

JS

Hoisting In JavaScript



Muhammad Uzair

@codewith_uzair



JS

What is Hoisting?

JavaScript Hoisting refers to the process whereby the interpreter appears to move the declaration of functions, variables or classes to the top of their scope, prior to execution of the code.

This means that no matter where functions and variables are declared, they are moved to the top of their scope regardless of whether their scope is global or local.



Muhammad Uzair

@codewith_uzair



JS

Example

Let's take a look at an example:

```
1 console.log(x);  
2  
3 var x = 5;
```

In this example, we are trying to log the value of **x** before it has been declared



Muhammad Uzair

@codewith_uzair



JS

Example (Cont)

However, because of hoisting, the variable declaration is moved to the top of scope resulting in undefined output.

This is what the code looks like after hoisting:

```
1  var x;  
2  console.log(x);  
3  
4  x = 5;  
5
```

3



Muhammad Uzair

@codewith_uzair



JS

Example

Let's take a look at an example:

```
1 // program to print the text
2 greet();
3
4 function greet() {
5     console.log("Hi, there.");
6 }
7
```

In this example, we are trying to call **greet** before it has been declared



Muhammad Uzair

@codewith_uzair



Example (Cont)

However, because of hoisting, the function declaration is moved to the top of scope resulting in "Hi, there." output.

This is what the code looks like after hoisting:

```
1  function greet() {  
2    console.log("Hi, there.");  
3  }  
4  
5  // program to print the text  
6  greet();
```

5



Muhammad Uzair

@codewith_uzair





Muhammad Uzair

@codewith_uzair

Did You Find It **Useful?**

Share Your Thoughts

