

What is call stack?



Call Stack is a data structure for javascript interpreters to keep track of function calls(creates execution context) in the program. It has two major actions,

- 1. Whenever you call a function for its execution, you are pushing it to the stack.**
- 2. Whenever the execution is completed, the function is popped out of the stack.**

Let's take an example and it's state representation in a diagram format-

The screenshot shows a browser developer tools console window. At the top left are three colored dots (red, yellow, green). The file name 'aiarif.js' is displayed at the top right. The console area contains the following code:

```
1 function hungry() {  
2   eatFruits();  
3 }  
4 function eatFruits() {  
5   return "I'm eating fruits";  
6 }  
7  
8 // Invoke the `hungry`  
9 function hungry();  
10
```

The above code processed in a call stack as below,

1. Add the hungry() function to the call stack list and execute the code.
2. Add the eatFruits() function to the call stack list and execute the code.
3. Delete the eatFruits() function from our call stack list.
4. Delete the hungry() function from the call stack list since there are no items anymore.

CallStack

hungry()

CallStack

eatFruits()

hungry()

CallStack

hungry()

CallStack

THANK
YOU!

The word "THANK" is stacked above "YOU!". Each letter is a white outline filled with a unique pattern: T has red diagonal stripes; H has yellow dots; A has blue circles; N has green vertical lines; K has yellow wavy lines. The word "YOU!" is below, with Y having blue dots, O having a green spiral with a small green wavy line at the bottom, U having yellow and green horizontal stripes, and ! having blue diagonal stripes and a red circle with a blue 'X'.

Follow [Ai Arif](#) for More