There would be negative marking on all objective type questions. -1 for wrong and +1 for right.

1. Output-

```
console.log(a);
console.log(b);
var b=8;
console.log(b);
```

a- undefined, 8, 8
b- null, reference error, 8
c- null, undefined, 8
d- reference error, undefined, 8

2- var obj = { “hasOwnProperty”:8, “a”:6, “b”:8 }
How will you use native hasOwnProperty method of JS Object on obj?

3. Implement such a function in one statement without using if-else, switch or ternary operator-

```
    func(3)   =>  7
    func(7)   => 3
```

4. Differentiate between call, apply and bind methods by giving suitable examples.

```
5. function parent(id){
        this.id = id
}
function child(name){
        this.name = name;
}
var obj = new child(6, ‘john’);
```

correct the code in such a way, that id and name can be called using obj.

```
6. <a href=”xyz.html” data-ng-click=”onAnchorClick($event)”>
$scope.onAnchorClick = function(e){
        console.log(‘anchor clicked, event: ’+ e);
}
```

Correct the code in such a way(without changing anchor tag) that it should not open xyz.html.

7. you are given a sorted array, you need sort on their magnitude values in order n complexity.

eg- input [-8, -6, -3 , 2, 45, 80, 127]
        output [2, -3, -6, -8, 45, 80, 127]

```
8.  22+ +’3’ + ‘567’
```

a- “223567”

b- 592

c- "25567"

d- 25567

e- error

```
9. for(var i=0; i<10; i++){
        setInterval(function(){
                console.log(i);
        }, 500);
}
```

a- 0,1,2,3,4,5,6,7,8,9

b- 0,1,2,3,4,5,6,7,8,9,0,1,2,...... upto stack completes

c- random output upto stack completes

d- 10, 10, 10,.... upto stack completes

e- none of the above

f- 0,0,0,0,0,0,......

```
10.  var obj={
        “name”: “john",
        “getName” : function(){
                console.log(this.name);
        }
}
var func = obj.getName;
console.log(func());
```

a- reference error

b- john

c- window object

d- undefined

e- none of the above

```
11. var a = {};
var b = {}, c= {};
c[a] = ‘I am good’;
c[b] = ‘I am a fool’;
console.log(c[a]);
c[a]=’I am awesome’;
console.log(c[b]);
```

a. I am a fool I am awesome

b. I am awesome I am good
c. I am good I am fool
d. I am fool I am good
e. Uncaught reference error

12- As `[]` is true, `[]==true` should also be true. right? Give reason.

13- Write a function(code) which will work properly when invoked with either syntax

```
console.log(sum(2,3)); //Output 5
console.log(sum(2)(3)); // Output 5
```

14. What will be the output of below code snippet ?

```
(function() {
var a = b = 10;
})();
console.log(typeof a);
console.log(typeof b);
```

15.

```
(function() {
console.log(1);
setTimeout(function(){console.log(2)}, 1000);
setTimeout(function(){console.log(3)}, 0);
console.log(4);
})();
```

a. 1,2,3,4
b. 1,3,2,4
c. 1,3,4,2
d. 1,4,3,2

16.

```
if ([0]) {
  console.log([0] == true);
  console.log(!![0]);
}
```

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

17. Write a native method of an Array named, ‘getFlatArray’ which will take Inputarray and return Outputarray:

Example:
Inputarray =[1,[2,3],[4,[5,6]]];
Outputarray=[1,2,3,4,5,6];

18. What is box model CSS?

```
19. if (‘102’ > ‘101’) {
            console.log(‘102 is greater’);
            } else {
            console.log(‘101 is greater’);
            }
            if (‘99’ > ‘101’) {
            console.log(‘99 is greater’);
            } else {
            console.log(‘101 is greater’);
}
```
then, what will be the result of console.log and why?

20. Which statement returns whether the regular expression is matched or not
var patt='/e/';
a). patt.equals('This is string');
b). patt.match('This is string');
c). patt.test('This is string');
d). patt.verify('This is string');

21-Does ng-repeat re-render the dom if scope value changes?

a- Yes
b- No

22- Write a code snippet which signifies JS is a single thread language.

23- write a function in such a way, which can take n number of arguments and returns the sum of all number type arguments.
eg- sum(3,4,7,2,’aa’, 9) => 25

24- How can we do display none in angularJS-
a- ng-if
b- ng-show
c- ng-include
d- ng-app
e- none of the above

25- What is the best practice to let angular know about the changes done by some external JS-
a- $apply
b- $timeout
c- $digest
d- none of the above

26- If we want to do some dom manipulation in a angular directive, what is the best practice to use-
a- controller
b- compile
c- link
d- none of the above

27- we can de-register the event handlers in angularJS using-
a- $destroy
b- on $locationChangeSuccess
c- on $stateChangeSuccess
d- on controller load
e- none of the above

28- Consider that we have jQuery on page, what will be the output-

```
  $('a') === document.querySelector('a')
```

a- true
b- false

29- Write a code snippet which creates a memory leak.

30- function example(){

```
function example(){
        this.name="John";
}
var obj = new example();
```

We can get class name 'example' from object 'obj' using-

a- obj.name
b- obj.constructor
c- obj.constructor.name
d- obj.prototype.name
d- obj.__proto__.name