Namaste 🙏 JavaScript

```js
1  function a() {
2      console.log(b);
3  }
4  var b =10;
5  a();
```

10   index.js:2

>

Scope chain is directly linked with lexical environment

2:06 / 19:47 · Code Example for understanding Scope >

127.0.0.1:5500

Inspiring · Knwlg · open · LendingKart · Angular4 · Hackathon · Talk · Uber · Youtube

Namaste 🙏 JavaScript

Elements | Console | Sources | Network | »

top ▼ | Filter | Custom levels ▼

🛑 ▶ Uncaught ReferenceError: b is not defined   index.js:10
    at index.js:10

>

Scope means very you can find a particular variable or a function in a code

```js
JS index.js ✕        JS scope.js        <> index.html

js > JS index.js
1    function a() {
2      var b =10;
3      c();
4      function c(){
5
6      }
7    }
8
9    a();
10   console.log(b);
```

3:35 / 19:47   · Understanding the meaning of Scope >

Call Stack c()

a()

Global EC

Mem	Code
	Get access to the lexical environment of the parent

Mem.	Code
b : 10	
C : { - }	

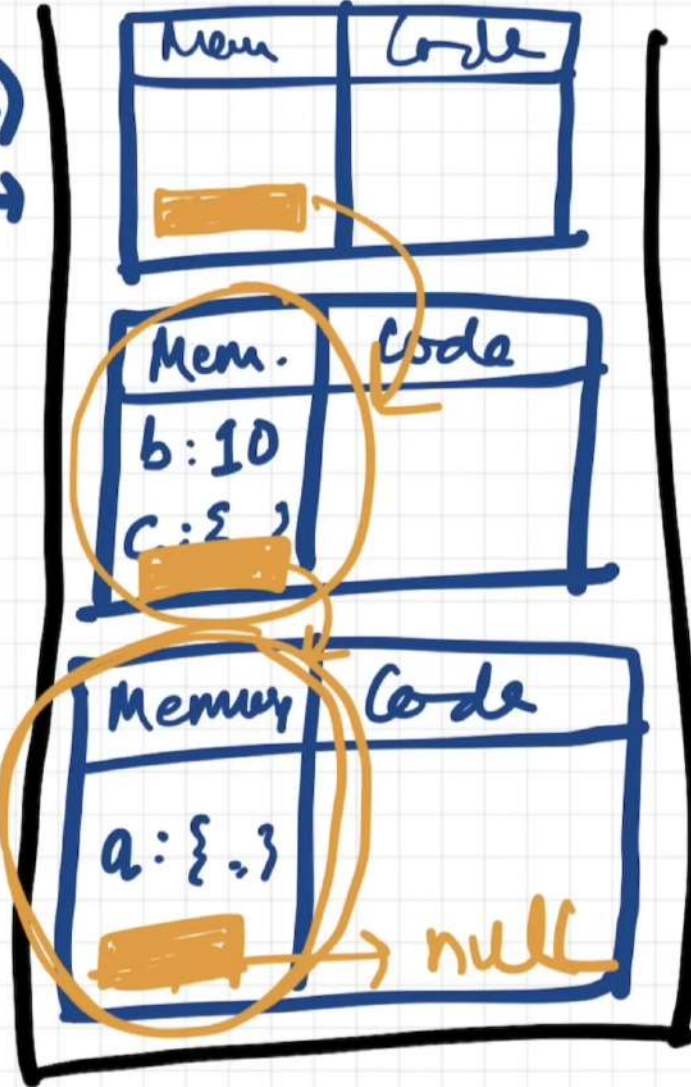| Memory	Code
a : { - }	

```
1  function a() {
2      var b =10;
3      c();
4      function c(){
5
6      }
7  }
8
9  a();
10 consol        );
```

Call Stack (c)

a()

Global EC

Mem	Code

Mem.	Code
b : 10	
c : { }	

Memory	Code
a : { . }	
	→ null

```
1   function a() {
2       var b =10;
3       c();
4       function c(){
5
6       }
7   }
8
9   a();
10  console.lo
```

Call
Stack C()

a()

Global
EC

Mem	Code

Mem.	Code
b : 10	

Memory	Code
a : { }	→ null
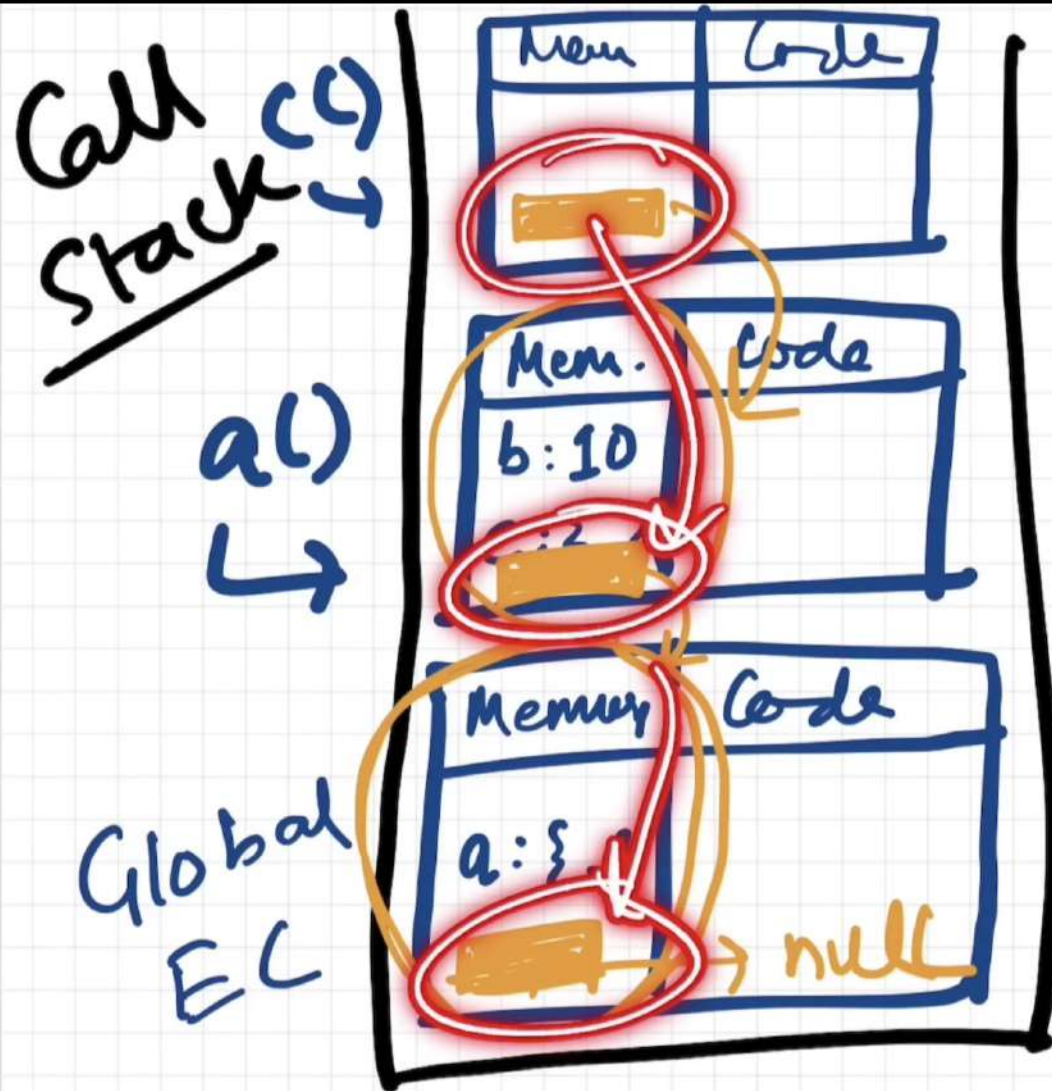
```
1    function a() {
2        var b =10;
3        c();
4        function c(){
5            con.log(b);
6        }
7    }
8
9    a();
10   console.log(b);
```

Scope chain is a chain of all
these lexical environment
and parent references

Namaste 🙏 JavaScript

Paused in debugger ❘▶ ⟳

Elements Console **Sources** Network »

▶ (index) index.js ✕

```javascript
1  function a() {
2    var b =10;
3    c();
4    function c(){
5      console.log(b);
6    }
7  }
8
9  a();
10
```

{} Line 5, Column 5   Coverage: n/a

▶Ɪ ⟳ ⬇ ⬆ ⇥ ⫫ ❚❚   **Scope** Watch

ⓘ **Paused on breakpoint**

▼ Call Stack

▶ c   index.js:5
  a   index.js:3
  (anonymous)   index.js:9

▼ Breakpoints

☑ index.js:5

⋮ Console   What's New   ✕

▶Ꞁ ⊘ | top   ▼ 👁 | Filter   Custom levels ▼   ⚙

>

▼ Local
  ▶ this: Window
▼ Closure (a)
    b: 10
▼ Global   Window
    PERSISTENT: 1
    TEMPORARY: 0
  ▶ a: *f a()*
  ▶ addEventListener: *f addEventListener()*
  ▶ alert: *f alert()*

---

js > JS index.js

```javascript
1  function a() {
2    var b =10;
3    c();
4    function c(){
5      console.log(b);
6    }
7  }
8
9  a();
10
11
12
13
```