JAVASCRIPT HOISTING

JS Hoisting

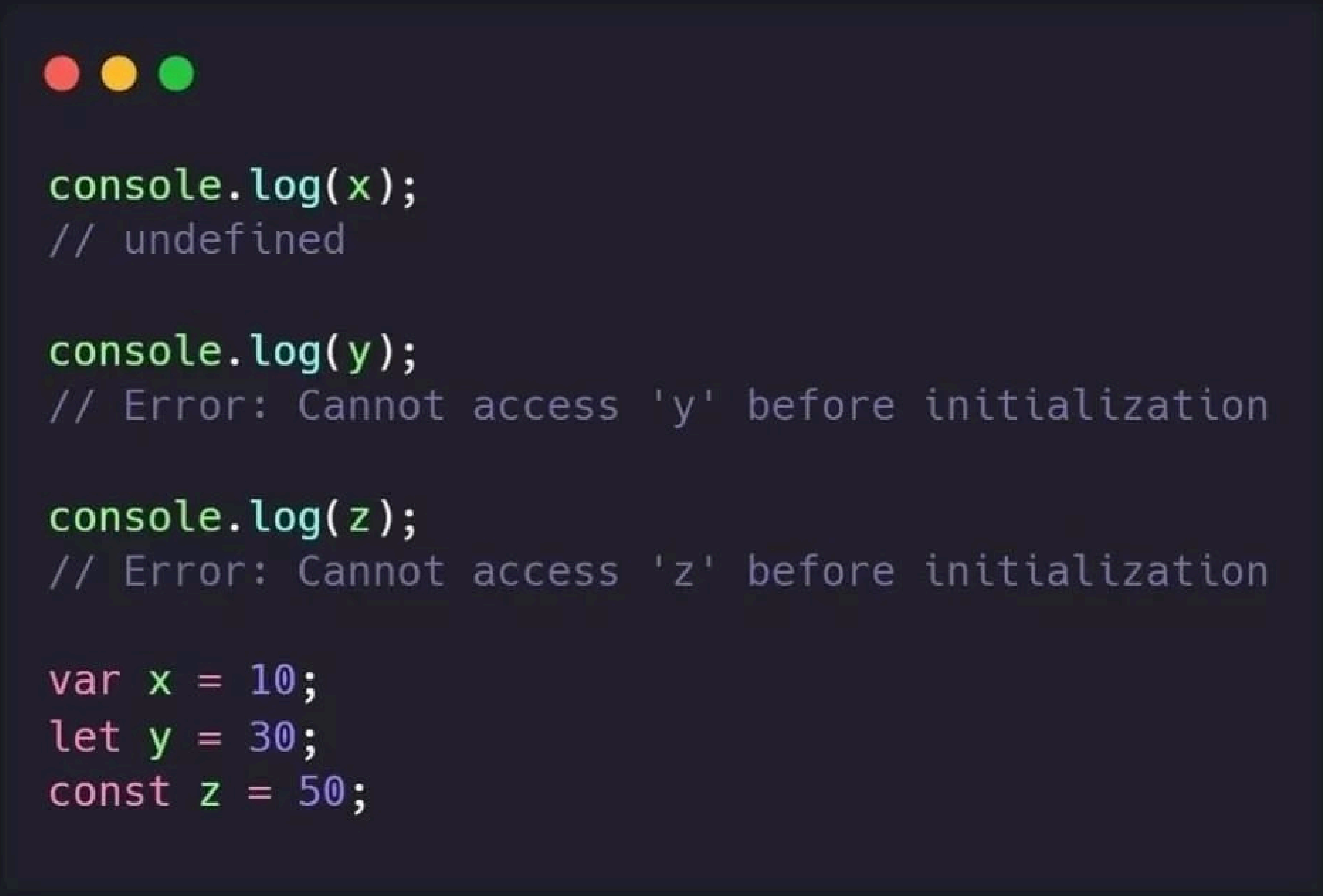
Hoisting is a process where **variable** & **function declarations** are moved to the top of their scope during code execution.

It allows you to use variables and call functions before they are defined in the code. However, **only the declarations are hoisted**, not the assignments/initialization.

Understanding hoisting helps in writing code that works as expecte

JS Hoisting (Variables)

Hoisting with let, const & var different

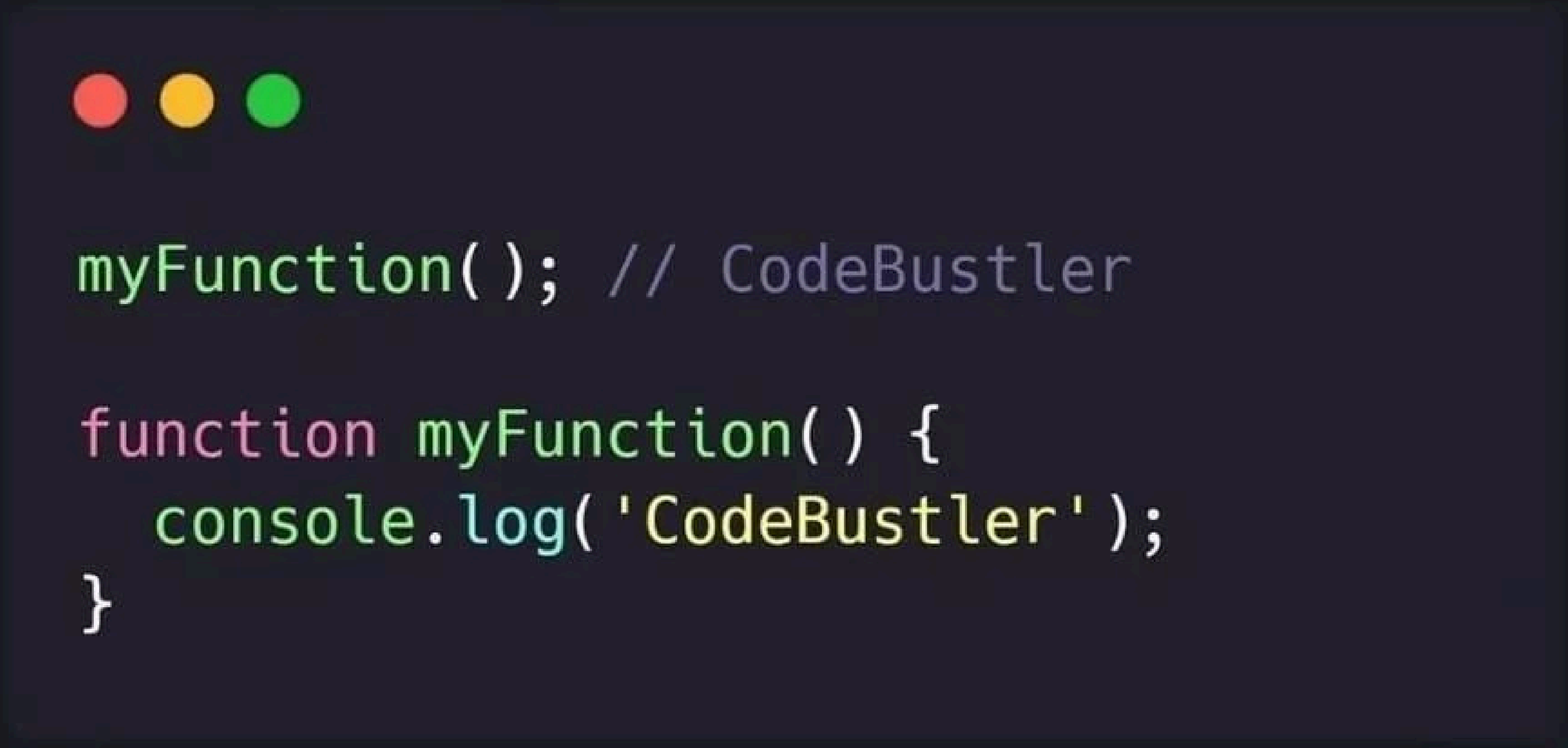


```
console.log(x);
// undefined

console.log(y);
// Error: Cannot access 'y' before initialization

console.log(z);
// Error: Cannot access 'z' before initialization

var x = 10;
let y = 30;
const z = 50;
```

The variable will be **undefined** until the line where its **initialized is reached.**

JS Hoisting (Functions)

Function hoisting means that you can **call a function before it is declared** in your code, and it will still work.



```javascript
myFunction(); // CodeBustler

function myFunction() {
  console.log('CodeBustler');
}
```

Function expression & class expression are not allowed ⚠

FOLLOW