



**TAKE 2 MINUTES  
TO LEARN  
'this' IN JS**

**JS**



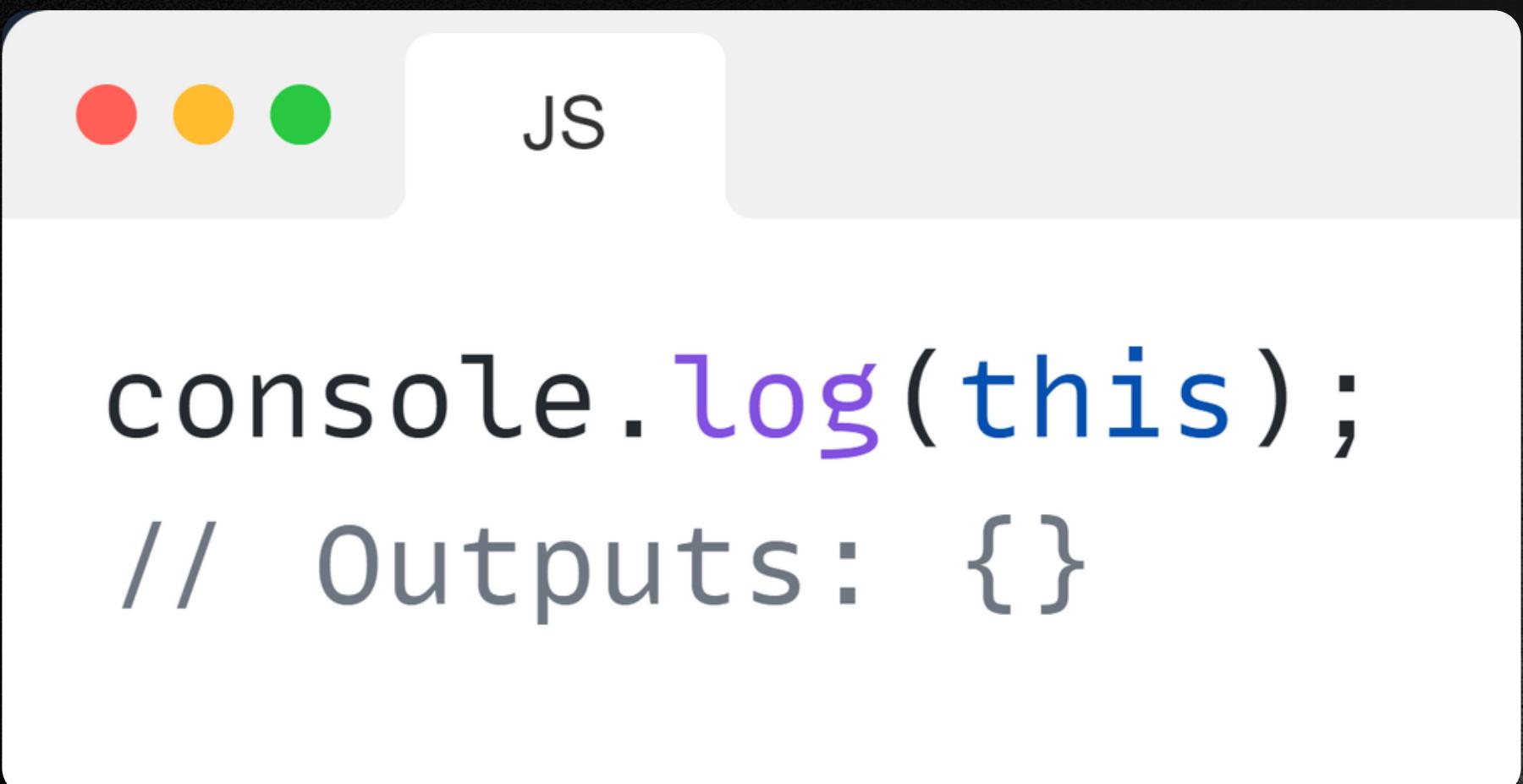
*You Must know*

[ Don't Miss It ]



# What is this ?

- this refers to the context within which a function is executed. It dynamically binds to different values depending on how a function is called.
- In the global context, this refers to the global object in Node.js.



```
console.log(this);
// Outputs: {}
```



# Function Context

- In regular functions, this refers to the object that calls the function. If no explicit context is provided, it defaults to the global object.



A screenshot of a browser window with a light gray header bar containing three colored dots (red, yellow, green) and the text "JS". The main content area shows the following JavaScript code:

```
function greet() {  
  console.log("Hello, " + this.name);  
}  
  
const person = { name: "Alice", greet: greet };  
person.greet(); // Outputs: Hello, Alice
```



# Arrow Functions

- Unlike regular functions, arrow functions do not have their own this context. Instead, they inherit this from the surrounding lexical scope.

```
JS

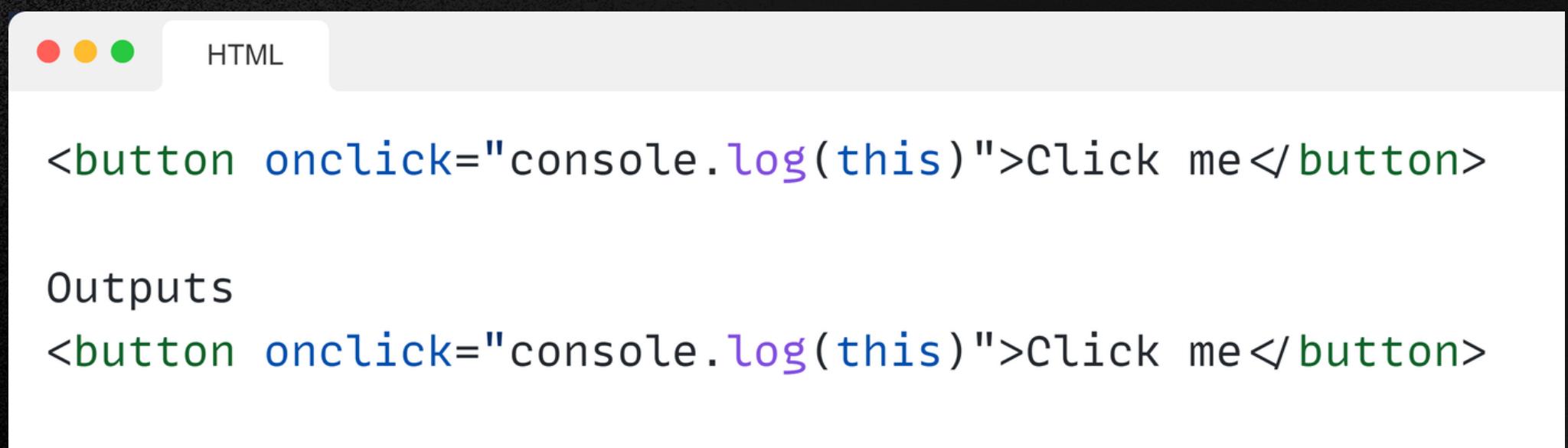
const printName = () => {
  console.log("My name is " + this.name);
};

const user = { name: "Bob", printName: printName };
user.printName(); // Outputs: My name is undefined
```



# Event Handlers

- In event handlers, this typically refers to the element that triggered the event. However, it may vary depending on how the handler is attached.

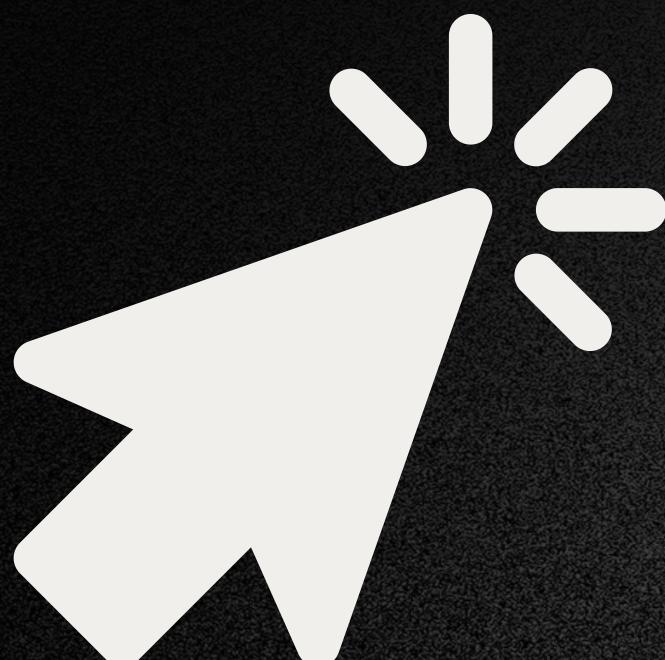


A screenshot of a web browser window. The title bar says "HTML". The main content area contains the following code:

```
<button onclick="console.log(this)">click me</button>
```

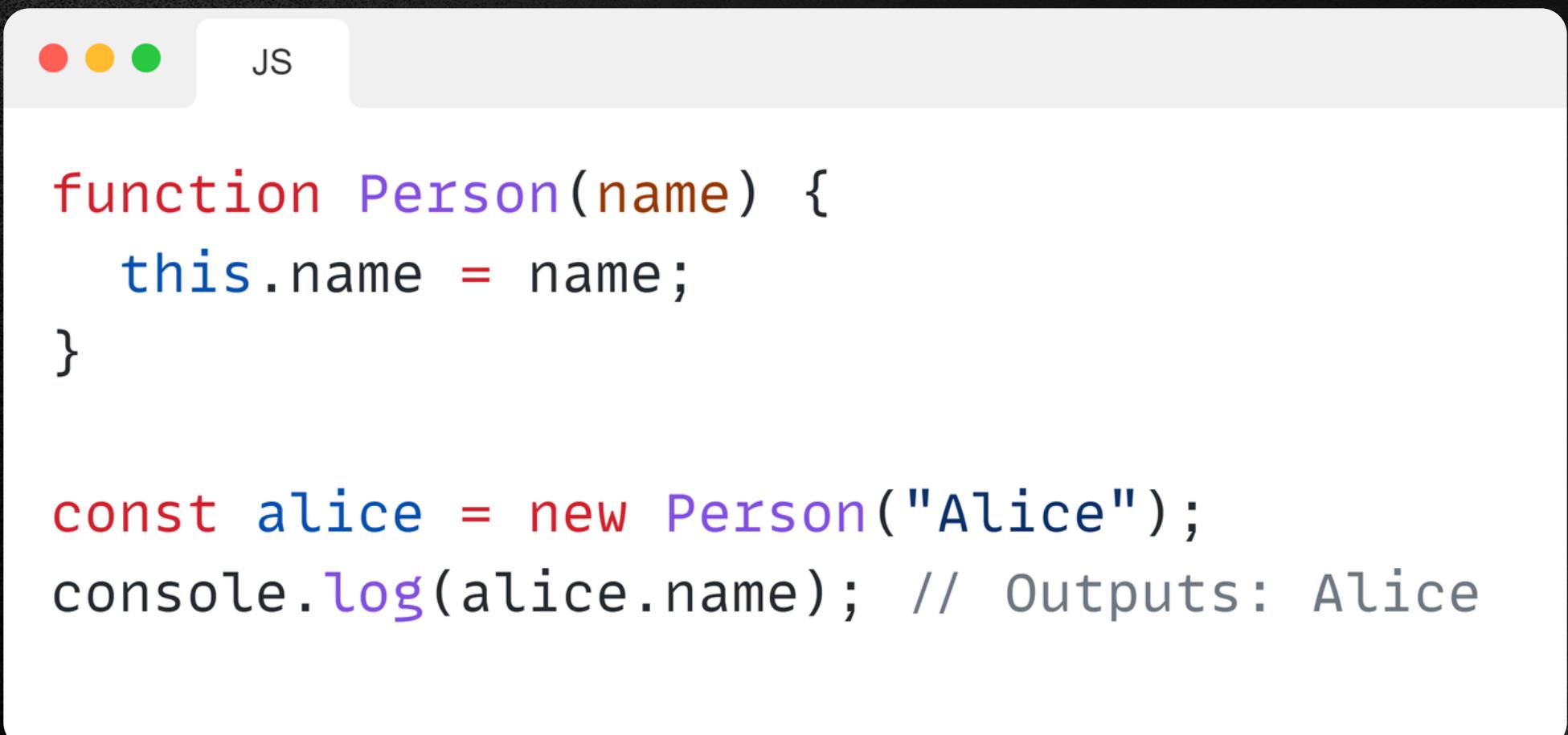
Below the code, the text "Outputs" is followed by the same code again:

```
<button onclick="console.log(this)">click me</button>
```



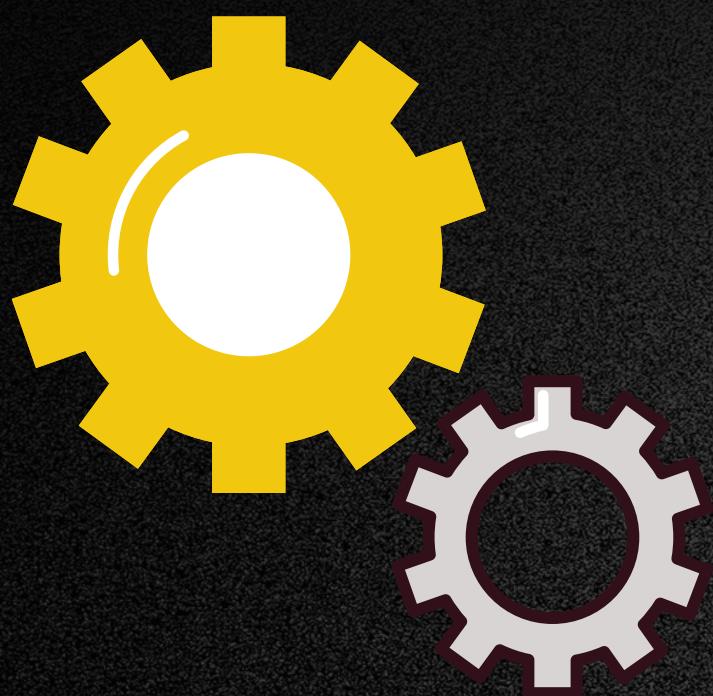
# Constructor Functions

- Inside constructor functions, this refers to the newly created instance of the object being constructed.



The screenshot shows a browser window with three colored window controls (red, yellow, green) at the top left. The tab bar has a single tab labeled "JS". The main content area contains the following JavaScript code:

```
function Person(name) {  
  this.name = name;  
}  
  
const alice = new Person("Alice");  
console.log(alice.name); // Outputs: Alice
```



# Bonus Interview Question

**What is the 'this' keyword in  
JavaScript?**

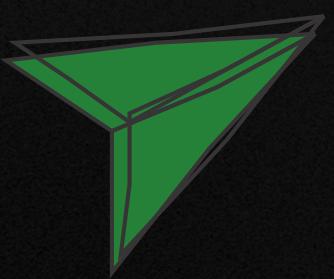
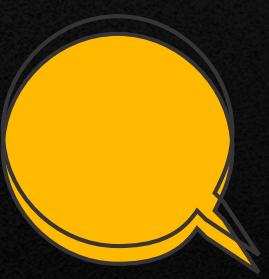


The 'this' keyword in JavaScript refers to the currently calling object.

It references the object that is executing the current function. If the function being referenced is a regular function, “this” references the global object.



**DID  
YOU FIND  
IT  
HELPFUL ?**



Share this with a friend who needs it!



**VINCENT RAJA**  
FULL STACK WEB DEVELOPER