

• github: divyaramjan 10



DATE _____

Chap 7: The Scope Chain, Scope & Lexical Environment

"Scope is a space where you can access a specific variable on function."

"Scope is directly dependant on the lexical environment."

"lexical environment is the local memory along with lexical environment of it's parent"

• Code example:

```
function a() {
```

```
  var b = 10;
```

```
  c();
```

```
  function c() {
```

```
  }
```

```
}
```

```
a();
```

```
console.log(b);
```

b()

↳

Memory

Code

71



Memory

Code

b: 10

c: { ... }



Memory

Code

a: { ... }

NULL

Global

EC

7.1



DATE _____

* Red box \rightarrow You can assume this as an reference to the lexical environment of it's parent.
So, you can say that $b()$ can access the variable of $a()$ as shown in the diagram.

"Scope chain is a chain of lexical environ-
-ment and the parent references".

• Summary of this chapter by Torvald Moon:

- 1) lexical environment = EC's local memory + Reference to the lexical environment of it's parent.
- 2) Whenever a new EC is pushed in the Call Stack it holds a reference to the LE of its parent, i.e EC's memory from where it was invoked.
- 3) GEC holds reference to null.
- 4) Transscript engine first looks for the variable / f'' being accessed in local scope of f'' , and if not found, it keeps on searching the LE of it's parent until it finds the variable / f'' being accessed, and this mechanism is called Scope chain.
- 5) If the variable is not found in Scope Chain, then you will get an error "Variable is not defined".