**C# - Lab 01**

1. Create a Console application that can read your name and batch as an input and print them into the console.

**using System;**

**namespace NameAndBatchConsoleApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter your name:");**

**string name = Console.ReadLine();**

**Console.WriteLine("Please enter your batch:");**

**string batch = Console.ReadLine();**

**Console.WriteLine("Name: " + name);**

**Console.WriteLine("Batch: " + batch);**

**// You can also use string interpolation instead of concatenation:**

**// Console.WriteLine($"Name: {name}");**

**// Console.WriteLine($"Batch: {batch}");**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Create a Console application that can read the radius from the user and calculate the Area of a circle.

**using System;**

**namespace CircleAreaConsoleApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the radius of the circle:");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double radius))**

**{**

**double area = Math.PI \* radius \* radius;**

**Console.WriteLine("Area of the circle: " + area);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for the radius.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Create a console application that can read two input values and show the summation of the inputs.

**using System;**

**namespace SummationConsoleApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the first number:");**

**string input1 = Console.ReadLine();**

**Console.WriteLine("Please enter the second number:");**

**string input2 = Console.ReadLine();**

**if (double.TryParse(input1, out double number1) && double.TryParse(input2, out double number2))**

**{**

**double sum = number1 + number2;**

**Console.WriteLine("Summation: " + sum);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter valid numeric values for both inputs.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Create a console application that can read salary of an employee and tax rate. Then show salary after the tax.

**using System;**

**namespace SalaryAfterTaxConsoleApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the employee's salary:");**

**string salaryInput = Console.ReadLine();**

**Console.WriteLine("Please enter the tax rate (in decimal form, e.g., 0.2 for 20%):");**

**string taxRateInput = Console.ReadLine();**

**if (double.TryParse(salaryInput, out double salary) && double.TryParse(taxRateInput, out double taxRate))**

**{**

**if (salary >= 0 && taxRate >= 0 && taxRate <= 1)**

**{**

**double taxAmount = salary \* taxRate;**

**double salaryAfterTax = salary - taxAmount;**

**Console.WriteLine("Salary after tax: " + salaryAfterTax);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Salary and tax rate must be non-negative, and tax rate must be between 0 and 1.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter valid numeric values for salary and tax rate.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**Lab 02**

1. Write a Console Application to calculate the sum of two user input numbers.

**using System;**

**namespace SumCalculatorApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the first number:");**

**string input1 = Console.ReadLine();**

**Console.WriteLine("Please enter the second number:");**

**string input2 = Console.ReadLine();**

**if (double.TryParse(input1, out double number1) && double.TryParse(input2, out double number2))**

**{**

**double sum = number1 + number2;**

**Console.WriteLine("Sum: " + sum);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter valid numeric values for both inputs.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Write a Console Application to calculate sum, subtraction, multiplication and division of two user input numbers.

**using System;**

**namespace CalculatorApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the first number:");**

**string input1 = Console.ReadLine();**

**Console.WriteLine("Please enter the second number:");**

**string input2 = Console.ReadLine();**

**if (double.TryParse(input1, out double number1) && double.TryParse(input2, out double number2))**

**{**

**double sum = number1 + number2;**

**double subtraction = number1 - number2;**

**double multiplication = number1 \* number2;**

**if (number2 != 0) // Division by zero check**

**{**

**double division = number1 / number2;**

**Console.WriteLine("Sum: " + sum);**

**Console.WriteLine("Subtraction: " + subtraction);**

**Console.WriteLine("Multiplication: " + multiplication);**

**Console.WriteLine("Division: " + division);**

**}**

**else**

**{**

**Console.WriteLine("Division by zero is not allowed.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter valid numeric values for both inputs.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Write a Console Application to calculate area and circumference of a circle for given radius.

**using System;**

**namespace CircleCalculatorApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the radius of the circle:");**

**string radiusInput = Console.ReadLine();**

**if (double.TryParse(radiusInput, out double radius))**

**{**

**if (radius >= 0)**

**{**

**double area = Math.PI \* radius \* radius;**

**double circumference = 2 \* Math.PI \* radius;**

**Console.WriteLine("Area of the circle: " + area);**

**Console.WriteLine("Circumference of the circle: " + circumference);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Radius must be non-negative.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for the radius.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Write a Console Application to check if a given number is even or odd.

**using System;**

**namespace EvenOddCheckerApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter a number:");**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int number))**

**{**

**if (number % 2 == 0)**

**{**

**Console.WriteLine(number + " is an even number.");**

**}**

**else**

**{**

**Console.WriteLine(number + " is an odd number.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid integer number.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Upgrade the above console application which enables 10 user inputs and displays even or odd for each user input.

**using System;**

**namespace EvenOddCheckerApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**int numberOfInputs = 10;**

**Console.WriteLine("Please enter 10 numbers:");**

**for (int i = 1; i <= numberOfInputs; i++)**

**{**

**Console.Write($"Input {i}: ");**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int number))**

**{**

**if (number % 2 == 0)**

**{**

**Console.WriteLine($"{number} is an even number.");**

**}**

**else**

**{**

**Console.WriteLine($"{number} is an odd number.");**

**}**

**}**

**else**

**{**

**Console.WriteLine($"Invalid input for number {i}. Please enter a valid integer number.");**

**i--; // Decrement i to re-prompt for the same input number.**

**}**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**Lab 03**

1. Write a C# program that takes an integer as input and checks whether it is even or odd. Display the result “Even” or “Odd” accordingly.

**using System;**

**namespace EvenOddChecker**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter an integer:");**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int number))**

**{**

**if (number % 2 == 0)**

**{**

**Console.WriteLine("Even");**

**}**

**else**

**{**

**Console.WriteLine("Odd");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid integer.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Write a C# program that counts the number of vowels in a given string. Consider both uppercase and lowercase vowels.

**using System;**

**namespace VowelCounter**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter a string:");**

**string input = Console.ReadLine();**

**int vowelCount = CountVowels(input);**

**Console.WriteLine($"Number of vowels in the string: {vowelCount}");**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**static int CountVowels(string str)**

**{**

**int count = 0;**

**foreach (char c in str)**

**{**

**if (IsVowel(c))**

**{**

**count++;**

**}**

**}**

**return count;**

**}**

**static bool IsVowel(char c)**

**{**

**char lowerC = Char.ToLower(c);**

**return lowerC == 'a' || lowerC == 'e' || lowerC == 'i' || lowerC == 'o' || lowerC == 'u';**

**}**

**}**

**}**

1. Write a C# program to find the sum of the digits of a given number using a for loop.

**using System;**

**namespace SumOfDigits**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter an integer:");**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int number))**

**{**

**int sum = 0;**

**// Using a for loop to iterate through each digit of the number**

**for (int temp = Math.Abs(number); temp != 0; temp /= 10)**

**{**

**int digit = temp % 10;**

**sum += digit;**

**}**

**Console.WriteLine($"Sum of the digits of {number}: {sum}");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid integer.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Write a C# program to calculate the sum of all the odd numbers from to a given positive integer.

**using System;**

**namespace SumOfOddNumbers**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter a positive integer:");**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int n) && n > 0)**

**{**

**int sum = 0;**

**// Using a for loop to iterate from 1 to the given positive integer**

**for (int i = 1; i <= n; i++)**

**{**

**if (i % 2 != 0) // Check if the number is odd**

**{**

**sum += i;**

**}**

**}**

**Console.WriteLine($"Sum of odd numbers from 1 to {n}: {sum}");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid positive integer.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**Lab 04**

1. Create a C# Console application to convert user given Kilo Meter (km) Value to Meter (m) value. Take a separate Class call “ConvertValues” and inside the class create a method call kilometerTOmeter. (No return type No Parameter Method). And display the answer within the method. Then create an object in main Class (program class) and call the method.

using System;

**namespace KilometerToMeterConverterApp**

**{**

**class ConvertValues**

**{**

**public void KilometerToMeter()**

**{**

**Console.WriteLine("Please enter the distance in kilometers (km):");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double kilometers))**

**{**

**double meters = kilometers \* 1000;**

**Console.WriteLine($"{kilometers} km is equal to {meters} meters.");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for kilometers.");**

**}**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**ConvertValues converter = new ConvertValues();**

**converter.KilometerToMeter();**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**2.** Modify the same user defined method to method which accepts a parameter value. That parameter value is the user given Km value. (No return type with parameter method). Display the answer by using the class object.

**using System;**

**namespace KilometerToMeterConverterApp**

**{**

**class ConvertValues**

**{**

**public void KilometerToMeter(double kilometers)**

**{**

**double meters = kilometers \* 1000;**

**Console.WriteLine($"{kilometers} km is equal to {meters} meters.");**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the distance in kilometers (km):");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double kilometers))**

**{**

**ConvertValues converter = new ConvertValues();**

**converter.KilometerToMeter(kilometers);**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for kilometers.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**3.** Modify the same user defined method to method which accept a parameter and returns the answer at the end of the method. You should return the calculated Meter value at the end of the method. (With return type with parameter method). Display the answer by using object.

**using System;**

**namespace KilometerToMeterConverterApp**

**{**

**class ConvertValues**

**{**

**public double KilometerToMeter(double kilometers)**

**{**

**double meters = kilometers \* 1000;**

**return meters;**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the distance in kilometers (km):");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double kilometers))**

**{**

**ConvertValues converter = new ConvertValues();**

**double meters = converter.KilometerToMeter(kilometers);**

**Console.WriteLine($"{kilometers} km is equal to {meters} meters.");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for kilometers.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

4.Create a C# Console application to find the area and the circumference of a circle.

**using System;**

**namespace CircleAreaAndCircumference**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the radius of the circle:");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double radius))**

**{**

**if (radius >= 0)**

**{**

**double area = Math.PI \* radius \* radius;**

**double circumference = 2 \* Math.PI \* radius;**

**Console.WriteLine($"Area of the circle: {area}");**

**Console.WriteLine($"Circumference of the circle: {circumference}");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Radius must be non-negative.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for the radius.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. User should insert the radius value to the program. Program should contain a separate class call “FindValues” inside the separate class add two methods call findArea and findCircumference Both these methods are methods which takes parameters. As the parameter you should pass the radius value. By using above two methods find the area and circumference of the circle and return the answer from both methods. Create a class object in main class and call both methods and display the answers.

**using System;**

**namespace CircleAreaAndCircumference**

**{**

**class FindValues**

**{**

**public double FindArea(double radius)**

**{**

**return Math.PI \* radius \* radius;**

**}**

**public double FindCircumference(double radius)**

**{**

**return 2 \* Math.PI \* radius;**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please enter the radius of the circle:");**

**string input = Console.ReadLine();**

**if (double.TryParse(input, out double radius))**

**{**

**if (radius >= 0)**

**{**

**FindValues calculator = new FindValues();**

**double area = calculator.FindArea(radius);**

**double circumference = calculator.FindCircumference(radius);**

**Console.WriteLine($"Area of the circle: {area}");**

**Console.WriteLine($"Circumference of the circle: {circumference}");**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Radius must be non-negative.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid numeric value for the radius.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

**Lab 05**

1. Create the above mentioned console application and display it to the user. If user need to do an Addition user need to insert 1 as the choice. For subtraction it should be 2 etc. Your program should contain a separate class call “CalculateValues” and inside the class you should add four methods which perform four arithmetic operations. All the methods should take two parameters which are user inserted numbers.

**using System;**

**namespace ArithmeticCalculatorApp**

**{**

**class CalculateValues**

**{**

**public double Add(double num1, double num2)**

**{**

**return num1 + num2;**

**}**

**public double Subtract(double num1, double num2)**

**{**

**return num1 - num2;**

**}**

**public double Multiply(double num1, double num2)**

**{**

**return num1 \* num2;**

**}**

**public double Divide(double num1, double num2)**

**{**

**if (num2 != 0)**

**{**

**return num1 / num2;**

**}**

**else**

**{**

**throw new DivideByZeroException("Division by zero is not allowed.");**

**}**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Enter the first number:");**

**string input1 = Console.ReadLine();**

**Console.WriteLine("Enter the second number:");**

**string input2 = Console.ReadLine();**

**if (double.TryParse(input1, out double num1) && double.TryParse(input2, out double num2))**

**{**

**Console.WriteLine("Choose the operation:");**

**Console.WriteLine("1. Addition");**

**Console.WriteLine("2. Subtraction");**

**Console.WriteLine("3. Multiplication");**

**Console.WriteLine("4. Division");**

**Console.Write("Enter your choice (1/2/3/4): ");**

**string choiceInput = Console.ReadLine();**

**if (int.TryParse(choiceInput, out int choice))**

**{**

**CalculateValues calculator = new CalculateValues();**

**switch (choice)**

**{**

**case 1:**

**Console.WriteLine($"Result of Addition: {calculator.Add(num1, num2)}");**

**break;**

**case 2:**

**Console.WriteLine($"Result of Subtraction: {calculator.Subtract(num1, num2)}");**

**break;**

**case 3:**

**Console.WriteLine($"Result of Multiplication: {calculator.Multiply(num1, num2)}");**

**break;**

**case 4:**

**try**

**{**

**Console.WriteLine($"Result of Division: {calculator.Divide(num1, num2)}");**

**}**

**catch (DivideByZeroException ex)**

**{**

**Console.WriteLine(ex.Message);**

**}**

**break;**

**default:**

**Console.WriteLine("Invalid choice. Please select a valid option (1/2/3/4).");**

**break;**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid choice. Please enter a valid numeric value for the choice.");**

**}**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter valid numeric values for both numbers.");**

**}**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**

1. Add a separate class file to Console application program and create a method call private void sayHello(). Inside the method display hello world. In main class create object and try to access the sayHello() method by using the class object. Can you access the method? Explain why?

**using System;**

**namespace ConsoleApplication**

**{**

**public class MyClass**

**{**

**private void SayHello()**

**{**

**Console.WriteLine("Hello, World!");**

**}**

**}**

**}**

**using System;**

**namespace ConsoleApplication**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**MyClass myClassObject = new MyClass();**

**myClassObject.SayHello(); // This line will cause a compilation error**

**Console.ReadLine();**

**}**

**}**

**}**

1. Declare a Single dimensional array with 10 elements. Input the values to the array and find the followings, • Minimum value. • Maximum value. • Average value. • Reverse order of values. Hint – use a method which in separate class. And call the method from main the method.

**namespace ArrayProcessingApp**

**{**

**public class ArrayProcessor**

**{**

**public void ProcessArray(int[] arr)**

**{**

**int n = arr.Length;**

**// Find the minimum value**

**int min = arr[0];**

**for (int i = 1; i < n; i++)**

**{**

**if (arr[i] < min)**

**{**

**min = arr[i];**

**}**

**}**

**// Find the maximum value**

**int max = arr[0];**

**for (int i = 1; i < n; i++)**

**{**

**if (arr[i] > max)**

**{**

**max = arr[i];**

**}**

**}**

**// Calculate the average value**

**int sum = 0;**

**for (int i = 0; i < n; i++)**

**{**

**sum += arr[i];**

**}**

**double average = (double)sum / n;**

**// Reverse the array**

**int[] reversedArray = new int[n];**

**for (int i = 0; i < n; i++)**

**{**

**reversedArray[n - 1 - i] = arr[i];**

**}**

**// Display the results**

**Console.WriteLine("Minimum value: " + min);**

**Console.WriteLine("Maximum value: " + max);**

**Console.WriteLine("Average value: " + average);**

**Console.WriteLine("Reverse order of values:");**

**foreach (int num in reversedArray)**

**{**

**Console.Write(num + " ");**

**}**

**Console.WriteLine();**

**}**

**}**

**}**

Next

**using System;**

**namespace ArrayProcessingApp**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**int[] numbers = new int[10];**

**Console.WriteLine("Please enter 10 integers:");**

**for (int i = 0; i < 10; i++)**

**{**

**string input = Console.ReadLine();**

**if (int.TryParse(input, out int num))**

**{**

**numbers[i] = num;**

**}**

**else**

**{**

**Console.WriteLine("Invalid input. Please enter a valid integer.");**

**i--; // Decrement i to re-prompt for the same input number.**

**}**

**}**

**ArrayProcessor arrayProcessor = new ArrayProcessor();**

**arrayProcessor.ProcessArray(numbers);**

**Console.ReadLine(); // This is added to keep the console window open after displaying the output.**

**}**

**}**

**}**