LAB 05

Question 03

```csharp
using System;

namespace ArithmeticCalculator
{
    public class CalculateValues
    {
        public double Addition(double num1, double num2)
        {
            return num1 + num2;
        }

        public double Subtraction(double num1, double num2)
        {
            return num1 - num2;
        }
```

```csharp
    public double Multiplication(double num1, double num2)
    {
        return num1 * num2;
    }

    public double Division(double num1, double num2)
    {
        if (num2 == 0)
        {
            throw new DivideByZeroException("Cannot divide by zero.");
        }
        return num1 / num2;
    }
}

class Program
```

```csharp
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Enter 01 for addition");
            Console.WriteLine("Enter 02 for subtraction");
            Console.WriteLine("Enter 03 for multiplication");
            Console.WriteLine("Enter 04 for division");

            Console.WriteLine("Enter your choice :");
            string choiceInput = Console.ReadLine();

            if (int.TryParse(choiceInput, out int choice))
            {
                if (choice >= 1 && choice <= 4)
```

```csharp
{
    Console.WriteLine("Enter number 1:");
    string num1Input = Console.ReadLine();

    Console.WriteLine("Enter number 2:");
    string num2Input = Console.ReadLine();

    if (double.TryParse(num1Input, out double num1) && double.TryParse(num2Input, out double num2))
    {
        CalculateValues calculator = new CalculateValues();
        double result = 0;

        switch (choice)
```

```
            {
                case 1:
                    result =
calculator.Addition(num1, num2);
                    break;
                case 2:
                    result =
calculator.Subtraction(num1, num2);
                    break;
                case 3:
                    result =
calculator.Multiplication(num1, num2);
                    break;
                case 4:
                    try
                    {
                        result =
calculator.Division(num1, num2);
                    }
                    catch
(DivideByZeroException ex)
```

```csharp
                {
Console.WriteLine($"Error:
{ex.Message}");
                    return;
                }
                break;
            }

                Console.WriteLine($"Your
answer is : {result}");
                }
            else
            {

Console.WriteLine("Invalid input. Please
enter valid numbers.");
                }
            }
        else
        {
```

```
                Console.WriteLine("Invalid
choice. Please enter a valid option (1-
4).");
```

```
            }
        }
        else
        {
            Console.WriteLine("Invalid
input. Please enter a valid choice (1-
4).");
        }
    }
}
}
```

Question 04

In a C# console application, you can
create a separate class file and add a
private method like `private void
sayHello()` to it. However, you won't be
able to access the private method
directly from the main class or any other
class.

Here's an example of how you can create the separate class and private method:

Create a new class file named `HelloWorld.cs`:

```csharp
using System;

namespace ConsoleApp
{
    public class HelloWorld
    {
        private void sayHello()
        {
            Console.WriteLine("Hello, World!");
        }
    }
}
```

In the `Program.cs` (Main class) file, try to

access the `sayHello()` method:

```
using System;

namespace ConsoleApp
{
    class Program
    {
        static void Main(string[] args)
        {
            HelloWorld hello = new HelloWorld();
            // hello.sayHello(); // This will give a compile-time error
        }
    }
}
```

The class object "hello.sayHello()" will cause a compile-time error when you attempt to call the "sayHello()" method from the "Main" method, stating that the

method is unreachable owing to its protection level. The "sayHello()" method is only accessible within the "HelloWorld" class and not from other classes because it is a private method.

Change the "sayHello()" method's access modifier to "public" if you wish to use it outside of the "HelloWorld" class. The object of the "HelloWorld" class can then be used to call it from other classes.

Therefore, to respond to your inquiry, the private function "sayHello()" cannot be accessed from outside the "HelloWorld" class. You must alter the access modifier on it to "public" or any other less limiting modifier (for example, "internal," "protected," or "protected internal").