

Closure and lexical Scope

Topics

1. Lexical scope
2. Closure scope chain

Lexical Scope

A variable is only accessible within the block of code where it is defined and any inner blocks, but not outside of it. The scope is determined at the time the code is written, not when it is executed.

[bhai dekh ek function ke andar jo bhi variable define krega usko function ke andar jo function bana ha bs usse hi tu access kr paega bhr se access nhi hoga issi ko lexical scope bolte ha]

```
function outerFunction() {  
  
    let outerVar = "I am outside!";  
    [outerVar jo variable ha usko sirf function ke andar hi use kr skte ha  
    outerFunction ke bhr nhi access hoga]  
  
    function innerFunction() {  
        console.log(outerVar); // Accesses outerVar from outerFunction's scope  
    }  
  
    innerFunction();  
}  
  
outerFunction();
```

Closure

A closure in JavaScript is a feature where an inner function has access to the outer (enclosing) function's variables, even after the outer function has finished executing. This means that the inner function "remembers" the variables from its surrounding scope, allowing it to use those variables later, even when the outer function is no longer active.

Closure एक ऐसा फीचर है JavaScript में, जहाँ एक inner function (अंदर का फंक्शन) अपने बाहरी (outer) function के variables को याद रखता है और उन पर काम कर सकता है, भले ही बाहरी function खत्म हो गया हो।

```
function outerFunction() {  
    let counter = 0; // This variable is in the outer function's scope  
  
    function innerFunction() {  
        counter++; // The inner function can access and modify the outer  
function's variable  
        console.log(counter);  
    }  
}
```

```

    }

    return innerFunction; // The outer function returns the inner function
}

const incrementCounter = outerFunction(); // outerFunction is called, and its
inner function is returned

incrementCounter(); // Output: 1
incrementCounter(); // Output: 2
incrementCounter(); // Output: 3

```

Practical use of Closure and lexical scope

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Closure aur chai</title>
</head>
<body style="background-color: #313131;">
  <button id="orange">Orange</button>
  <button id="green">Green</button>
</body>

  <script>

    function clickHandler(color){
      return function(){
        document.body.style.backgroundColor = `${color}`
      }
    }

    document.getElementById('orange').onclick = clickHandler("orange")
    document.getElementById('green').onclick = clickHandler("green")

  </script>
</html>

```