JavaScript Hoisting

By CODEMIND Technology

Contact us 966 5044 698
966 5044 598

Hoisting

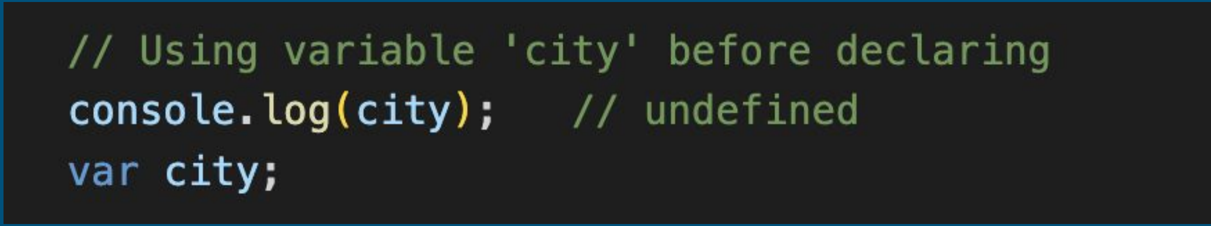
- JavaScript Hoisting concept understanding
- Variable hoisting
  - Variable declared using var keyword
  - Variable declared using let and const keyword
- Function Hoisting
  - Regular function hoisting
  - Function expression hoisting

Hoisting in JavaScript

Hoisting in JS is a behavior in which a variable or a function can be used before declaration.

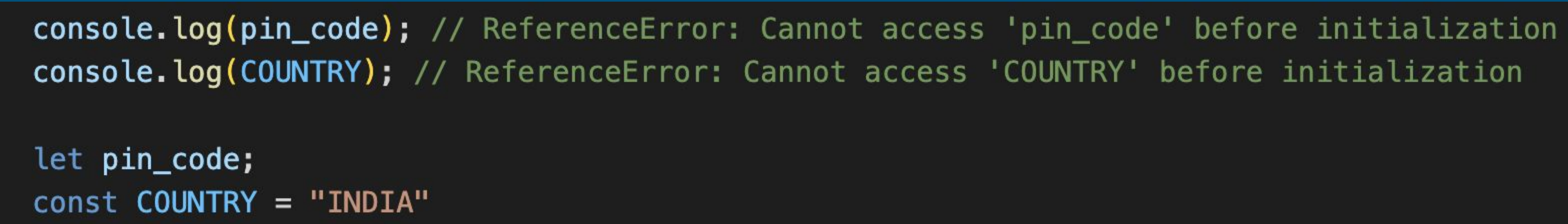
For variables only those variables will be hoisted that is declared using ‘var’ keyword and not using the ‘let’ and ‘const’ keyword

This will work as variable ‘city’ is declared using ‘var’ keyword



```javascript
// Using variable 'city' before declaring
console.log(city);    // undefined
var city;
```

Variable declared using ‘let’ and ‘const’ keyword are not hoisted as shown in below snippet



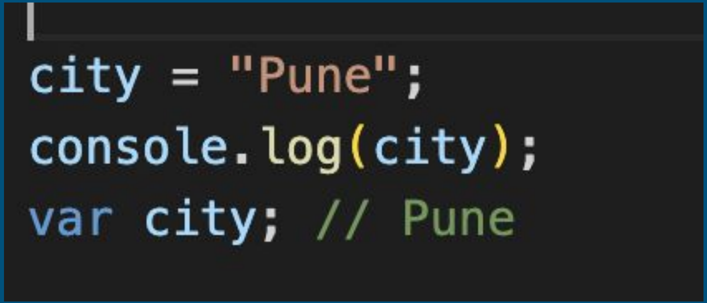
```javascript
console.log(pin_code); // ReferenceError: Cannot access 'pin_code' before initialization
console.log(COUNTRY); // ReferenceError: Cannot access 'COUNTRY' before initialization

let pin_code;
const COUNTRY = "INDIA"
```

Variable Hoisting

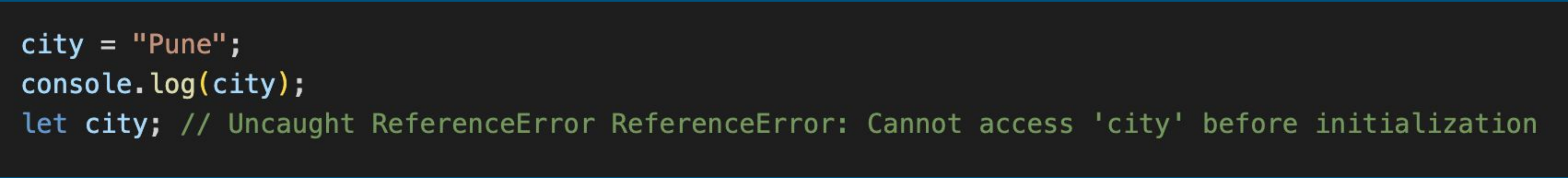
In terms of variables and constants, keyword var is hoisted and let and const does not allow hoisting.

Here variable ‘city’ is declared using var keyword
Hence it will be hoisted and will get the output



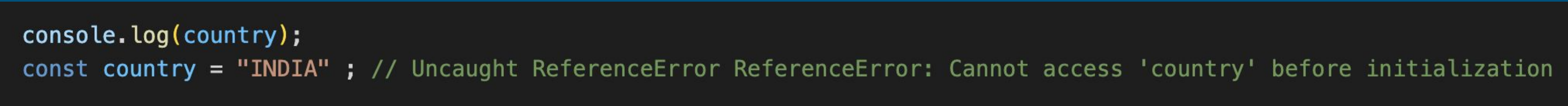
```
city = "Pune";
console.log(city);
var city; // Pune
```

In below snippet variable ‘city’ is declared using let keyword, Hence It will not be hoisted



```
city = "Pune";
console.log(city);
let city; // Uncaught ReferenceError ReferenceError: Cannot access 'city' before initialization
```

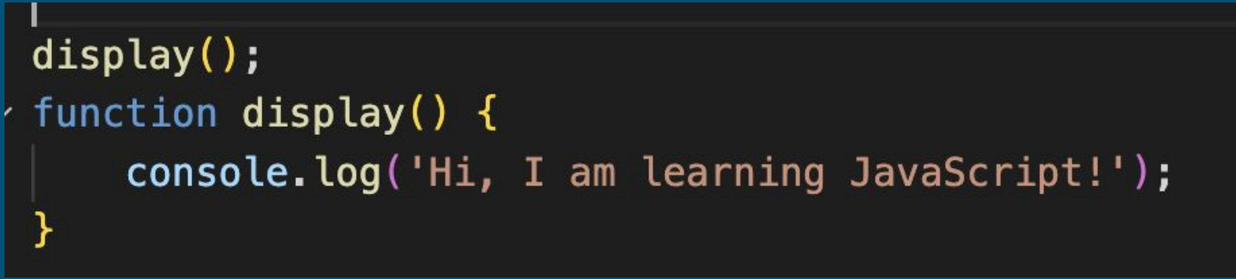
In below snippet variable country is declared using const keyword, Hence It will not be hoisted



```
console.log(country);
const country = "INDIA" ; // Uncaught ReferenceError ReferenceError: Cannot access 'country' before initialization
```

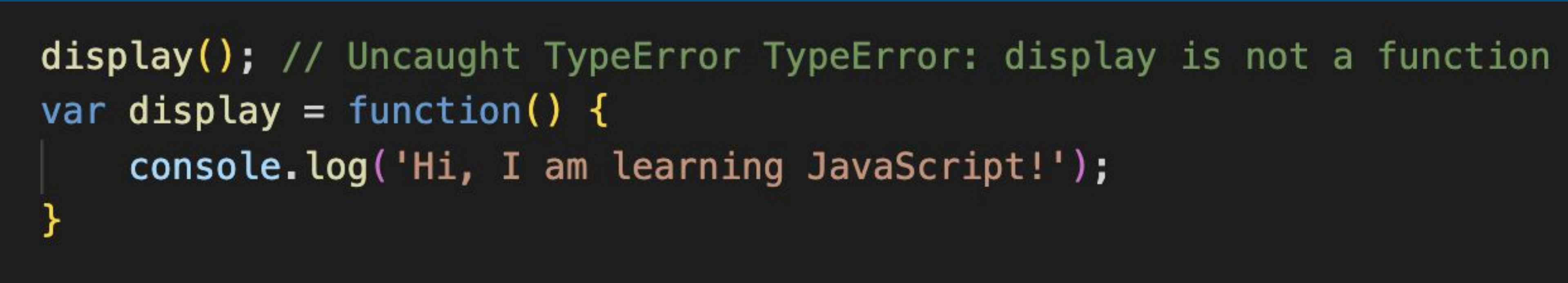
Function hoisting: A function can be called or invoked before declaring it

In the below snippet, the function display( ) is called before declaring it and the program shows the correct output. This is due to in JS function are hoisted.



```
display();
function display() {
    console.log('Hi, I am learning JavaScript!');
}
```

However, when a function is used as an expression, an error occurs because only regular function declarations are hoisted and not the function expression. As shown in below snippet



```
display(); // Uncaught TypeError TypeError: display is not a function
var display = function() {
    console.log('Hi, I am learning JavaScript!');
}
```

Contact us - 96650 44698

Thank you