Lesson 05 Demo 02

Working with IIFEs and Functions

Objective: To demonstrate the process of implementing Immediately Invoked Function Expressions (IIFEs) and functions in JavaScript for enhancing higher-order function usage, and efficient execution

Tools required: Visual Studio Code

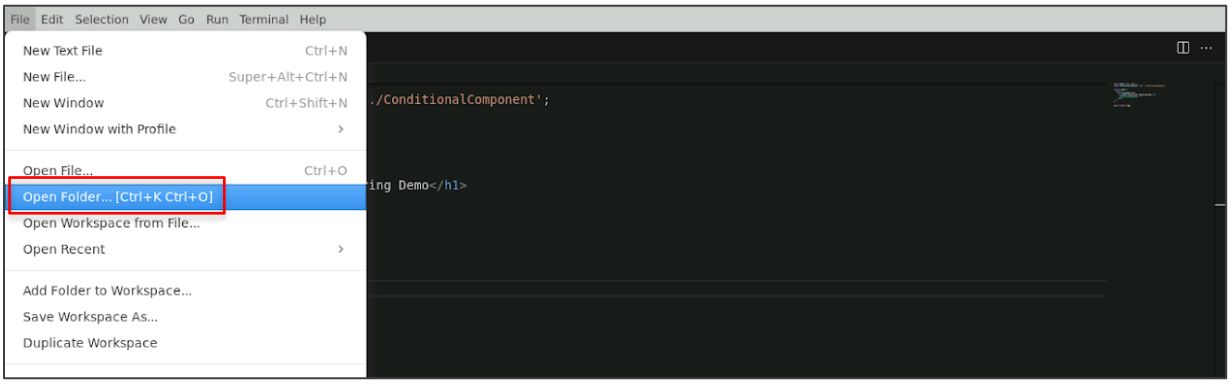
Prerequisites: None

Steps to be followed:

1. Write a JavaScript program using IIFEs and functions
2. Execute and verify the functionality of IIFEs and functions

Step 1: Write a JavaScript program using IIFEs and functions

1.1 Navigate to Visual Studio Code, right-click on the **File** menu of the code editor, and select the **Open Folder…** option:



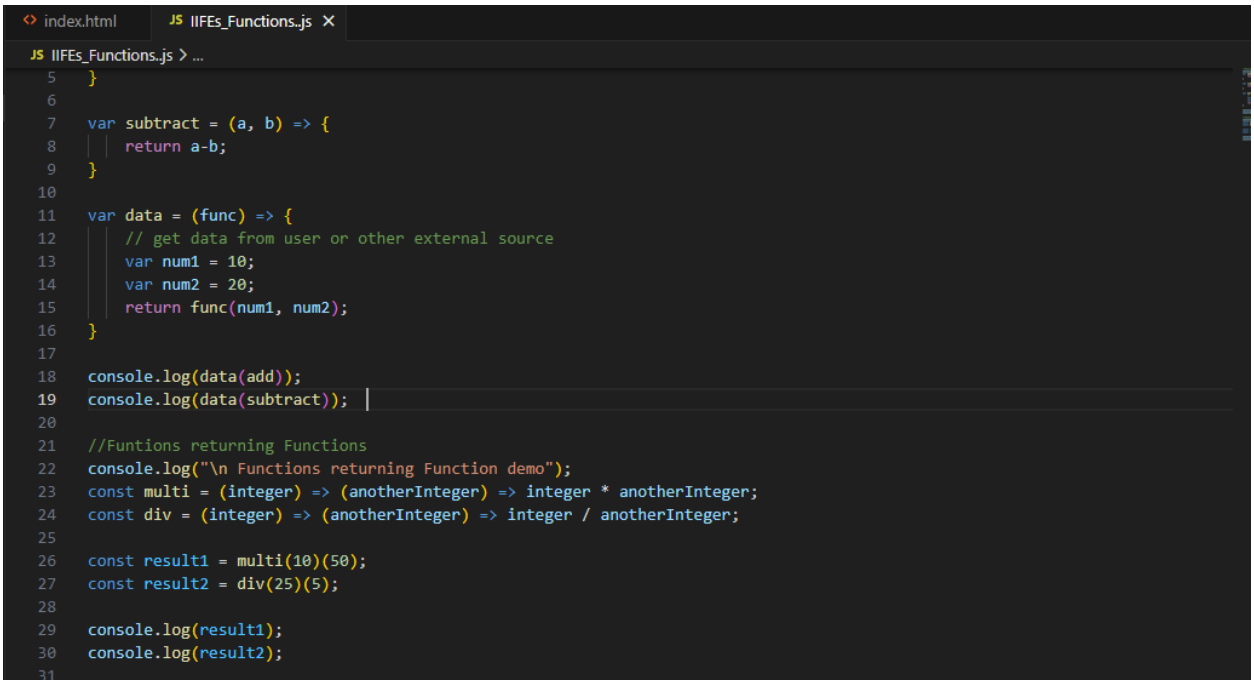
1.2 Write the given code in **index.html**:

```
<html>
  <body>
    <h1>MEAN Stack</h1>
    <p> IIFEs Function Demo</p>
    <script src="IIFEs_Functions.js"></script>
  </body>
</html>
```

1.3 Click on the **src** folder of the project, select New File option, enter the filename as **IIFEs_Functions.js,** and write the code shown below:

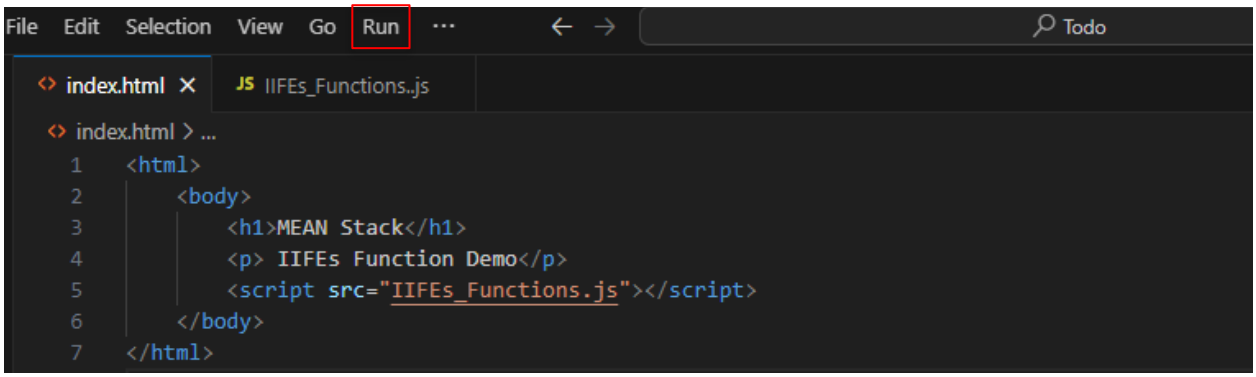
```
//Function as an argument
console.log("\n Functions as an argument demo");
var add = (a, b) => {
   return a+b;
}
var subtract = (a, b) => {
   return a-b;
}
var data = (func) => {
   // get data from user or other external source
   var num1 = 10;
   var num2 = 20;
   return func(num1, num2);
}
console.log(data(add));
console.log(data(subtract));
```

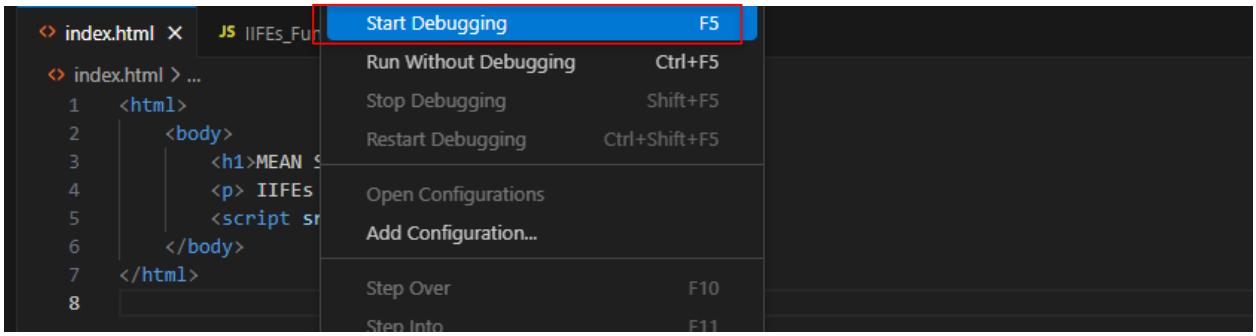
```
//Funtions returning Functions
console.log("\n Functions returning Function demo");
const multi = (integer) => (anotherInteger) => integer * anotherInteger;
const div = (integer) => (anotherInteger) => integer / anotherInteger;
const result1 = multi(10)(50);
const result2 = div(25)(5);
console.log(result1);
console.log(result2);
```



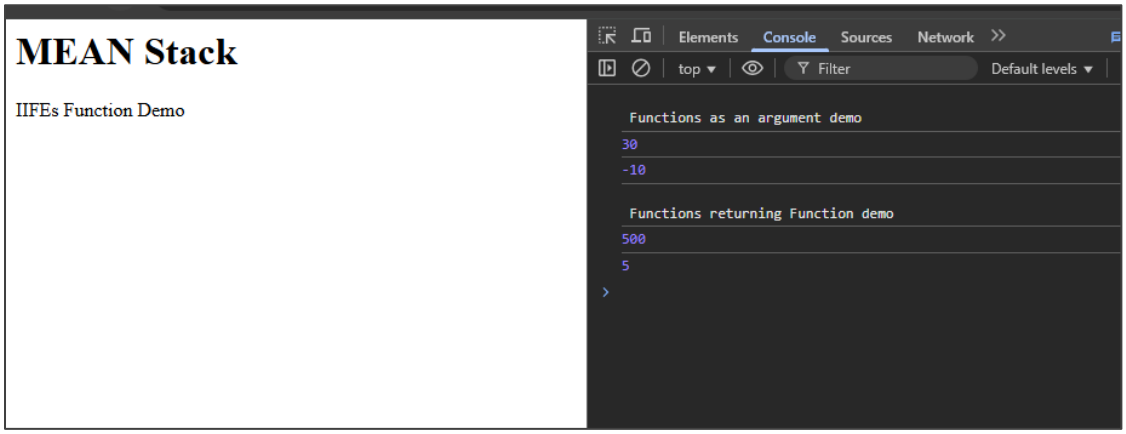
Step 2: Execute and verify the functionality of IIFEs and functions

2.1 Click on **Run** and then on **Start Debugging** to execute the JavaScript file





2.2 When the server starts running, right-click and select the **Inspect Element** option and click on the **Console** tab to see the output



By following the above steps, you have successfully implemented IIFEs and functions in JavaScript, demonstrating their role in modular programming, higher-order function execution, and functional programming techniques for cleaner and more reusable code.