FRONT END – PART 2

OBJECTS AND TIMING EVENTS

(Collection)

1. Javascript object is collection of -

a) values

b) variables

c) properties

d) none of the above

(Correct Statement(s) About Object)

2. What is true about JavaScript objects?

a) they are mutable

b) they are immutable

c) they are created with curly brackets i.e. { }

d) they are collection of key value pair(s)

e) the values are of same data type

(What is It)

3. Consider the following code snippet -

```
var person = {
    name: "Jack",
    age: 50
};
```

What is 'age: 50' written in the above object 'person'?

a) Value

b) property

c) variable

d) none of the above

(Access Object Value)

4. Which among these is/are the correct way of accessing value of an object using a key(whose name is a valid identifier)?

a) object.key

b) object(key)

c) object["key"]

d) object.(key)

e) object[key]

(Access 3rd Key)

5. An object is given as -

```
var obj = {
    key1: value1,
    key2: value2
}
```

What will be the output when you enter "obj.key3" on the console?

a) value3

b) undefined

c) null

d) Error - key3 is not defined

(Delete Property)

6. Which of the following will delete 'property' of an 'object'?

a) delete object.property

b) del object.property

c) remove object.property

d) none of the above

(Print Keys)

7. Which of the following 'for' loop(s) will print the keys of an object 'obj' containing 'n' properties?

a) for(var i=0; i<n; ++i) { console.log(obj[i]) }

b) for(var i in obj) { console.log(i) }

c) for(var i in obj) { console.log(obj[i]) }

d) none of the above

(Access Nested Values)

8. Which among these is/are the correct way of accessing values of nested objects ?

a) object.key1.key2

b) object.value.value

c) object["key1"]["key2"]

d) object.key1["key2"]

e) object.[key1].key2

f) object.key1.[key2]

(Array of Keys)

9. Which of the following will return an array of keys of an object 'vehicle'?

a) vehicle.getKeys();

b) vehicle.getOwnPropertyNames();

c) Object.keys(vehicle);

d) Object.getOwnPropertyNames(vehicle);

(Run Function)

10. Which of the method will run a function 'abc' after 5 seconds?

a) setTimeOut(abc, 5000);

b) window.setTimeout(abc, 5000);

c) Function.setTimeout(abc, 5000);

d) none of the above

(Repeat Execution)

11. Which of the following will repeat execution of a function 'rotate'?

a) setTimeout(rotate, 1000);

b) setTimeout(rotate);

c) setInterval(rotate);

d) setIntervalTime(rotate, 1000);

e) setInterval(rotate, 1000);

(Stop Execution)

12. Which of the following method is used to stop the execution of the below code?

```
setInterval(functionName, 1000);
```

a) clearInterval(functionName);

b) clearInterval(functionName, 1000);

c) the execution cannot be stopped

d) none of the above

(Spread Opertaor)

13. Consider the following object. What happens when we spread into a new object and change the firstName property of the object?

```
let obj1 = { firstName: 'James' };
let obj2 = {...obj1};
obj2.firstName = 'John';
```

```
    console.log(obj1);
```

a) { firstName: "John" }

b) { firstName: "James" }

c) Type Error

d) None of these

(Purpose of map())

14. The main purpose of the array map() function is that-

a) It maps the elements of another array into itself.

b) It passes each element of the array and returns the necessary mapped elements.

c) It passes each element of the array on which it is invoked to the function you specify and returns an array containing the values returned by that function.

d) It passes the elements of the array into another array.

(Find Output)

15. What will be the output of the following code ?

```
    var array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15];
    var myArr= array.filter(v => v % 3 === 0);
    console.log(myArr);
```

a) myArr

b) [3, 6, 9, 12, 15]

c) [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

d) [1, 2, 4, 5, 7, 8, 10, 11, 13, 14]

ASSIGNMENT

(Create Empty Object)

16. Which of the following will create an empty object?

a) var obj

b) var obj = {}

c) obj = new Object()

d) none of the above

(An Empty Object)

17. Let’s say that you created an empty object 'obj'? What will the below statements print on the console?

```
    obj.key1 = "value1";
    console.log(obj.key1);
```

a) undefined

b) null

c) value1

d) Error - key1 does not exist

(Find the Output)

18. Consider following code snippet -

```
    var student = {
        firstName: "jonathan",
        lastName: "doe"
    }
```

```
delete student.firstName;
console.log(student.firstName+ " "+ student.lastName);
```

What is the expected output?

a) jonathan doe b) undefined doe c) doe d) Error

(Function With Variable Argument)

19. What is the correct way of declaring a function with variable arguments?

a) function functionName(args . . .) b) function functionName(. . . args)

c) function functionName(args) d) none of the above

(Print on Console)

20. What will the following code print on the console?

```
function abc(arr) {
    var arr2;
    for(i in arr) {
        arr2[i] = i;
    }
    return arr2;
}

var arr = [12, 56, 48, "hello"];
console.log(abc(arr)[1]);
```

a) 56 b) 1 c) 12 d) error

(Three Objects)

21. What will the below JavaScript lines print on the console?

```
var obj1 = {};
var obj2 = new Object();
var obj3 = {};
console.log(obj1==obj2, obj1 == obj3);
```

a) true true b) true false c) false false d) false true

(What is Printed)

22. What will the below code do -

```
var abc = setInterval(counting, 300);
var count = 0;

function counting() {
    console.log(count++);
    if(count == 5) {
        clearInterval(abc);
    }
}
```

a) Print numbers from 1 to 5

b) Print numbers from 0 to 4

c) Print numbers from 0 to 5

e) Print numbers from 1 to 4

e) Infinitely print numbers starting from 0

(Find the Output – III)

23. What will be the output of the following code:

```
let list = [4, 5, 6];
for (let i in list) {
   console.log(i);
}

for (let i of list) {
   console.log(i);
}
```

a) 0, 1, 2 and 4, 5, 6

b) 4, 5, 6 and 0,1, 2

c) 1, 2, 3 and 4,5,6

d) None of these

(Guess the output)

24. Find the output of the code:

```
setTimeout(function () {
  console.log("a")
},20)
setTimeout(function () {
  console.log("b")
})
console.log("c")
```

a) a c b

b) c a b

c) c b a

d) b c a

(Output 2)

25. What will be the output of the following code?

```
var arr = []

arr.push("hello")
arr.hello = 0
var value = 0
for(var i = 0;i < 5; i++) {
    value = arr[value]
}
console.log(value)
```

a) 0

b) hello

c) Undefined

d) It will give a reference error

(setTimeout())

26. Consider the following function:

```
function add(a,b,c,d) {
  console.log(a+b+c+d)
}
```

If we need to pass the function to setTimeout() that will run after 10 milliseconds, how many arguments will setTimeout take?

a) 4 b) 6 c) 2 d) 1

(pass by ref)

27. What would be the output of the following code?

```
function change1(obj) {
  obj.name = "ninjas"
}
function change2(obj) {
  obj = {name: "changed"}
}

var obj = {name:"coding"}

change1(obj)
change2(obj)
console.log(obj.name)
```

a) coding b) ninjas c) changed

(copying)

28. What would be the output of the following code?

```
var d = {h: 2, j: 3}

var e = d
var f = {...d}

console.log(e === d)
console.log(f === d)
```

a) true false b) false true c) true true d) false false

(copying 2)

29. what would be the output of the following code?

```
var d = {h: 2, j: 3}

var e = d
var f = {...d, j: 4}

console.log(f.h)
console.log(f.j)
```

a) 2 3 b) 3 4 c) 2 4 d) 4 2

****