?

○ 2
● -1

Ans: B

2.

using which BOM object, we find the laptop battery performance ?

● navigator
○ location
○ history

Ans: A

3.

```
function fn(n1,n2,…data){


}
```

is it valid syntax ?

● Yes
○ No

5.

localStorage data type

● object
○ array

Ans: A

6.

var a=10

var b=6

a/b

○ 4
● 1

Ans: B

7.

In the Array, Using pop method we can ?

```
})
const res=await obj;
}catch(e){
  console.log("catch",e);
}
```

what is output in console ?

● Catch, 100
○ Catch, undefined
○ Catch,20
○ Catch,null

Ans: A

9.

```
const obj=new Promise((a,b)=>{
    a(100)
})
obj
.then((data)=>{
 console.log("then",data)
})
.catch((data)=>{
```

Ans: A

10.

```
const obj=new Promise(function(a,b){
    b(100)
    a(20)
})
const res=await obj;
```

res ?

● 100
○ 20
○ throws exception

Ans: C

11.

```
var obj={n1:10,n2:20}
```

obj has n1 property key or not , how we can check ?

○ obj.get("n1")
● obj.hasOwnProperty("n1")

13.

var arr=[22,33,44,55]

arr.splice(1,1)

arr ?

○ [22,1,33,44,55]
● [22,44,55]

Ans: B

14.

How many ways can get the data from promise object ?

☑ then
☑ await
□ observable

Ans: AB

15.

undefined+0

Ans: A

17.

```
const arr=[22,11,10,33,55]

arr.slice(2,4) ?
```

● [10,33]
○ [11,10]

Ans: A

18.

```
function fn(){
   const res=await <promise object>
}
```

is it valid syntax

○ Yes
● No

Ans: B

Ans: A

20.

```
class A{
 n1=10
 n2=20
 sum(){
   console.log(this.n1+this["n2"]);
 }
}
const obj=new A()

obj.sum()
```

● 30
○ Error

Ans: A

WRITE AGAIN