Practical Application of Calling Methods

Objective:

By the end of this activity, you will be able to call methods in C# within a complete program. You will gain hands-on experience in defining methods, calling methods with parameters, and using method return values to solve practical problems.

Step 1: Defining and Calling a Simple Method

Create a method that performs a task and call it in a program. This method should print a welcome message to the console.

Instructions:

1. Define a method named DisplayWelcomeMessage that prints "Welcome to our Program!" to the console.
2. Call the method to display the message.

Code:

```
class CallingMethods
{
    static void Main()
    {
        // Step 1
        DisplayWelcomeMessage();

        // Step 2
        Console.Write("Input your name: ");
        string? input = Console.ReadLine();
        if (string.IsNullOrWhiteSpace(input))
        {
            Console.WriteLine("Input cannot be empty. Please try again.");
            return;
        }
        DisplayWelcomeMessageWithName(input);

        // Step 3
        Console.WriteLine("Let's calculate the sum of two values.");

        Console.Write("Input a: ");
        input = Console.ReadLine();
        if (string.IsNullOrWhiteSpace(input) || !int.TryParse(input, out int
a))
        {
            Console.WriteLine("Invalid input for 'a'. Please enter a valid
number.");
            return;
        }

        Console.Write("Input b: ");
        input = Console.ReadLine();
        if (string.IsNullOrWhiteSpace(input) || !int.TryParse(input, out int
b))
```

```
        {
            Console.WriteLine("Invalid input for 'b'. Please enter a valid
number.");
            return;
        }

        int result = CalculateSum(a, b);
        Console.WriteLine($"The sum of {a} and {b} is {result}.");

        // Step 4
        Console.WriteLine("Let's check if a number is positive.");
        Console.Write("Input a number: ");
        input = Console.ReadLine();
        if (string.IsNullOrWhiteSpace(input) || !int.TryParse(input, out int
number))
        {
            Console.WriteLine("Invalid input. Please enter a valid number.");
            return;
        }

        bool isPositive = IsPositive(number);
        if (isPositive)
        {
            Console.WriteLine("The number is positive.");
        }
        else
        {
            Console.WriteLine("The number is negative.");
        }

        // Step 5
        Console.WriteLine("Let's check if you're old enough to drive.");
        Console.Write("Input your age: ");
        input = Console.ReadLine();
        if (string.IsNullOrWhiteSpace(input) || !int.TryParse(input, out int
age))
        {
            Console.WriteLine("Invalid age input. Please enter a valid
number.");
            return;
        }

        bool canDrive = IsOldEnoughToDrive(age);
        if (canDrive)
        {
            Console.WriteLine("You are old enough to drive.");
        }
        else
        {
            Console.WriteLine("Sorry, you are not old enough to drive.");
        }
    }

    static void DisplayWelcomeMessage()
    {
        Console.WriteLine("Welcome to our Program!");
    }

    static void DisplayWelcomeMessageWithName(string name)
    {
        Console.WriteLine("Hello " + name + "!");
    }

    static int CalculateSum(int x, int y)
    {
```

```
        return x + y;
    }

    static bool IsPositive(int number)
    {
        return number >= 0;
    }

    static bool IsOldEnoughToDrive(int age)
    {
        return age >= 18;
    }
}
```