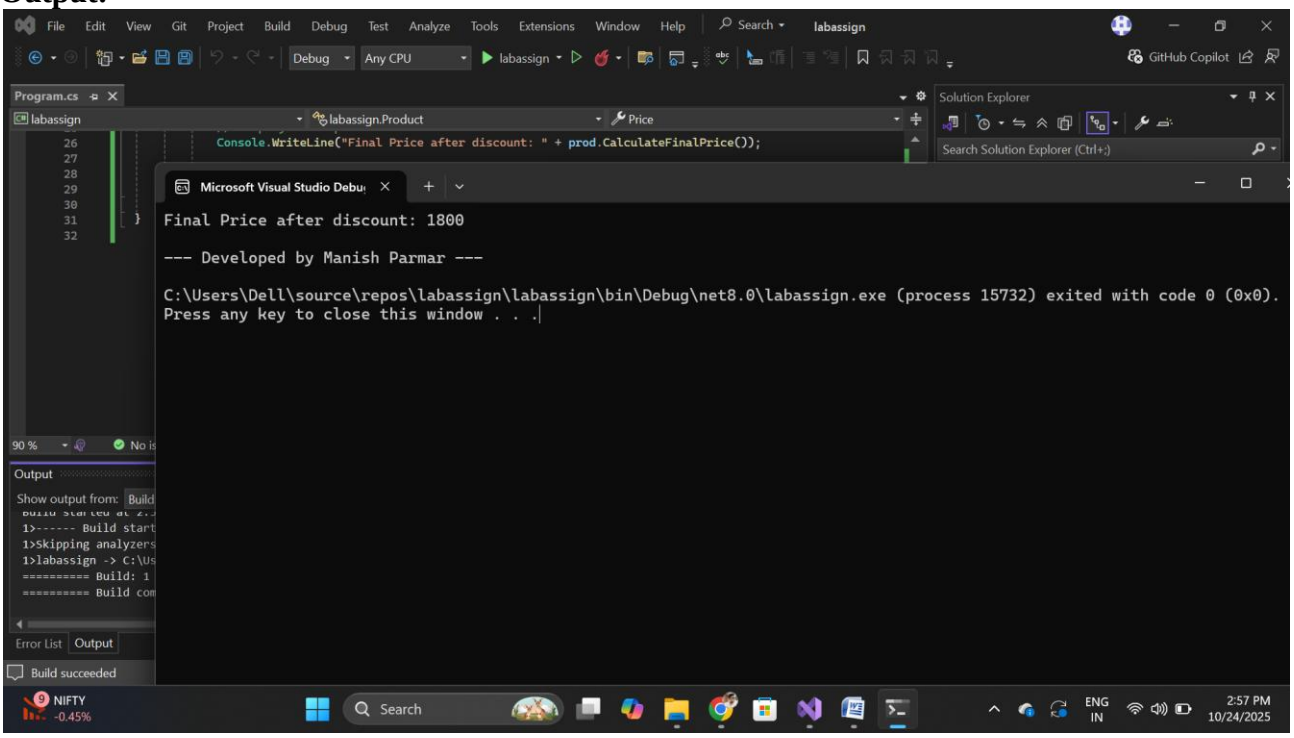
Show result for a product with Price = 2000 and Discount = 10%.

Ans 4.

```
using System;

class LabAssignment5
{
    public decimal Price { get; set; }
    public decimal Discount { get; set; }
    public decimal FinalPrice() { return Price - (Price * Discount / 100); }
}
```

```
class  flrogram
{
    static void Main()
    {
        LabAssignment5 p = new LabAssignment5 { flrice = 2000, Discount = 10 };
        Console.WriteLine(p.Finalflrice());
        Console.WriteLine("Developed  by  Manish Parmar.");
    }
}
```

Output:



Q 5. Create a Car class with a property Speed.
Speed should not exceed 180 km/h; if it exceeds, reset to 180.

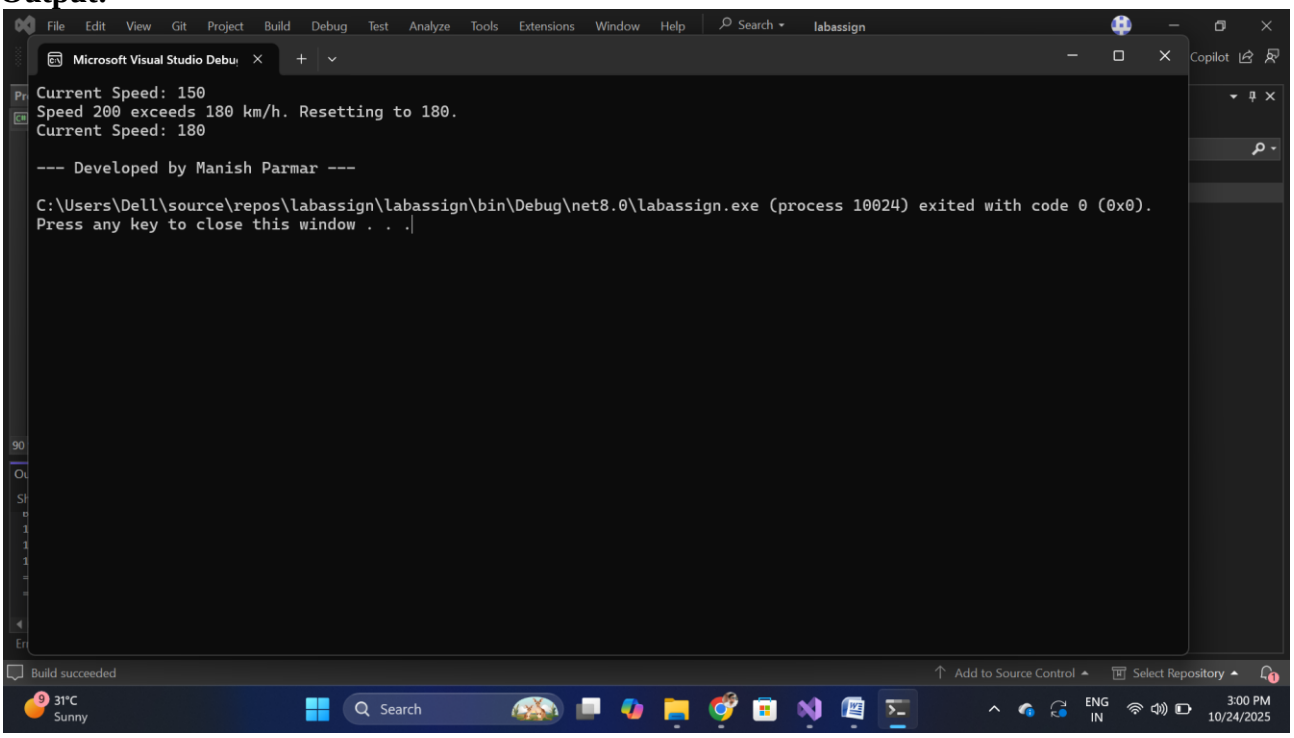
Write code to set speed = 150, then 200, and display the final speed.

Ans 5.

```
using System;

class LabAssignment5
{
    private int speed;
    public int Speed
    {
        get { return speed; }
        set { speed = value > 180 o 180 : value; }
    }
}

class  flrogram
{
    static void Main()
    {
        LabAssignment5  car  =  new  LabAssignment5();
        car.Speed  =  150;
        car.Speed  =  200;
        Console.WriteLine(car.Speed);
```

```
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



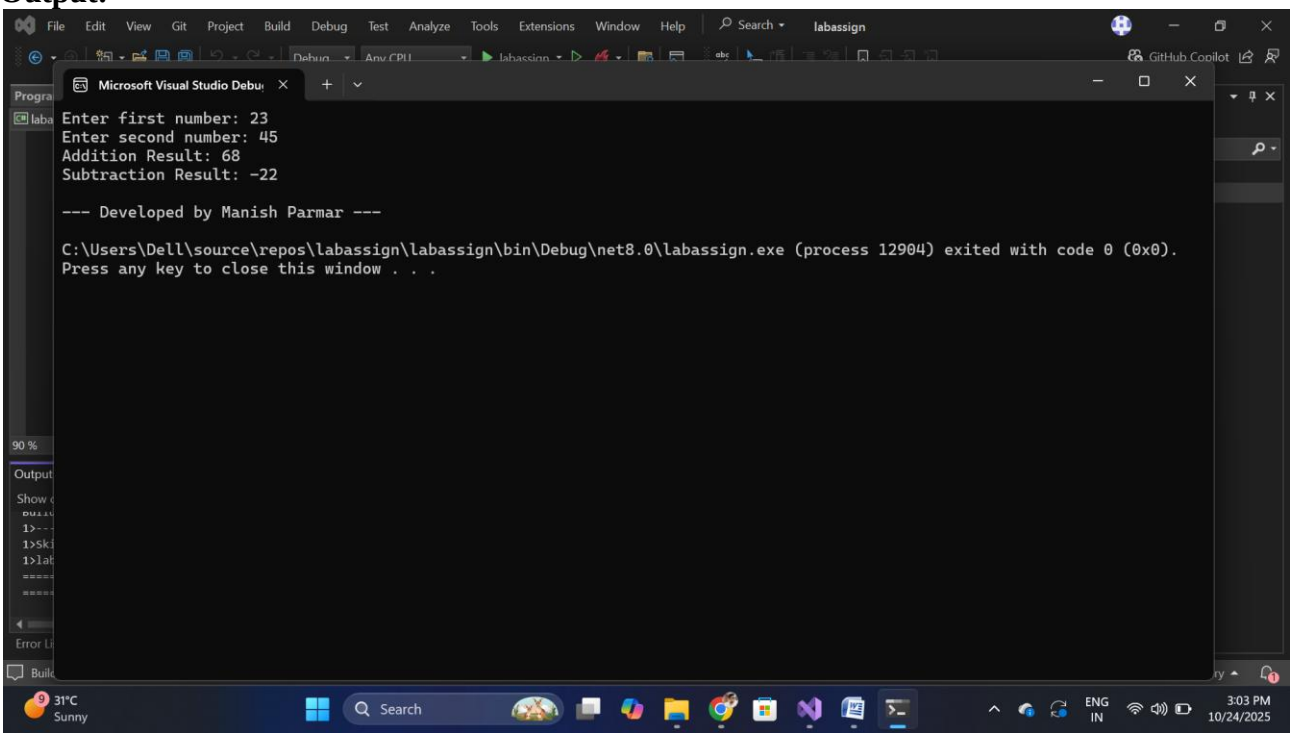
Q 6. Define a delegate Operation for performing arithmetic operations. Implement methods Add and Subtract.

Ask the user for two numbers and apply both operations using the delegate.

Ans 6.

```
using System;

delegate int Operation(int a, int b);

class LabAssignment5
{
    static int Add(int a, int b) { return a + b; }
    static int Subtract(int a, int b) { return a - b; }

    static void Main()
    {
        Console.Write("Enter first number: ");
        int x = int.flarse(Console.ReadLine());
        Console.Write("Enter second number: ");
        int y = int.flarse(Console.ReadLine());

        Operation opAdd = Add;
        Operation opSub = Subtract;

        Console.WriteLine("Addition: " + opAdd(x, y));
        Console.WriteLine("Subtraction: " + opSub(x, y));
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



Q 7. Create a delegate Format Text that accepts a string.
Implement methods to return: The string in uppercase.

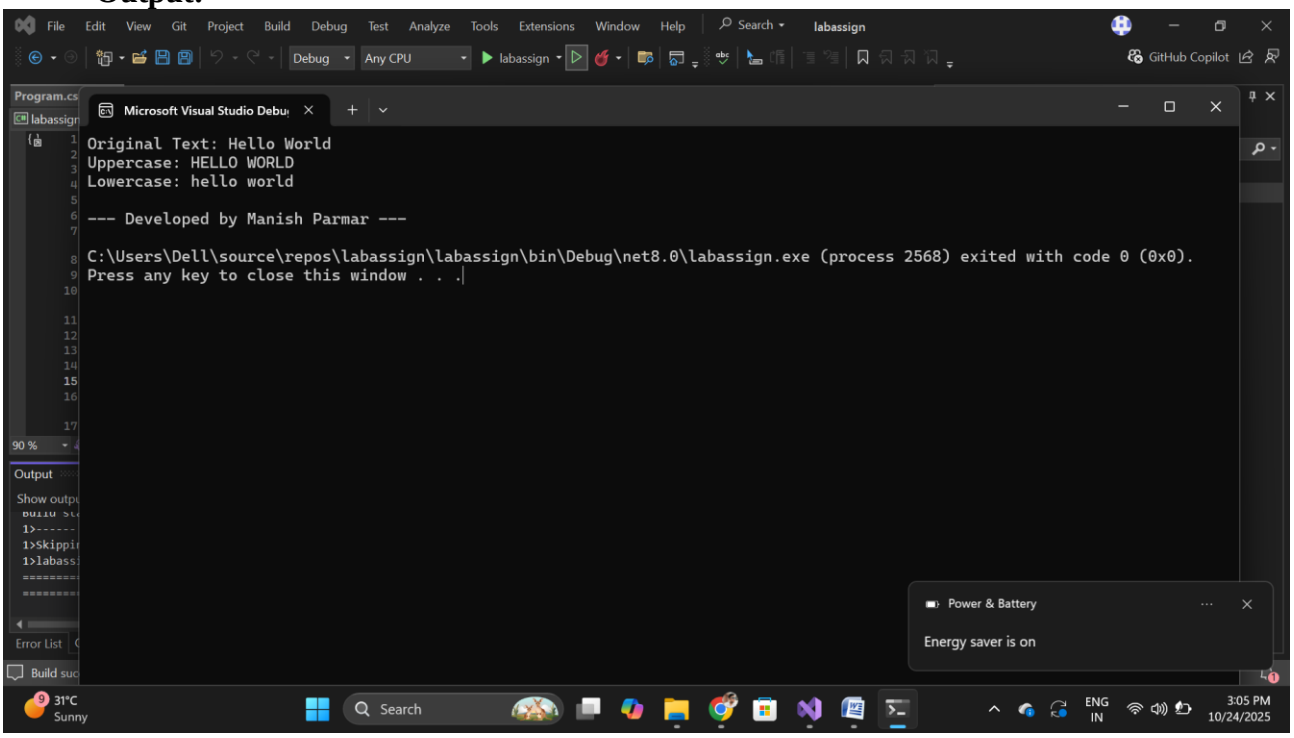
The string in lowercase.

Demonstrate calling both through the delegate on input "Hello World".

Ans 7.

```
using System;

delegate string FormatText(string text);

class LabAssignment5
{
    static string ToUpperCase(string text) { return text.ToUpper(); }
    static string ToLowerCase(string text) { return text.ToLower(); }

    static void Main()
    {
        string input = "Hello World";
        FormatText upper = ToUpperCase;
        FormatText lower = ToLowerCase;
        Console.WriteLine(upper(input));
        Console.WriteLine(lower(input));
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



Q 8. Create a delegate BillingOperation that accepts a product amount.
Implement four related methods: ShowTotal → Display original price.

ApplyDiscount → Apply 10% discount.

AddTax → Add 18% GST on discounted price.

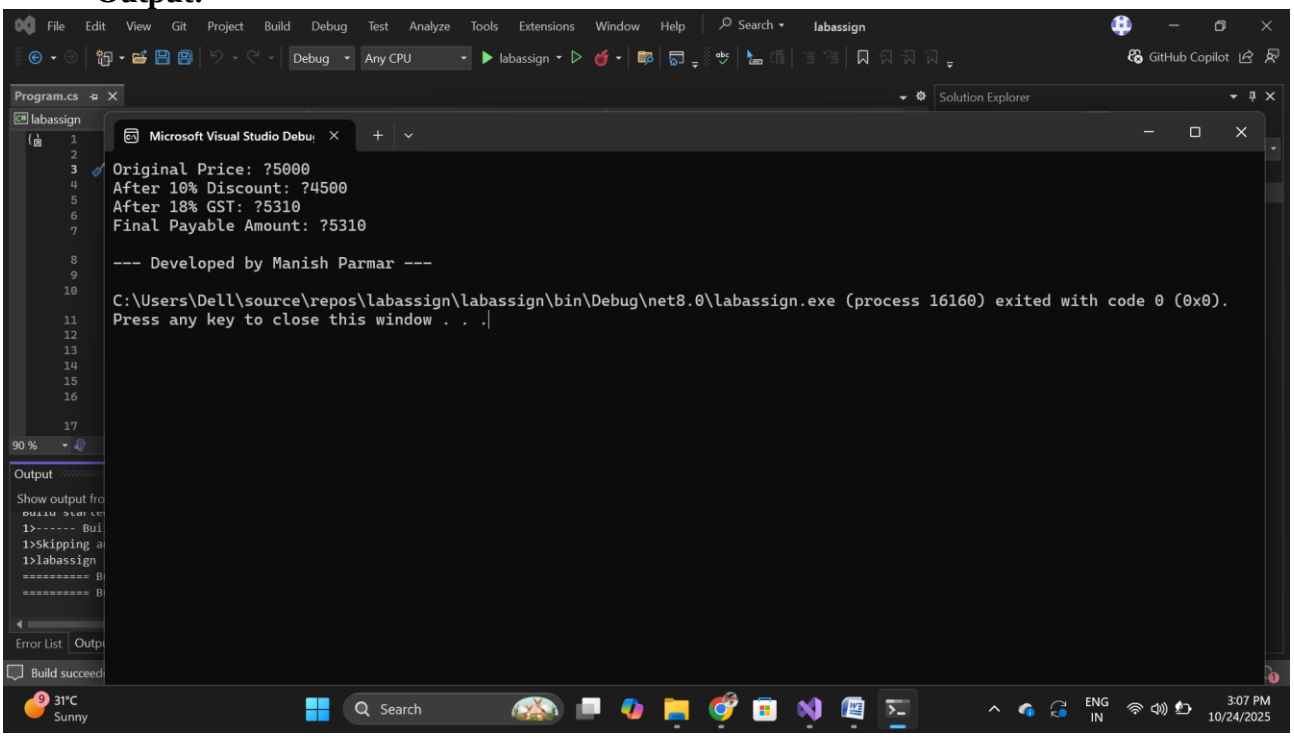
FinalBill → Display final payable amount.

Use delegate chaining to call these methods step by step for an item worth ₹5000.

Ans 8.

```
using System;

delegate void BillingOperation(decimal amount);

class LabAssignment5
{
    private decimal current;
    public void ShowTotal(decimal amount)
    {
        current = amount;
        Console.WriteLine("Original flrice: ₹" + current.ToString("0.00"));
    }
    public void ApplyDiscount(decimal amount)
    {
        current = current * 0.90m;
        Console.WriteLine("After 10% discount: ₹" + current.ToString("0.00"));
    }
    public void AddTax(decimal amount)
    {
        current = current + current * 0.18m;
        Console.WriteLine("After 18% GST: ₹" + current.ToString("0.00"));
    }
    public void FinalBill(decimal amount)
    {
        Console.WriteLine("Final flayable: ₹" + current.ToString("0.00"));
    }
}

class flrogram
{
    static void Main()
    {
        LabAssignment5 bill = new LabAssignment5();
        BillingOperation ops = bill.ShowTotal;
        ops += bill.ApplyDiscount;
        ops += bill.AddTax;
        ops += bill.FinalBill;
        ops(5000m);
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:



Q 9. Define a delegate ConvertTemperature that takes double input.
Implement two methods: Celsius to Fahrenheit.

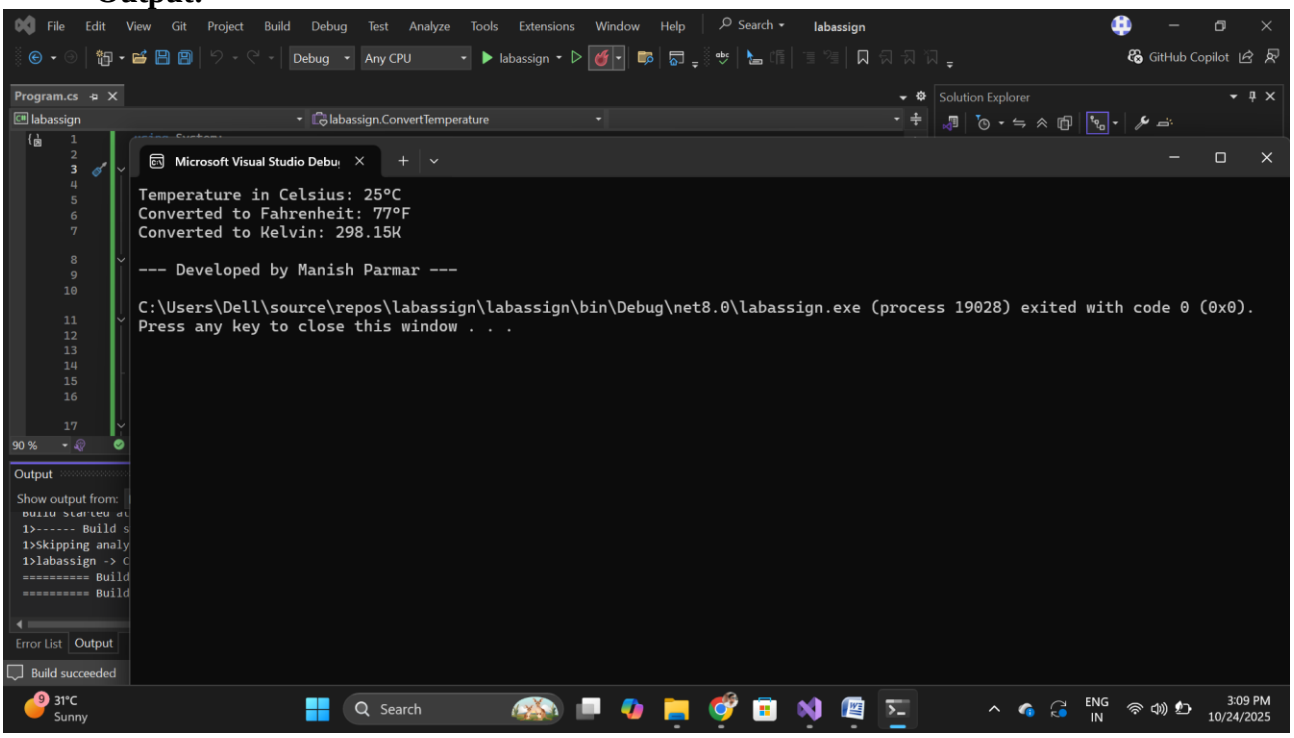
Celsius to Kelvin.

Show result for 25°C.

Ans 10.

```
using System;

delegate double ConvertTemperature(double celsius);

class LabAssignment5
{
    static double ToFahrenheit(double celsius) { return (celsius * 9 / 5) + 32; }
    static double ToKelvin(double celsius) { return celsius + 273.15; }

    static void Main()
    {
        double temp = 25;
        ConvertTemperature toF = ToFahrenheit;
        ConvertTemperature toK = ToKelvin;
        Console.WriteLine("Fahrenheit: " + toF(temp));
        Console.WriteLine("Kelvin: " + toK(temp));
        Console.WriteLine("Developed by Manish Parmar.");
    }
}
```

Output:


Temperature in Celsius: 25°C
Converted to Fahrenheit: 77°F
Converted to Kelvin: 298.15K
--- Developed by Manish Parmar ---
C:\Users\Dell\source\repos\labassign\labassign\bin\Debug\net8.0\labassign.exe (process 19028) exited with code 0 (0x0).
Press any key to close this window . . .
Build succeeded