Javascript Scope

Last Updated : 26 Jun, 2024

JavaScript Scope is the area where a variable (or function) exists and is accessible. We can layer the scope in a system which means the child scope can access the parent scope but not vice-versa.

Javascript has different scopes

Table of Content

Global Scope

Those variables which are declared outside the function or blocks or you can say curly braces({}) are having a global scope.

In a JavaScript program, global variables can be accessed from anywhere.

The scope of var, let, and const are quite equivalent when declared outside the function.

Example: In this example, we will declare all the variables outside the function and now we can access those variables from anywhere in the Javascript program.

```
var global_variable1 = "Geeksforgeeks";
let global_variable2 = "Geeks";
const global_variable3 = "GFG";

function check_global_variables(){
```

```
    console.log(global_variable1);
    console.log(global_variable2);
    console.log(global_variable3);
}

check_global_variables();
```

Output

```
Geeksforgeeks
Geeks
GFG
```

Function Scope

Those variables that are declared inside a function have local or function scope which means variables that are declared inside the function are not accessible outside that function.

When declared inside a function, variables declared with var, let, and const look quite similar.

Note: Functions and Objects are also variables in Javascript.

> *Variables which are declared within the function have function scope. When var declared in the function have a function scope. Function uses the curly braces to store the function body . Therefore, block scope and function scope is same for the function body.*

Example: In this example, we will declare a variable inside the function and try to access it from outside the function.

```
function check_function_scope(){
    var variable1 = "Geeksforgeeks";
    let variable2 = "Geeks";
    const variable3 = "GFG";
    console.log(variable1);
    console.log(variable2);
}

check_function_scope();
console.log(variable3);
```

Output: In the above example, we tried to access the variable outside the function but it gives the reference error because of the function

scope.

```
Geeksforgeeks
Geeks
ReferenceError: variable3 is not defined
```

Block Scope

Before the ECMA6(2015) we have only global and function scope but when the Let and Const keywords were introduced in ES6 they provide the block scope.

Variables that are declared inside the { } (curly braces) can only access inside that scope not from the outside of it.

> *Variables declared with var do not have block scope only let and const have block scope.*

Example: In this example, we will declare the variable inside the block scope.

```
{
    var variable_1 = "GFG";
    const variable_2 = "Geeksforgeeks";
    let x=2;
    x*=2;
    console.log(x);
    console.log(variable_2);
}

console.log(variable_1);
```

Output: In the above example, we have successfully accessed the variable with the var keyword because var does not have a block scope.

```
4
Geeksforgeeks
GFG
```

Lexical Scope

The variable is declared inside the function and can only be accessed inside that block or nested block is called lexical scope.

Example: In this example, we will declare the variable inside the function and going to access inside the nested function.

```
function outerFunction() {
    const outerVariable = "Hello";

    function innerFunction() {
        const innerVariable = "Geeks";
        console.log(`${outerVariable} ${innerVariable}`);
    }

    innerFunction();
}

outerFunction();
```

Output:

```
Hello Geeks
```

Global Variables in HTML

The Global variables defined with the var keyword are part of the window object. The window object is the global scope in HTML.

Lifetime of JavaScript Variables

- A JavaScript variable's lifetime begins when it is declared.
- When the function is completed, the function (local) variables are erased.
- When you close a browser window, global variables are erased.

Comment More info

Campus Training Program

Next Article

How to Hide Option in Select Menu with CSS ?

Similar Reads

Lexical Scope in JavaScript