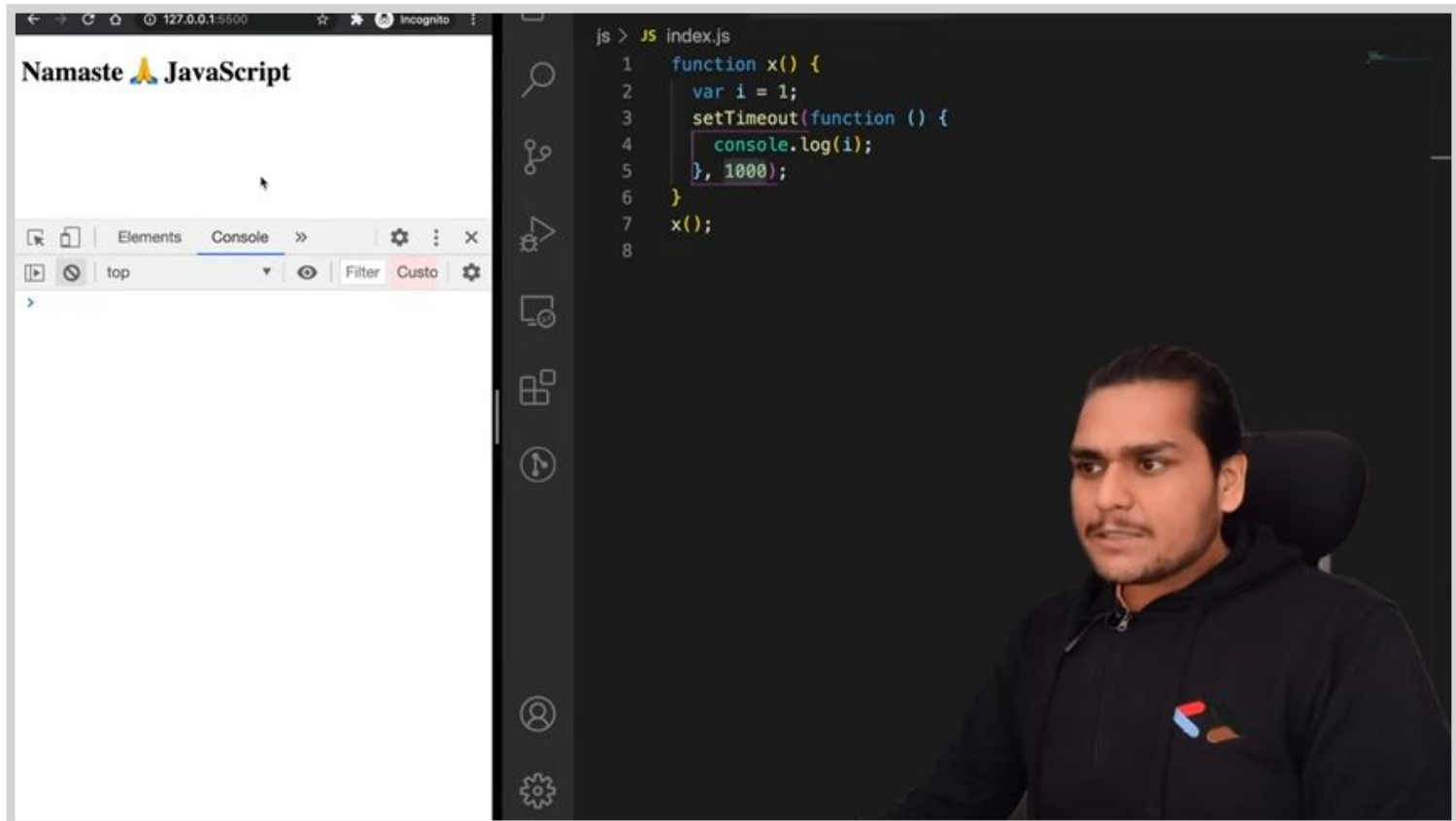


set time out

01:19

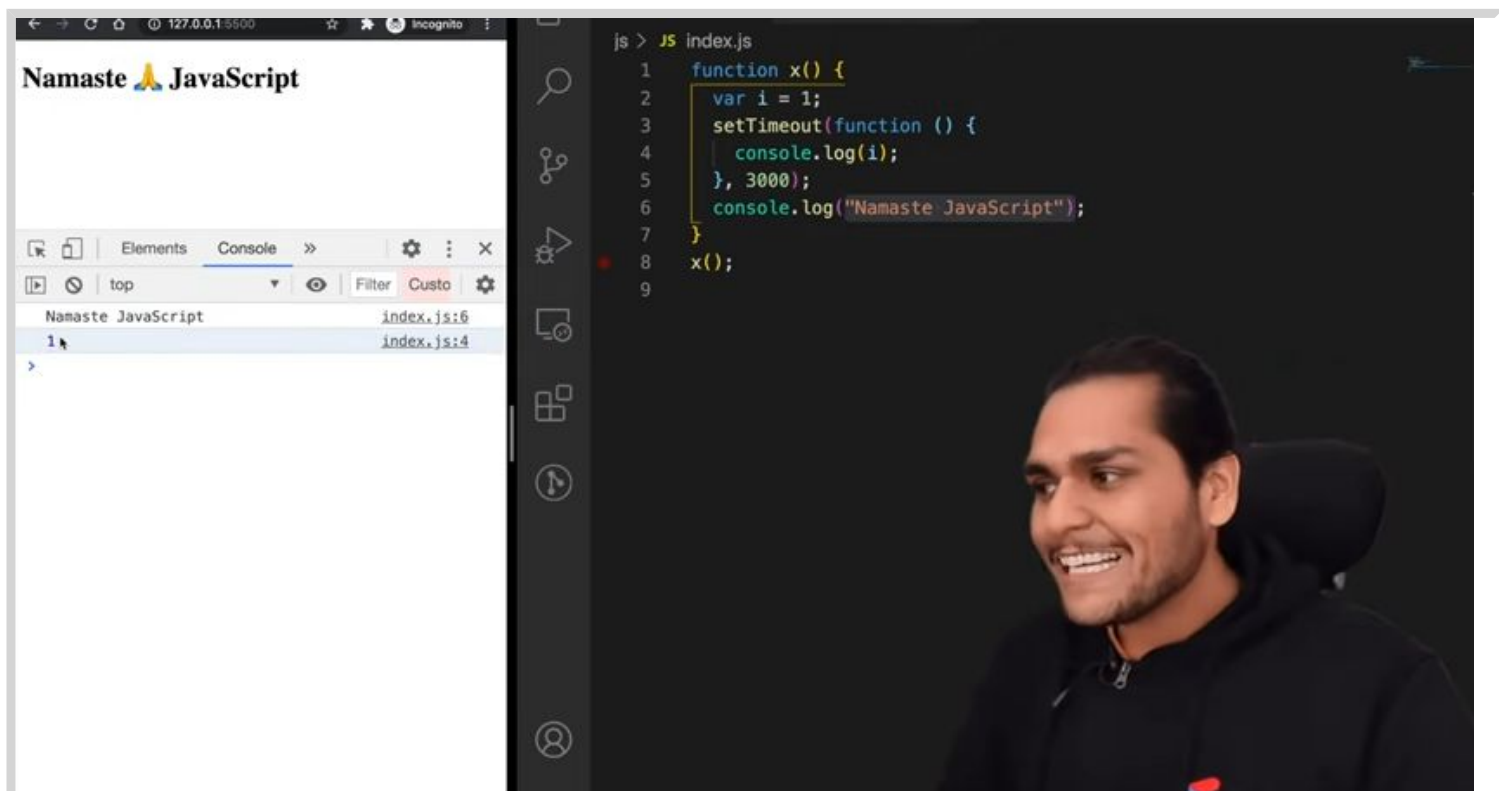


The screenshot shows a web browser on the left and a code editor on the right. The browser displays "Namaste 🙏 JavaScript". The code editor shows the following JavaScript code in `index.js`:

```
js > JS index.js
1 function x() {
2   var i = 1;
3   setTimeout(function () {
4     console.log(i);
5   }, 1000);
6 }
7 x();
8
```

The browser's console is empty, indicating the timeout has not yet executed.

02:17

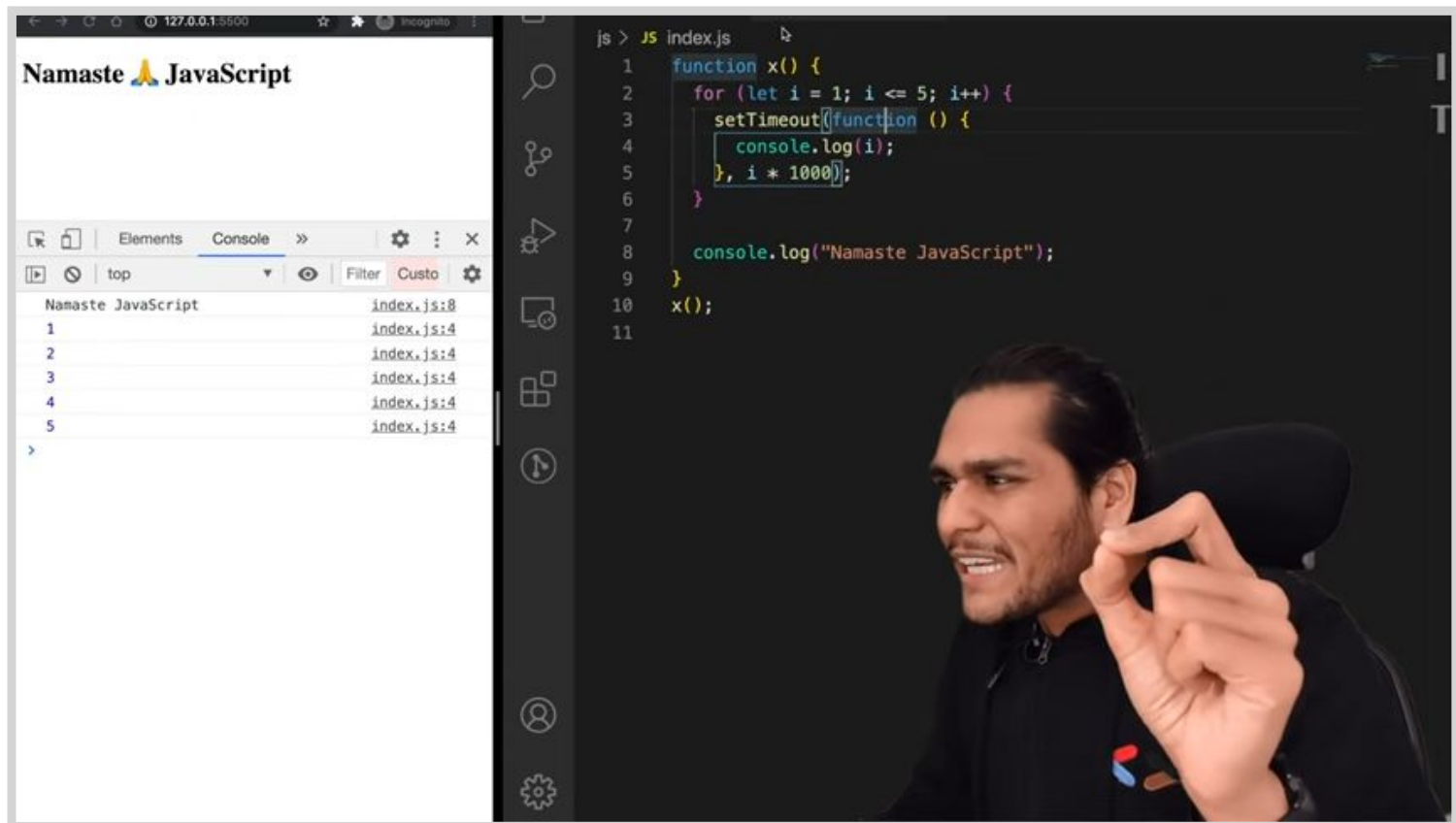


The screenshot shows the same web browser and code editor. The browser's console now displays the output of the timeout function:

```
Namaste JavaScript    index.js:6
1                      index.js:4
```

The code editor shows the same JavaScript code as before, but the `setTimeout` function is now highlighted, indicating it has been executed.

11:12

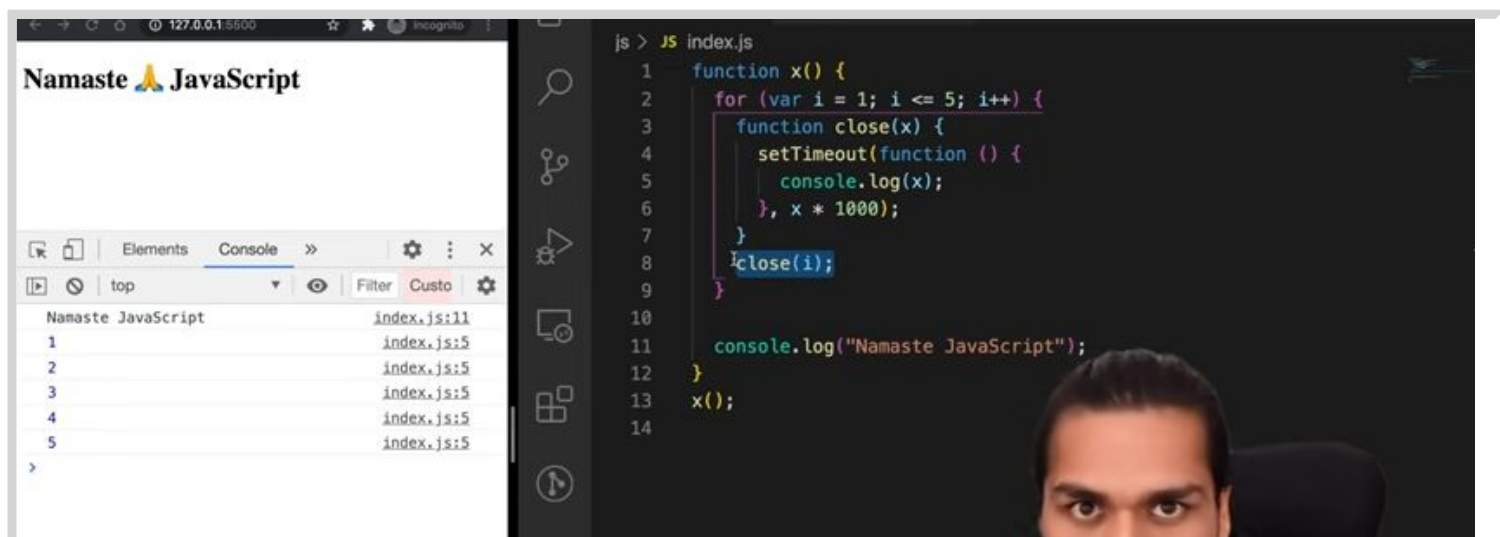


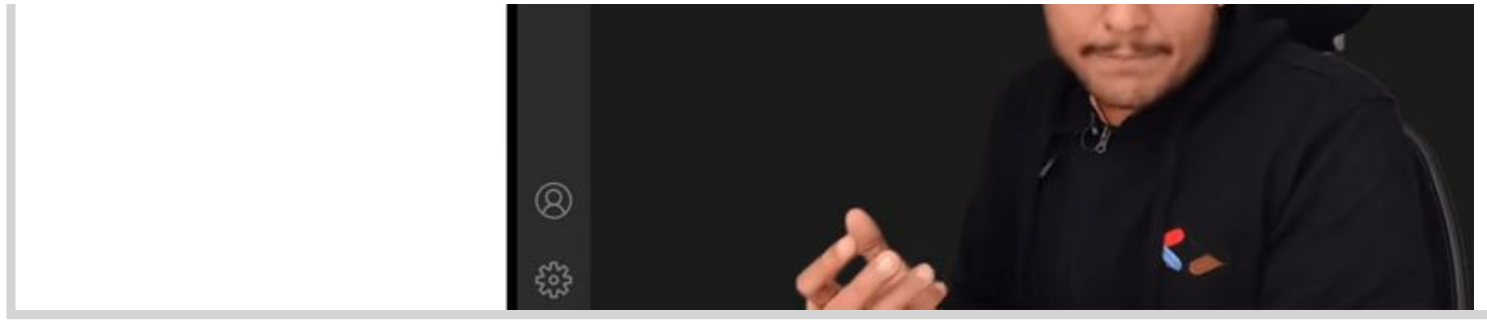
each n every time this function is called it is refering to different memory location.

-why it is working with LETbut not with VAR?

bcoz LETis block scoped & it creates a new copy everytime this for loop is executed.

16:04





By this way , we can use VAR, and pass new memory of i.

