Developers World
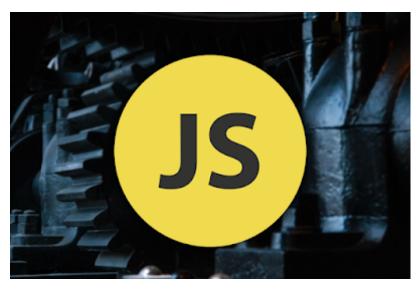
Blogs related to software engineers, developers and programmers.

JavaScript Logical Output Based Interview Questions

- March 31, 2023

You mostly seen in **JavaScript** interviews there interviewers must asks output based questions in which candidate stuck mostly so here in this blog I'm telling you most commonly asked output based JavaScript interview questions.

By Education Funda

Most Commonly Asked JavaScript Output Based Questions

Question: 1

function sum(){

```
    let a = 8;
    const b=2;
    var c=a+b;
}
console.log(a)
// Reference error: a is not defined


console.log(b)
// Reference error: a is not defined


console.log(c)
// Reference error: a is not defined


sum()


console.log(a)
// Reference error: a is not defined


console.log(b)
// Reference error: b is not defined


console.log(c)
//Output: undefined
```

Question: 2

```
let arr=[1,2,3,4,5];


console.log(arr[2], arr.length);
```

//Output: 3 5

arr.length=0;
console.log(arr[2], arr.length);
//Output: undefined 0

Note: As we can see in above code 1st questions is so easy but second is bit tricky one for second we are re-assigning the value of arr.length which logically we can do that's why after assign new value to arr.length our output is undefined for index value and 0 for arr.length.

Question: 3

```
for (var i = 0; i < 3; i++) {
  setTimeout(function() { alert(i); }, 1000 + i);
}
```

//Output: 3 3 3

Note: As we are using **var** so which is a global scope that's why as we are calling setTimeout which itself take time to RUN is call stack so within that time all Loops is finished that's why it takes last value which is 3 in each iteration.

Question: 4

```
for(let i = 0; i <5; i++){
    setTimeout(()=>{
        console.log(i);
   },0)
}
```

//Output: 0 1 2 3 4

Note: It relates to previous questions, as we are using **let** so which is a block scope that's why it keeps increase until loop didn't finished.

Question: 5

```
let count = 0;
(function immediate() {
  if (count === 0) {
    let count = 1;
    console.log(count); // What is logged?
  }
  console.log(count); // What is logged?
})();
```

//Output 1 0

Note: As we know code runs top to bottom so when go inside function count === 0 is TRUE as count variable defined before function called so output will be firstly 1 then 0 for second console.log().

Question: 6

```
console.log(1+false); // Output: 1
console.log(1+true); // Output: 2
console.log(1-false); // Output: 1
console.log(1+'2'-1); // Output: 11
```
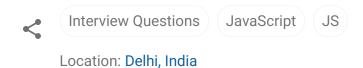

Question: 7

```
(function fnA(a) {
  return (function fnB(b) {
    console.log(a); // What is logged?
  })(1);
})(0);


//Output: 0
```

Note: It is immediate runs function as we are calling into () and as we can see we are calling function inside function which is also a **Closure** concept that's why we are able to get the value of variable b.

Happy Coding !!

Follow Me Personally: Click Here

Interview Questions   JavaScript   JS

Location: Delhi, India

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

Popular posts from this blog

Create and Deploy AWS SAM Application

- May 09, 2023

How To Create and Deploy Your First Lambda Function on AWS Serverless Application Modal AWS SAM - Education Funda Create Your SAM Application Software Requirements: AWS CLI (Reference Link) SAM CLI (Reference Link) Docker (Reference Link) AWS Serverless Architecture - Educa…

READ MORE

MongoDB Mongoose Use Lookup Guide

- May 23, 2023

In This Blog We Will See Complete Guide For Using Joins With Lookup in Mongoose MongoDB Mongoose - Education Funda Simply in MongoDB Mongoose we can use Aggregate Query with $ lookup to get data basis upon JOINS. such as below: await MainModel . aggregate ([{ $match : {…

READ MORE

EDUCATION FUNDA INDIA

VISIT PROFILE

Archive ⌄

Labels ⌄

Report Abuse