

Soujan

L1

L1	L2	L3
12	22	9

43

✕ 1) What will be the output of the following code?

```
1. if (function f()){}) {
2.   alert(typeof f);
3. }
```

- ☐ undefined
- ☒ function
- ☐ null
- ☐ object
- ☐ there is an error in the code

L1 - 12
 L2 - 22
 L3 - 9

 43

✕ 2) What will be the result of this example?

```
function f() {
  let a = 5;
  return new Function('b', 'return a + b');
}
```

```
alert( f()(1) );
```

- ☐ 1
- ☒ 6
- ☐ undefined
- ☐ NaN
- ☐ there will be an error

✕ 3) What will be the output of the following code?

```
let y = 1;
let x = y = 2;
```

```
alert(x);
```

- ☐ 1
- ☐ 2
- ☐ NaN
- ☐ undefined
- ☒ the error will occur

✓ 4) What the alert will output?

```
alert(str); // ?  
var str = "Hello";
```

- ☐ Hello
- ✓ ☒ undefined
- ☐ there will be an error

✗ 5) Attentively look at the code syntax. What will be the output?

```
let f = function(x) {  
  alert(x)  
}
```

```
(function() {  
  f(1)  
})();
```

- ☐ no result
- ✓ ☒ output 1
- ☐ output undefined
- ☐ there will be an error

✓ 6) Under what condition `elem.scrollHeight == elem.clientHeight` ?

- ☐ element content completely scrolled down
- ☐ page is scrolled so that element is fully visible and is in the border of the window
- ☐ the element height is equal to scroll bar height
- ✓ ☒ element context is completely visible, has no scroll
- ☐ the element has no padding

✓ 7) How can we get full height of block DOM-element, if it has border and padding?

- ☐ `parseInt(elem.style.width)`
- ☐ `parseInt(getComputedStyle(elem).width)`
- ✓ ☒ `elem.offsetWidth`
- ☐ `elem.clientWidth`

8) Here is the button `<button id="elem"></button>`, how can we show the string "`<hello>`"?

The result has to be like this:

```
<!--+ run -->
```

```
<button id="elem">&lt;hello&gt;</button>
```

- ✓ `elem.innerHTML = "<hello>"`
- `elem.innerText = "<hello>"`
- `elem.textContent = "<hello>"`
- `elem.setText("<hello>")`

9) Link `` has event listener:

```
link.addEventListener('click', function(event) {  
    // ...  
});
```

What the code event listener should consist to prevent redirection on click?

If several of the listed above are suitable, select several

- `event.stopPropagation()`
- `event.stopImmediatePropagation()`
- ✓ `event.preventDefault()`
- `return false`
- `event.cancel()`

10) How can we get Dom-element form in the document?

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<body>
```

```
    <form name="myform">...</form>
```

```
</body>
```

```
</html>
```

- `document.forms.myform`
- `document.getElementById("myform")`
- `document.body.children[0]`
- ✓ `document.body.firstChild`
- `document.forms[0]`

14) How many children
<div id="div">
<script>
div.in
div

11) Calling `setInterval(func, 100)` calls func...

- ☒ exactly every 100ms.
- ☐ approximately every 100ms.
- ☐ usually every 100ms, but possibly calls will happen much less often

12) This code will not output anything?

```
document.onclick = function(event) {  
    alert(event.type);  
};
```

```
document.body.dispatchEvent(new CustomEvent('click'));
```

- ☐ you are not allowed to generate built-in events in JavaScript
- ☐ listener through on-property does not work for event generated from JavaScript
- ☒ built-in event has to be generated otherwise, constructor `CustomEvent` should be replaced with something
- ☐ when generating event it is not specified that it should bubble
- ☐ OMG, what is it?

13) Here is the link:

```
<a href="http://site.com">Link text</a>
```

document has listener:

```
document.onclick = function(e) { e.preventDefault(); };
```

Will the pass on link happen?

- ☐ yes
- ☒ no
- ☐ depends on position of stars at the click moment

14) How many children will <div> have after the code below?

```
<div id="div">text</div>
```

```
<script>
```

```
div.insertAdjacentHTML('afterBegin', '<p>Hello</p>');
```

```
div.insertAdjacentHTML('beforeEnd', '<p>Hello</p>');
```

```
</script>
```

- ☒ 0
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ the error will occur

15) In which cases for element elem is true `elem.firstChild == elem.lastChild`?

- ☐ when elem has no children
- ☒ when elem is the only one descendant
- ☐ when elem has one two or more descendants

16) Which events from the list will occur when you press and release the 'space' button?
Another buttons are not clickable.

- ☐ keydown
- ☒ keyup
- ☐ keypress
- ☐ no one will occur

17) What is the right way to output greeting in 5 seconds?

- ☐ `sleep(5); alert("Hello!");`
- ☐ `sleep(5000); alert("Hello!");`
- ☒ `setTimeout(function() {alert("Hello!");}, 5000);`
- ☐ `setTimeout(function() {alert("Hello!");}, 5);`

18) How can we get the indent size margin-top of <div>?

- ☐ div.style.marginTop
- ☒ div.style["margin-top"] .
- ☐ getComputedStyle(div, "margin").top
- ☐ getComputedStyle(div).marginTop
- ☐ getComputedStyle(div).margin.top

19) Can the script connect another outer js-files to the page while it is working?

- ☐ yes, but only once
- ☒ yes, but only before the page is completely loaded
- ☐ yes, as many files are needed and whenever it needed.

20) There is an element <div id="elem"><input/></div>. It has two handlers:

```
elem.addEventListener("click", handler1, false);  
elem.addEventListener("click", handler2, true);
```

Which of them will work earlier while clicking <input>?

- ☐ the first
- ☐ the second
- ☐ no one
- ☒ the standard does not specify the exact sequence

2

21) How many children (DOM-elements) has element in the example below:

```
<ul>  
  <li>Hello</li>  
  <li>World</li>  
</ul>
```

- ☐ 1
- ☒ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6

L2

22) What would the following lines of code output to the console?

```
function parent() {  
  var hoisted = "I'm a variable";  
  function hoisted() {  
    return "I'm a function";  
  }  
  return hoisted();  
}  
console.log(parent());
```

"I'm a function"

23) What will the code print?

```
function foo() {  
  function bar() {  
    return 3;  
  }  
  return bar();  
  function bar() {  
    return 8;  
  }  
}  
console.log(foo());
```

*function bar() {
 return 3;
}*

24) What will the following code output to the console:

```
var a = 1;
```

```
function b() {
```

```
  a = 10;
```

```
  return;
```

```
  function a() {}
```

```
}
```

```
b();
```

```
console.log(a);
```

10

25) What will the code print?

```
alert(foo());
```

```
function foo() {
```

```
  var bar = function() {
```

```
    return 3;
```

```
  };
```

```
  return bar();
```

```
  var bar = function() {
```

```
    return 8;
```

```
  };
```

```
}
```

Error

26) Consider the code snippet below. What will the console output be?

```
var myVar = 'foo';
```

```
(function() {
```

```
  console.log('Original value was: ' + myVar);
```

```
  var myVar = 'bar';
```

```
  console.log('New value is: ' + myVar);
```

```
})();
```

undefined

bar

27) What will the code below output to the console?

```
var myObject = {  
  foo: "bar",  
  func: function() {  
    var self = this;  
    console.log("outer func: this.foo = " + this.foo);  
    console.log("outer func: self.foo = " + self.foo);  
    (function() {  
      console.log("outer func: this.foo = " + this.foo);  
      console.log("outer func: self.foo = " + self.foo);  
    })();  
  }  
};  
myobject.func();
```

bar
bar

Error

28) What will the code below output?

```
console.log(0.1 + 0.2);  
console.log(0.1 + 0.2 == 0.3);
```

0.3

True

29) In what order will the numbers 1-4 be logged to the console when the code below is executed?

⇒ 1 3 4 2

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

30) Assuming d is an "empty" object in scope, say:

var d = {};

...what is d array?

```
['zebra', 'horse'].forEach(function(k) {  
  d[k] = undefined;  
});
```

d = ['zebra': undefined,
'horse': undefined]

31) What will the code below output to the console?

```
var arr1 = "john".split('');  
var arr2 = arr1.reverse();  
var arr3 = "jones".split('');  
arr2.push(arr3);  
console.log("array 1: length = " + arr1.length + " last = " + arr1.slice(-1));  
console.log("array 2: length = " + arr2.length + " last = " + arr1.slice(-1));
```

array 1: length = 4 last = n
array 2: length = 9 last = n

32) What will the code below output of the following code:

```
for (var i = 0; i < 5; i++) {  
    setTimeout(function() {console.log(i); } i * 1000);  
}
```

⇒ 5 5 5 5 5

33) What would the following lines of code output to the console?

```
console.log("0 || 1 = " + (0 || 1)); False  
console.log("1 || 2 = " + (1 || 2)); True  
console.log("0 && 1 = " + (0 && 1)); False  
console.log("1 && 2 = " + (1 && 2)); True
```

34) What will be the output out of the following code?

```
var a={},  
    b={key: 'b'},  
    c={key: 'c'};
```

```
a[b]=123;
```

```
a[c]=456;
```

```
console.log(a[b]); 123
```

35) What will be the output out of the following code?

```
console.log((function f(n){return (n > 1) ? n * f(n-1) : n})(10));
```

Error

36) Consider the code snippet below. What will the console output be?

```
(function(x) {  
  return (function(y) {  
    console.log(x);  
  })(2);  
})(1);
```

⇒ 1

37) What will the following code output to the console:

```
var hero = {  
  _name: 'John Doe',  
  getSecretIdentity: function () {  
    return this._name;  
  }  
};
```

```
var stoleSecretIdentity = hero.getSecretIdentity;
```

```
console.log(getSecretIdentity());  
console.log(hero.getSecretIdentity());
```

~~John Doe~~

John Doe

John Doe

38) What will be the output of this code?

```
var x = 21;  
var girl = function () {  
  console.log(x);  
  var x = 20;  
};  
girl();
```

undefined

39) What will the code print?

```
for (let i=0; i<5; i++) {  
  setTimeout(function() { console.log(i); }, i* 1000);  
}
```

\Rightarrow 0 2 3 4

40) What do the following lines output?

```
console.log(1 < 2 < 3); True True  
console.log(3 > 2 > 1); False False
```

41) Imagine you have this code:

```
var a = [1, 2, 3];
```

a) Will this result in a crash? *No*

```
a[10] = 99;
```

b) What will this output? *undefined*

```
console.log(a[6]);
```

42) What would following code return?

```
console.log(typeof typeof 1);
```

Number

L3

43) What values resulting array will consist of?

`[1,2,3,4,5,6].filter((v) => !!v % 2);`

=> [1, 0, 1, 0, 1, 0]

44) What outcome of the following code?

```
let arr = [];  
while (arr.length < 5) {  
  if (arr.length % 2) {  
    arr.push(arr.length * 2);  
  } else {  
    arr.shift(arr.length);  
  }  
}
```

*=> Goes into infinite loop
leads to an exception or error*

45) What will be outputted to console by the following code?

```
const input = [6,3,2,5,1];  
console.log(parseInt(input.reduce(function (mem, val, ind) {return mem[val] =  
ind;}, []))));
```

46) What values resulting array will consist of?

```
Array.prototype.map.call('Ericsson Project', (v, i) => i).filter((i, v) => v  
% 2);
```

47) What will be outputted to console by the following code?

```
var arr = [];  
while (arr.length < 10) {  
  if (arr.length % 2) {  
    arr.push(arr.length);  
  } else {  
    arr.unshift(arr.length);  
  }  
}  
console.log(arr);
```

[8, 6, 4, 2, 0, 1, 3, 5, 7, 9]

48) What will be outputted to console by the following code?

```
const res = [3,4,7,2,1].reduce(function (mem, val, ind) {return (mem[val] =  
ind, mem);}, []);  
console.log(parseInt(res));
```

49) What values resulting array will consist of?

```
Array.prototype.map.call('Ericsson Project', (v, i) => i).filter((i, v, a) =>  
(a.length - v) % 2);
```

50) What result output of the following code?

```
'Javascript'.split('a').map(function (v) {  
  if (v.length <= 1) {  
    return v + 'a';  
  }  
}).join(''); => Javascript
```

51) What values resulting array will consist of?

```
[18, 3, 15, 4, 3, -3, 1, 0, -5, 8, 13].sort(function (a1, a2) {  
  return (Math.abs(a1) < Math.abs(a2)) ? 1 : (a1 === a2 ? 0 : -1);  
});
```

=> [0, 1, 3, 3, -3, 4, -5, 8, 13, 15, 18]

52) What will be outputted to console by the following code?

```
const arr = [13, 8, 2, 11, 5, 1, 9, 6, 1, 1],  
  init = Array([5]),  
  result = arr.reduce(function (mem, val, ind) {  
    mem[val] = ind;  
    return mem;  
  }, init);  
console.log(parseInt(result));
```


53) What values resulting array will consist of?

```
Array.prototype.map.call('Ericsson Project', (v, i, a) => a.length -  
i).filter((i, v) => v % 2);
```

54) What values resulting array will consist of?

```
[[0,0], [0,1], [1,0], [1,1]].map(function(d,i,grid){  
  const side = Math.sqrt(grid.length);  
  return [Math.abs(i % side - side+1), Math.floor(i/side)];  
});
```

$\Rightarrow [-3, 0], [-3, 0], [-2, 0], [-2, 0]$

55) What will be outputted to console by the following code?

```
const red = [2, 5, 6, 1, 4, 1]  
  .reduce(function (mem, val, ind) {  
    return (mem[val] = ind, mem);  
  }, Array(5)),  
res = parseInt(red);  
console.log(red);
```

56) What is arr final state will be?

```
for (let arr = []; arr.length < 10;) {  
  if (arr.length % 2) {  
    arr[arr.length] = arr.length;  
  } else {  
    for (var aBI = arr.length; aBI > 0; aBI--) {  
      arr[aBI] = arr[aBI - 1];  
    }  
    arr[0] = arr.length;  
  }  
}
```

$\Rightarrow [0, 0, 0, 0, 0, 1, 3, 5, 7, 9]$

Answers

1. (1 point) object
2. (1 point) there will be an error
3. (1 point) 2
4. (1 point) undefined
5. (1 point) there will be an error
6. (1 point) element context is completely visible, has no scroll
7. (1 point) elem.offsetWidth
8. (1 point) elem.textContent = "<hello>"
9. (1 point) event.preventDefault() and return false
10. (1 point) document.forms.myform, document.body.children[0], document.forms[0], document.body.firstChild
11. (1 point) usually every 100ms, but possibly calls will happen much less often
12. (1 point) built-in event has to be generated otherwise, constructor CustomEvent should be replaced with something
13. (1 point) no
14. (1 point) 2
15. (1 point) when elem has no children and when elem is the only one descendant
16. (1 point) keydown, keyup, keypress
17. (1 point) setTimeout(function() {alert("Hello!");}, 5000);
18. (1 point) getComputedStyle(div).marginTop
19. (3 points) yes, as many files are needed and whenever it needed
20. (1 point) the second
21. (2 points) 2
22. (3 points) TypeError: hoisted is not a function
23. (2 points) 8
24. (3 points) 1
25. (2 points) 3
26. (2 points) 'Original value was: undefined', 'new value is: bar'
27. (2 points)
outer func: this.foo = bar
outer func: self.foo = bar
outer func: this.foo = undefined
outer func: self.foo = bar
28. (3 points)
0.30000000000000004
false
29. (2 points)
1
4
3
2

30. (2 points) {zebra: undefined, horse: undefined}
31. (4 points)
 "array 1: length=5 last=j,o,n,e,s"
 "array 2: length=5 last=j,o,n,e,s"
32. (2 points) 5 5 5 5 5
33. (2 points)
 0 || 1 = 1
 1 || 2 = 1
 0 && 1 = 0
 1 && 2 = 2
34. (5 points) 456
35. (2 points) 3,628,800
36. (2 points) 1
37. (3 points)
 undefined
 John Doe
38. (3 points) undefined
39. (2 points) 0 1 2 3 4
40. (2 points)
 true
 false
41. (4 points)
 a) It will not crash
 b) undefined
42. (4 points) string
43. (5 points) [1,2,3,4,5,6]
44. (4 points) infinite loop
45. (4 points) 4
46. (4 points) [1, 3, 5, 7, 9, 11, 13, 15]
47. (5 points) [8, 6, 4, 2, 0, 1, 3, 5, 7, 9]
48. (4 points) NaN
49. (4 points) [1, 3, 5, 7, 9, 11, 13, 15]
50. (4 points) Java
51. (5 points) [18, 15, 13, 8, -5, 4, 3, 3, -3, 1, 0]
52. (4 points) 5
53. (5 points) [15, 13, 11, 9, 7, 5, 3, 1]
54. (6 points) [[0,1],[1,1],[0,0],[1,0]]
55. (6 points) [empty, 5, 0, empty, 4, 1, 2]
56. (7 points) [7, 5, 3, 0, 1, 3, 5, 7, 9]