(Q1) 1. Create a C# Console application to convert user given Kilo Meter (km) Value to Meter (m) value. Take a separate Class call “Convert Values” and inside the class create a method call kilometer to meter. (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.

class ConvertValues

{

public void KilometerToMeter()

{

Console.Write("Enter the kilometer value: ");

double kilometer = double.Parse(Console.ReadLine());

double meter = kilometer \* 1000;

Console.WriteLine($"{kilometer} kilometer is equal to {meter} meters.");

}

public void KilometerToMeter(double kilometer)

{

double meter = kilometer \* 1000;

Console.WriteLine($"{kilometer} kilometer is equal to {meter} meters.");

}

public double ConvertToMeter(double kilometer)

{

return kilometer \* 1000;

}

}

class Program

{

static void Main()

{

ConvertValues converter = new ConvertValues();

converter.KilometerToMeter();

Console.Write("Enter the kilometer value: ");

double kilometer = double.Parse(Console.ReadLine());

converter.KilometerToMeter(kilometer);

Console.Write("Enter the kilometer value: ");

kilometer = double.Parse(Console.ReadLine());

double meter = converter.ConvertToMeter(kilometer);

Console.WriteLine($"{kilometer} kilometer is equal to {meter} meters.");

}

}

**LAB. 04**

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.

namespace DistanceConverter

{

public class ConvertValues

{

public void KilometerToMeter(double kilometers)

{

double meters = kilometers \* 1000;

Console.WriteLine(kilometers + " km is equal to " + meters + " meters.");

}

}

}

namespace DistanceConverter

{

class Program

{static void Main(string[] args)

{ Console.WriteLine("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 number for kilometers.");

}

Console.ReadLine(); // To prevent the console from closing immediately

}

}

}

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 DistanceConverter

{

public class ConvertValues

{

public double KilometerToMeter(double kilometers)

{

double meters = kilometers \* 1000;

return meters;

}

}

}

namespace DistanceConverter

{

class Program

{ static void Main(string[] args)

{ Console.WriteLine("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 number for kilometers.");

}

Console.ReadLine(); // To prevent the console from closing immediately

}

}

}

(Q2) 1. Create a C# Console application to find the area and the circumference of a circle.

using System;

namespace CircleCalculator

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter the radius of the circle:");

string radiusInput = Console.ReadLine();

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

{

double area = CalculateCircleArea(radius);

double circumference = CalculateCircleCircumference(radius);

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

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

}

else

{

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

} Console.ReadLine(); // To prevent the console from closing immediately

} static double CalculateCircleArea(double radius)

{

return Math.PI \* Math.Pow(radius, 2);

} static double CalculateCircleCircumference(double radius)

{

return 2 \* Math.PI \* radius;

}

}

}

2. User should insert the radius value to the program. Program should contain a separate class call “Find Values” inside the separate class add two methods call find Area and find Circumference 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.

namespace CircleCalculator

{ public class FindValues

{public double FindArea(double radius)

{ return Math.PI \* Math.Pow(radius, 2);

} public double FindCircumference(double radius)

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

}

}

}

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter the radius of the circle:");

string radiusInput = Console.ReadLine();

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

{ FindValues finder = new FindValues();

double area = finder.FindArea(radius);

double circumference = finder.FindCircumference(radius);

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

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

} else

{ Console.WriteLine("Invalid input. Please enter a valid number for the radius.");

} Console.ReadLine(); // To prevent the console from closing immediately

}

}

}