EP. 7

THE FRONTEND INTERVIEW

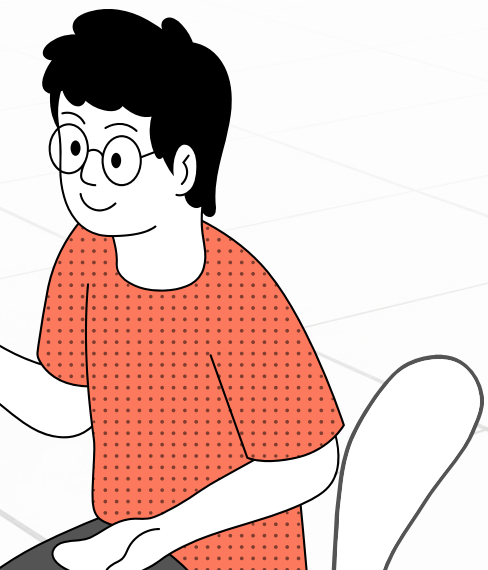
@frontendhabibi

I have written a JS code snippet, you have to tell me the result it will print.

Sure...

@frontendhabibi

```
1  console.log(a);
2  var a = 10;
3  console.log(a);
```



While executing the first line it will print undefined since the variable a has not been initialized, in the second line 10 will be assigned to the variable and the same will be logged to the console in the next line.

Don't you think that it should give an error since we are trying to use the variable before declaring it?

No, in JS all the variables and the function declarations are hoisted before execution of the code so they can be accessed.

Can you please explain, how this hoisting actually happens?

Yup, JS program runs in two phases,
i) Creation of execution context within the call stack and assigning memory space to all the variables with values as undefined and functions as their decalaration.

ii) Line by line code execution.

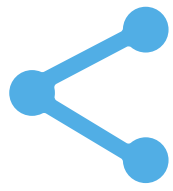
In our code snippet, there is only one variable that will get stored in the memory with the default value as undefined before the code execution starts.
In the first line, the code tries to use that variable so the value from its reference in the memory will be fetched and printed.

In the second line, a new value 10 is being assigned to the variable which will replace the value undefined of the variable and thus prints the newly assigned value when logged again.

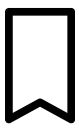
TO BE CONTINUED...

LIKE

COMMENT



SHARE



SAVE FOR FUTURE REFERENCE

@frontendhabibi