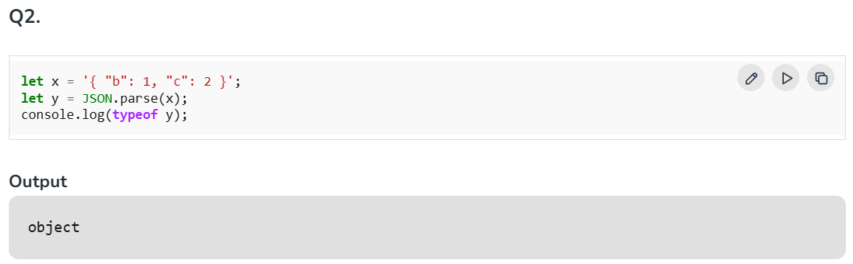
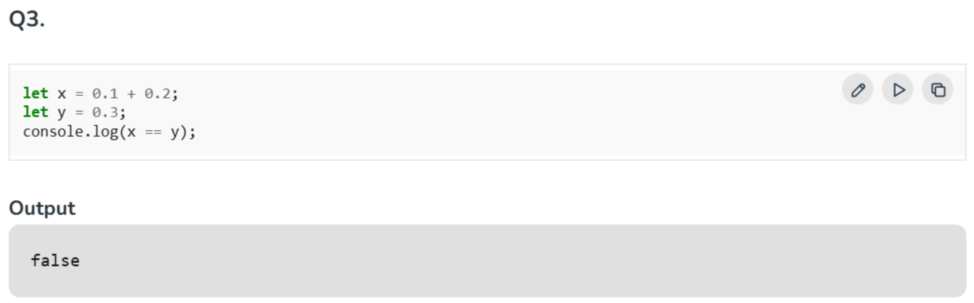


Q2.

```
let x = '{ "b": 1, "c": 2 }';
let y = JSON.parse(x);
console.log(typeof y);
```

Output

```
object
```

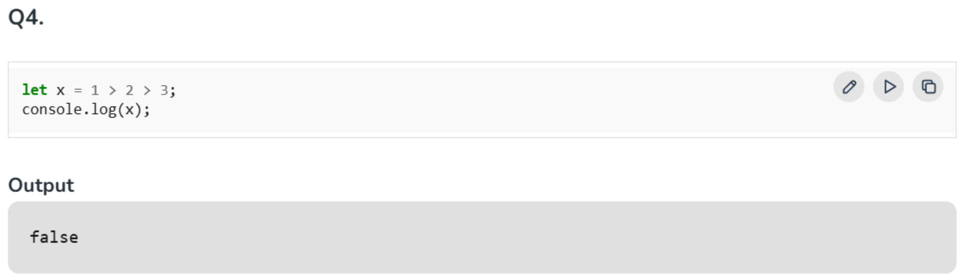


Q3.

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

Output

```
false
```

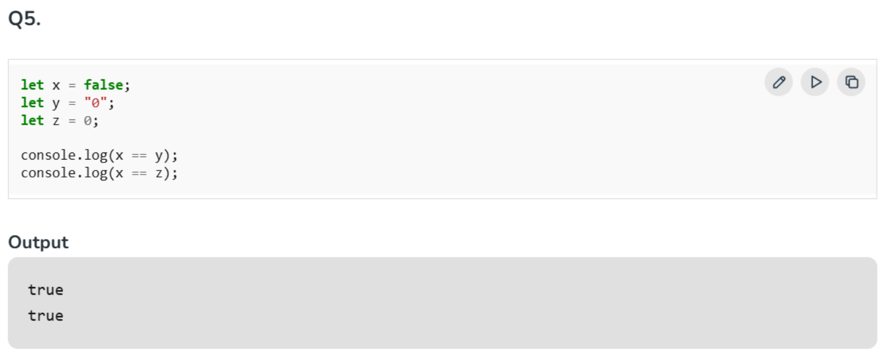


Q4.

```
let x = 1 > 2 > 3;
console.log(x);
```

Output

```
false
```



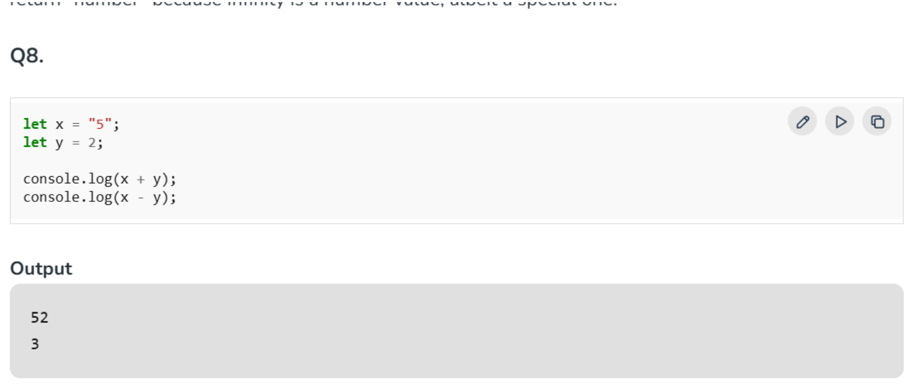
Q5.

```
let x = false;
let y = "0";
let z = 0;

console.log(x == y);
console.log(x == z);
```

Output

```
true
true
```



return "number" because Infinity is a number value, albeit a special one.

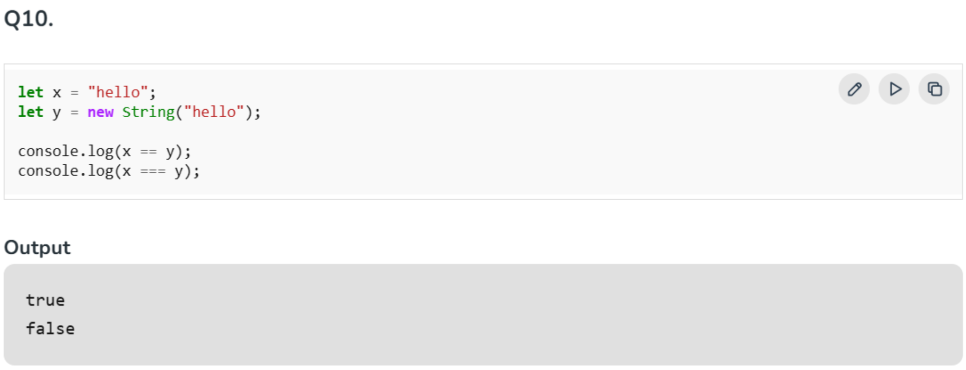
Q8.

```
let x = "5";
let y = 2;

console.log(x + y);
console.log(x - y);
```

Output

```
52
3
```



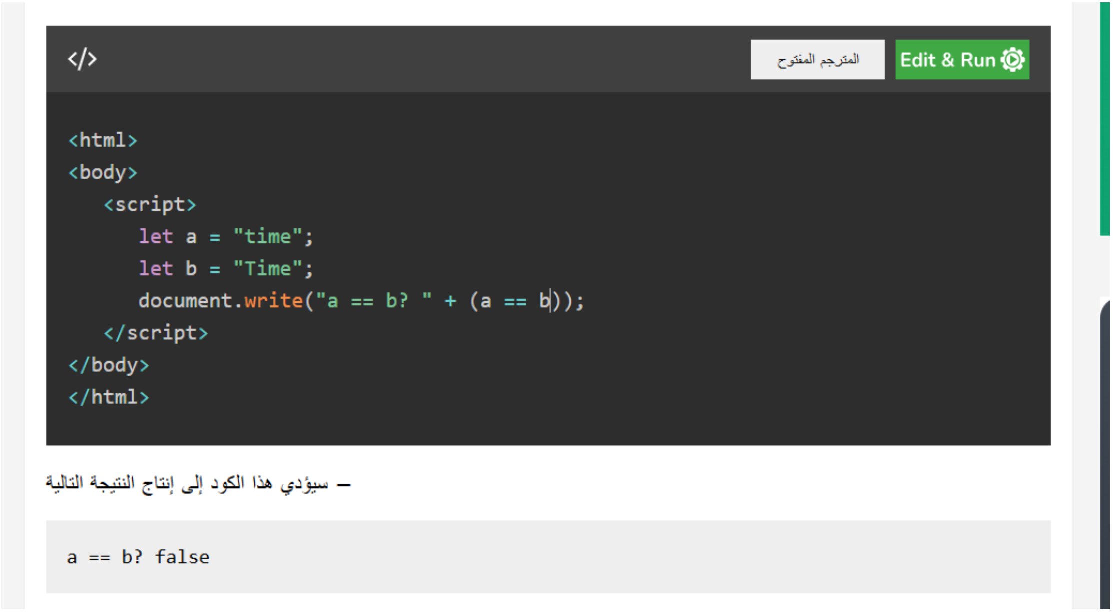
Q10.

```
let x = "hello";
let y = new String("hello");

console.log(x == y);
console.log(x === y);
```

Output

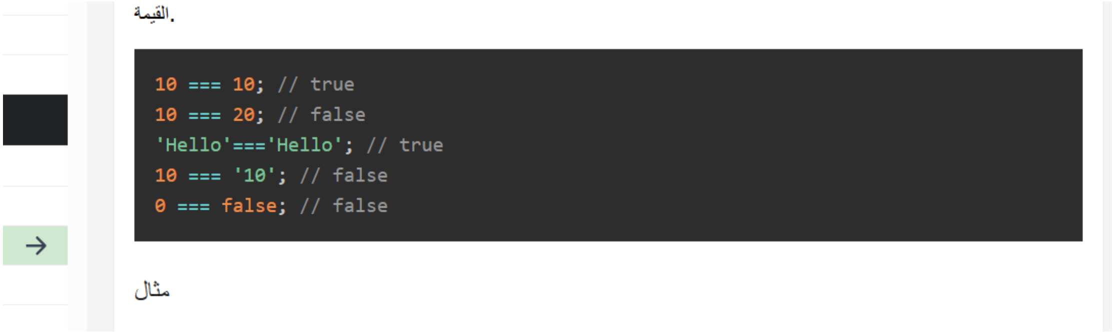
```
true
false
```



```
<html>
<body>
   <script>
      let a = "time";
      let b = "Time";
      document.write("a == b? " + (a == b));
   </script>
</body>
</html>
```

– سيؤدي هذا الكود إلى إنتاج النتيجة التالية

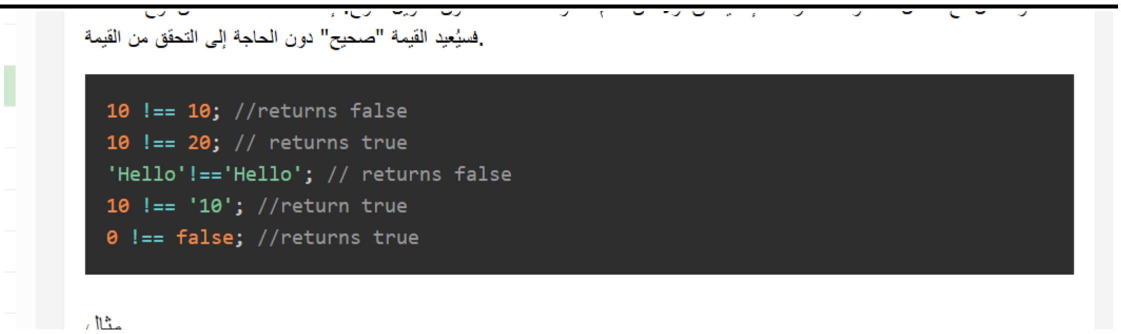
```
a == b? false
```



القيمة.

```
10 === 10; // true
10 === 20; // false
'Hello'==='Hello'; // true
10 === '10'; // false
0 === false; // false
```

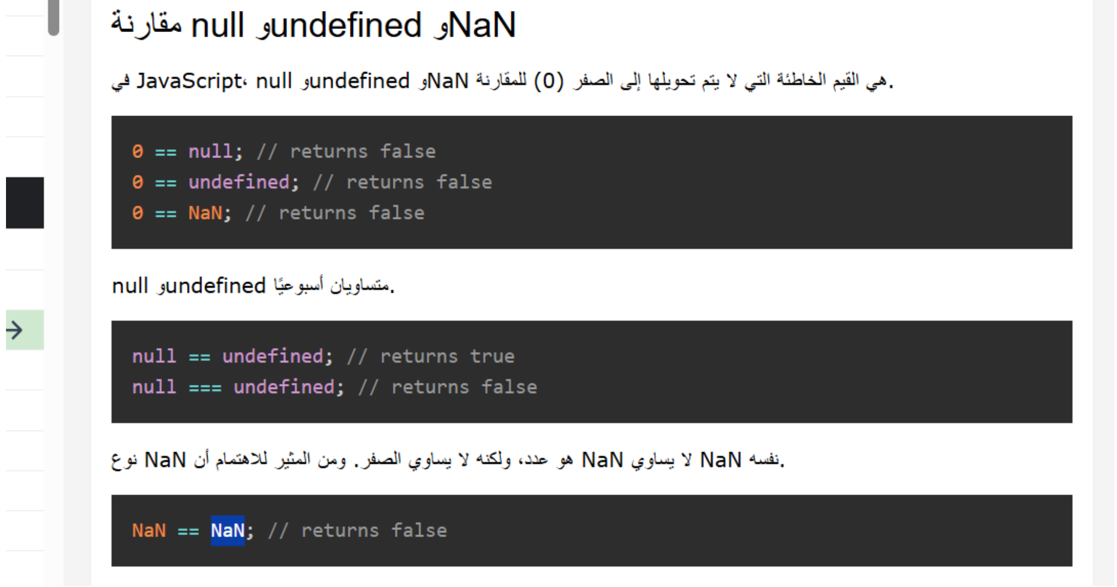
مثال



فسيُعيد القيمة "صحيح" دون الحاجة إلى التحقق من القيمة.

```
10 !== 10; //returns false
10 !== 20; // returns true
'Hello'!=='Hello'; // returns false
10 !== '10'; //return true
0 !== false; //returns true
```

مثال



مقارنة NaN وundefined وnull

في JavaScript، null وundefined وNaN هي القيم الخاطئة التي لا يتم تحويلها إلى الصفر (0) للمقارنة.

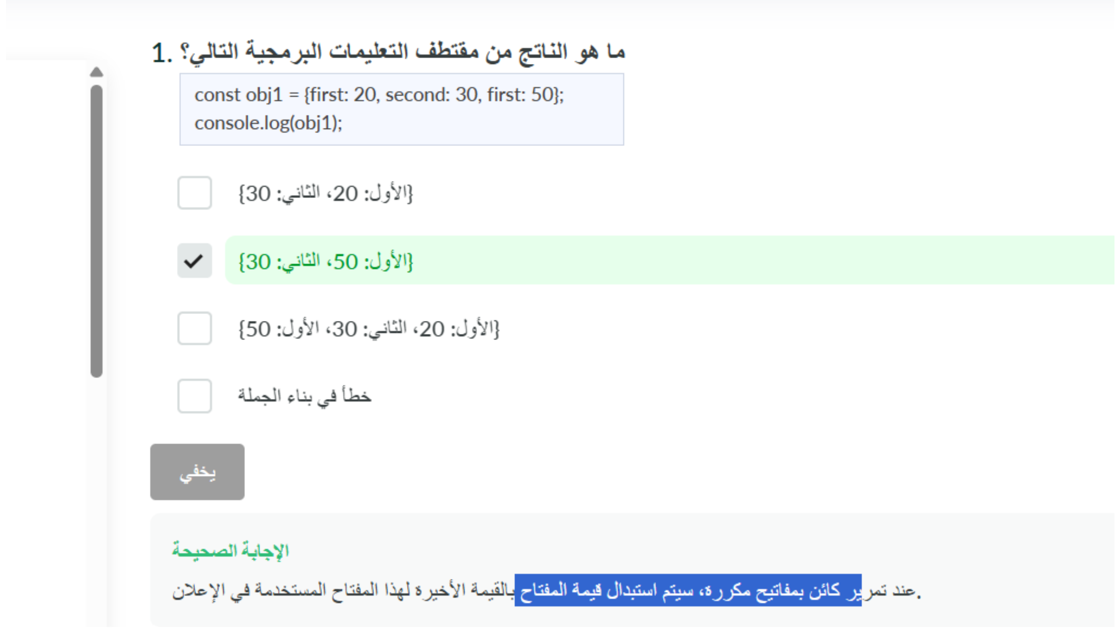
```
0 == null; // returns false
0 == undefined; // returns false
0 == NaN; // returns false
```

null وundefined متساويان أسبوعيًا.

```
null == undefined; // returns true
null === undefined; // returns false
```

NaN لا يساوي NaN نفسه. NaN هو عدد، ولكنه لا يساوي الصفر. ومن المثير للاهتمام أن نوع

```
NaN == NaN; // returns false
```



1. ما هو الناتج من مقتطف التعليمات البرمجية التالي؟

```
const obj1 = {first: 20, second: 30, first: 50};
console.log(obj1);
```

□ {الأول: 20، الثاني: 30}

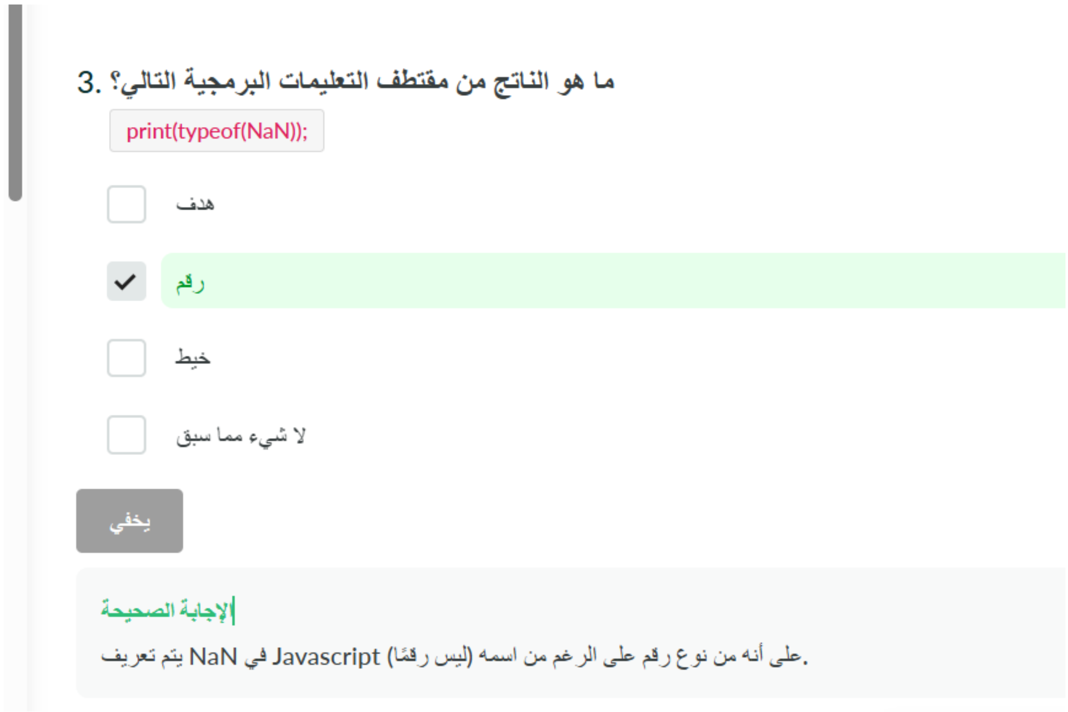
☑ {الأول: 50، الثاني: 30}

□ {الأول: 20، الثاني: 30، الأول: 50}

□ خطأ في بناء الجملة

الإجابة الصحيحة

عند تمرير كائن بمفاتيح مكررة، سيتم استبدال قيمة المفتاح بالقيمة الأخيرة لهذا المفتاح المستخدمة في الإعلان.



3. ما هو الناتج من مقتطف التعليمات البرمجية التالي؟

```
print(typeof(NaN));
```

□ هدف

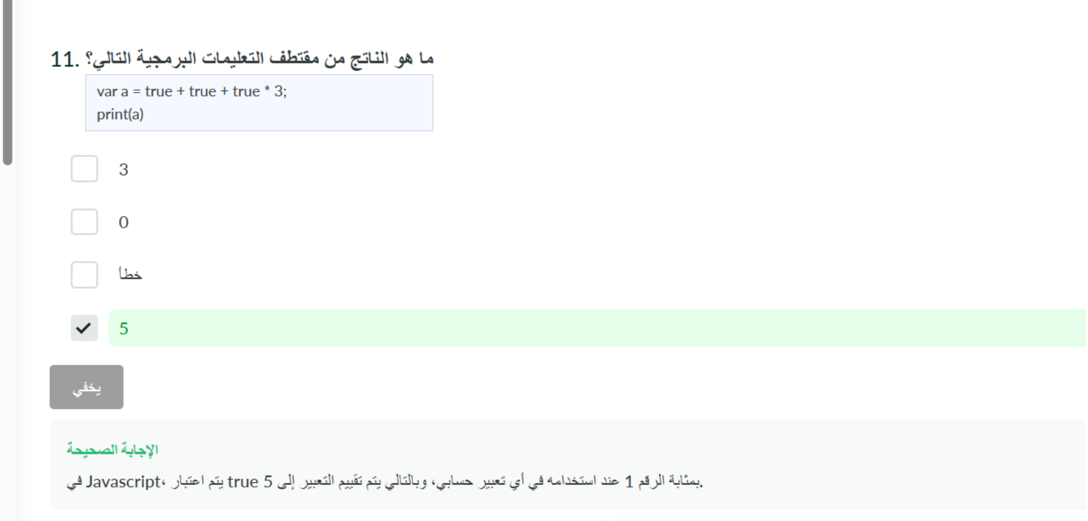
☑ رقم

□ خيط

□ لا شيء مما سبق

الإجابة الصحيحة

يتم تعريف NaN في Javascript على أنه من نوع رقم على الرغم من اسمه (ليس رقمًا).



11. ما هو الناتج من مقتطف التعليمات البرمجية التالي؟

```
var a = true + true + true * 3;
print(a)
```

□ 3

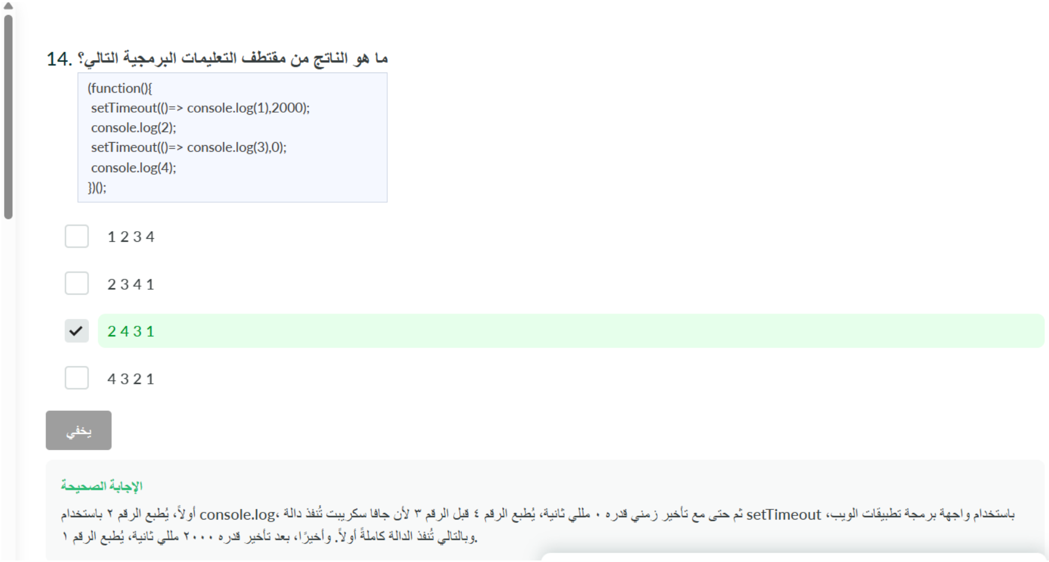
□ 0

□ خطأ

☑ 5

الإجابة الصحيحة

بمثابة الرقم 1 عند استخدامه في أي تعبير حسابي، وبالتالي يتم تقييم التعبير إلى 5 true يتم اعتبار ،Javascript في.



14. ما هو الناتج من مقتطف التعليمات البرمجية التالي؟

```
(function(){
 setTimeout(()=> console.log(1),2000);
 console.log(2);
 setTimeout(()=> console.log(3),0);
 console.log(4);
})();
```

□ 1 2 3 4

□ 2 3 4 1

☑ 2 4 3 1

□ 4 3 2 1

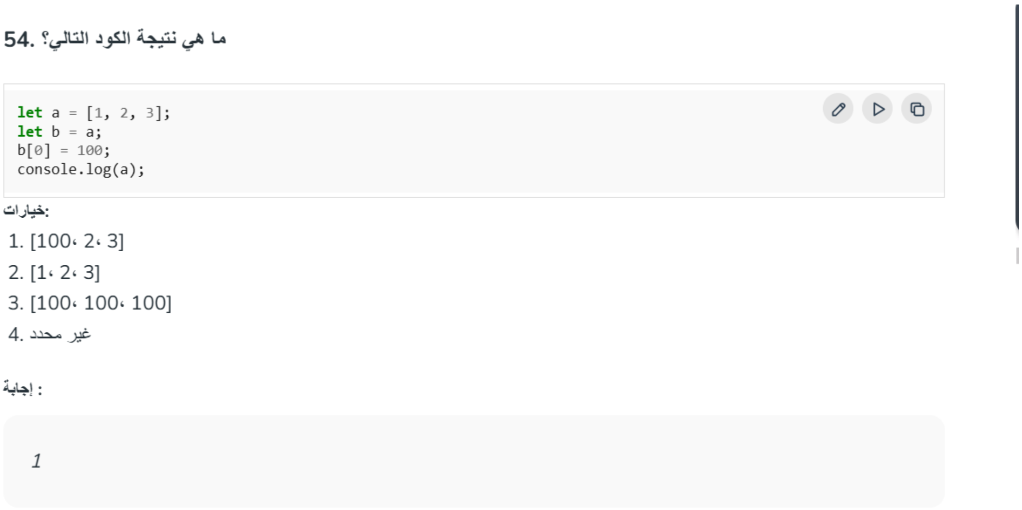
الإجابة الصحيحة

باستخدام واجهة برمجة تطبيقات الويب، setTimeout ثم حتى مع تأخير زمني قدره ٠ مللي ثانية، يُطبع الرقم ٤ قبل الرقم ٣ لأن جافا سكريبت تُنفذ دالة ،console.log أولاً، يُطبع الرقم ٢ باستخدام
وبالتالي تُنفذ الدالة كاملةً أولاً. وأخيرًا، بعد تأخير قدره ٢٠٠٠ مللي ثانية، يُطبع الرقم ١.

ماذا سيعيد مقتطف الكود هذا

console.log(typeof null);

1. Boolean
2. Object
3. Number
4. String



54. ما هي نتيجة الكود التالي؟

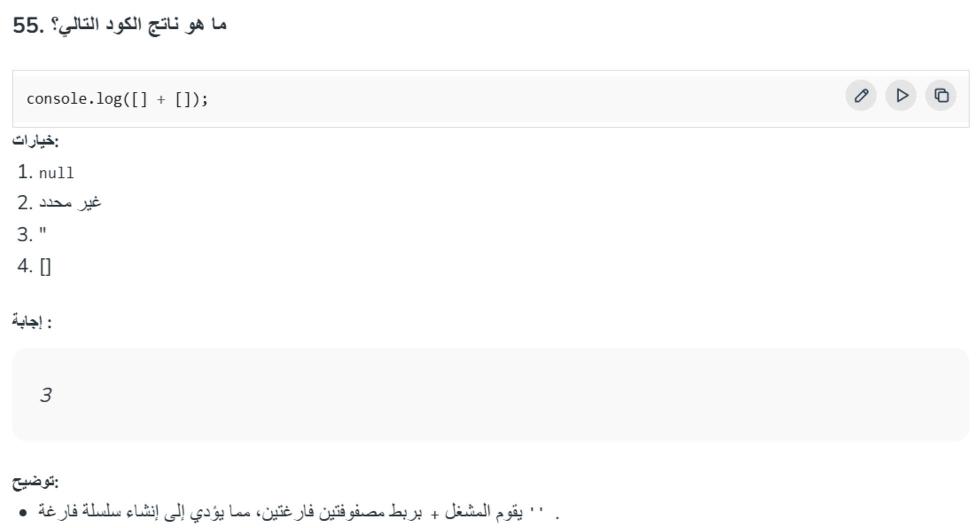
```
let a = [1, 2, 3];
let b = a;
b[0] = 100;
console.log(a);
```

خيارات:

1. [100، 2، 3]
2. [1، 2، 3]
3. [100، 100، 100]
4. غير محدد

إجابة :

1



55. ما هو ناتج الكود التالي؟

```
console.log([] + []);
```

خيارات:

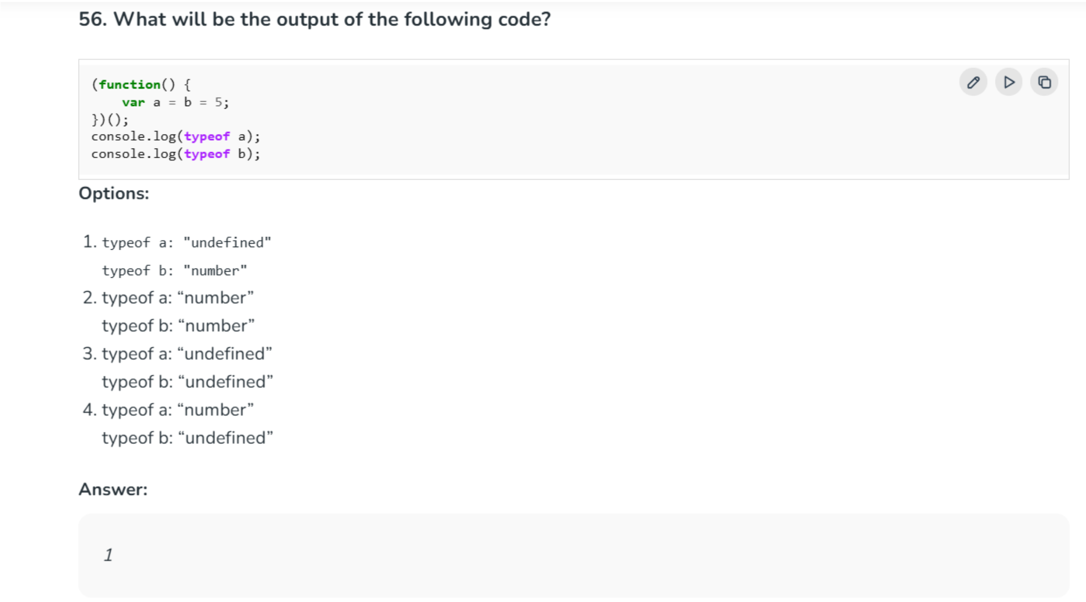
1. `null`
2. غير محدد
3. ''
4. []

إجابة :

3

توضيح:

- . '' يقوم المشغل + بربط مصفوفتين فارغتين، مما يؤدي إلى إنشاء سلسلة فارغة



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

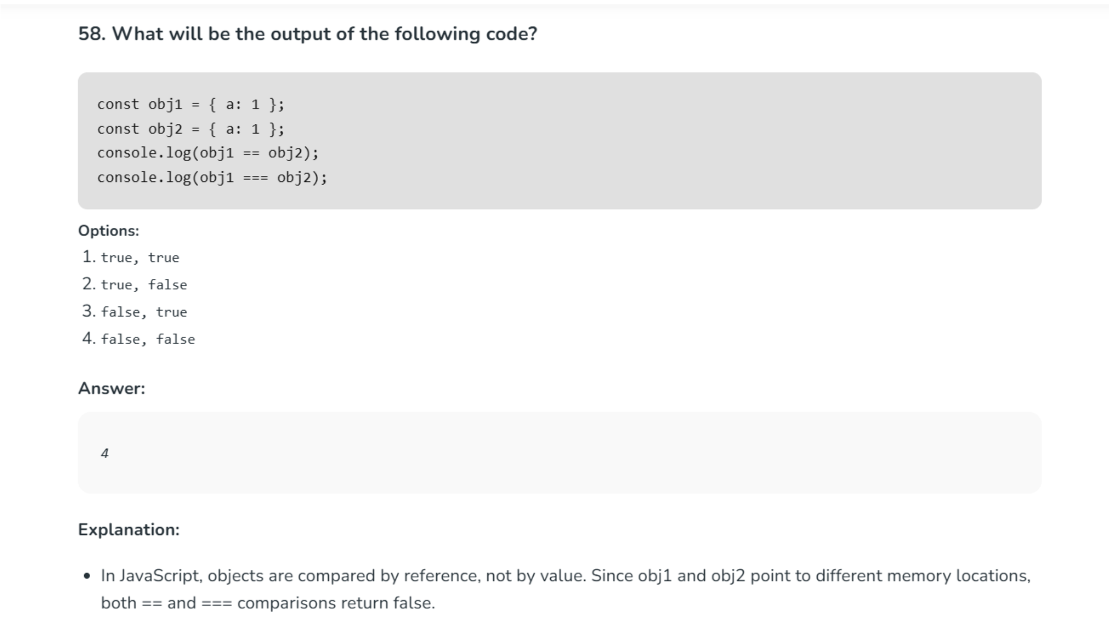
```
(function() {
    var a = b = 5;
})();
console.log(typeof a);
console.log(typeof b);
```

Options:

1. `typeof a: "undefined"`
   `typeof b: "number"`
2. typeof a: “number”
   typeof b: “number”
3. typeof a: “undefined”
   typeof b: “undefined”
4. typeof a: “number”
   typeof b: “undefined”

Answer:

1



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

```
const obj1 = { a: 1 };
const obj2 = { a: 1 };
console.log(obj1 == obj2);
console.log(obj1 === obj2);
```

Options:

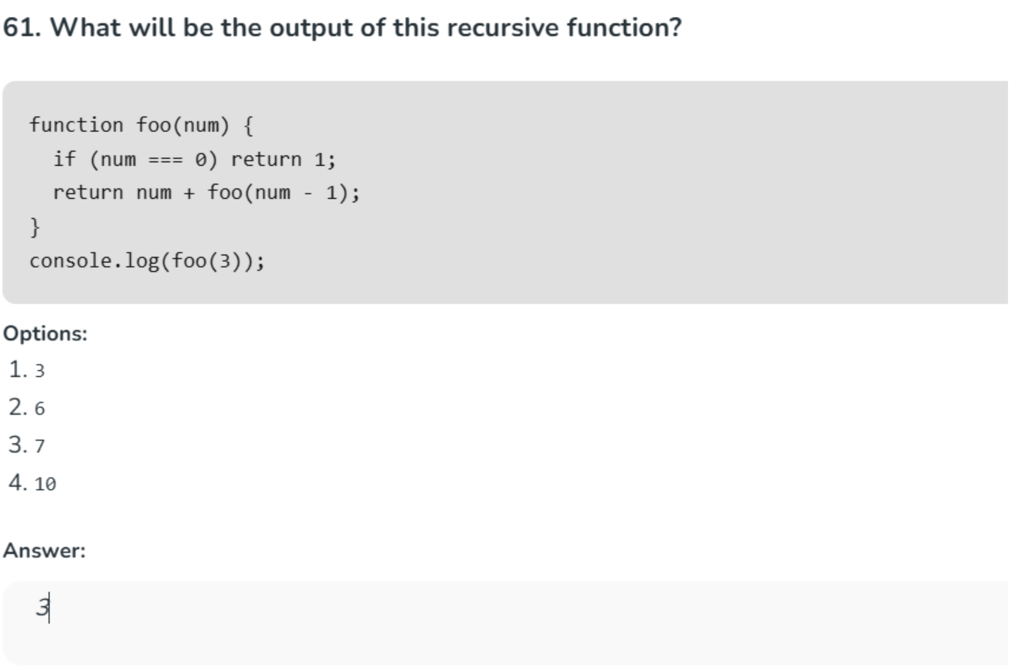
1. `true, true`
2. `true, false`
3. `false, true`
4. `false, false`

Answer:

4

Explanation:

- In JavaScript, objects are compared by reference, not by value. Since obj1 and obj2 point to different memory locations, both == and === comparisons return false.



61. What will be the output of this recursive function?

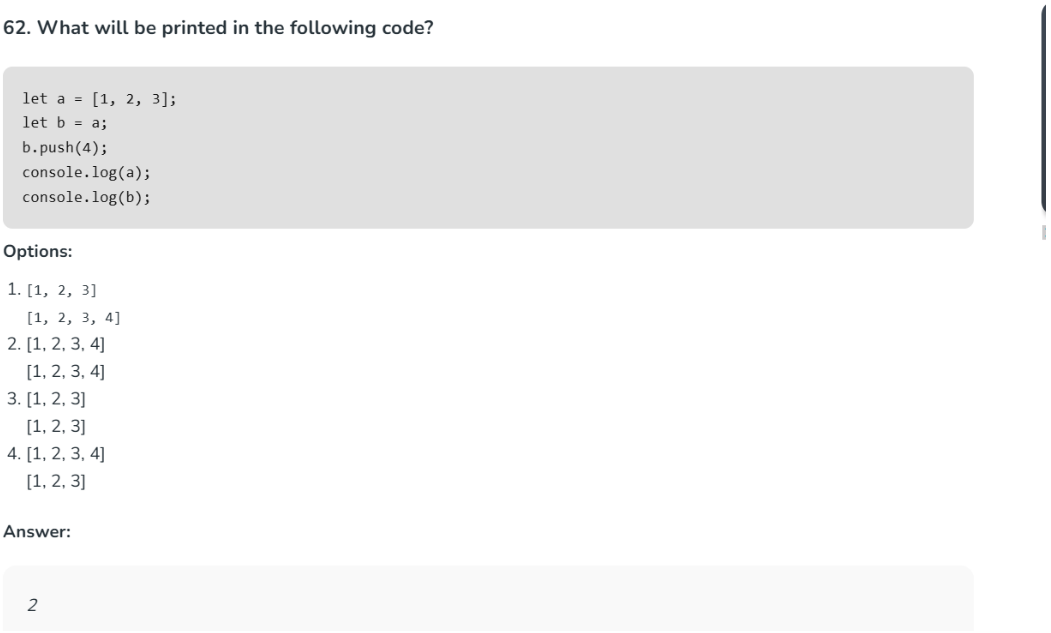
```
function foo(num) {
  if (num === 0) return 1;
  return num + foo(num - 1);
}
console.log(foo(3));
```

Options:

1. 3
2. 6
3. 7
4. 10

Answer:

3



62. What will be printed in the following code?

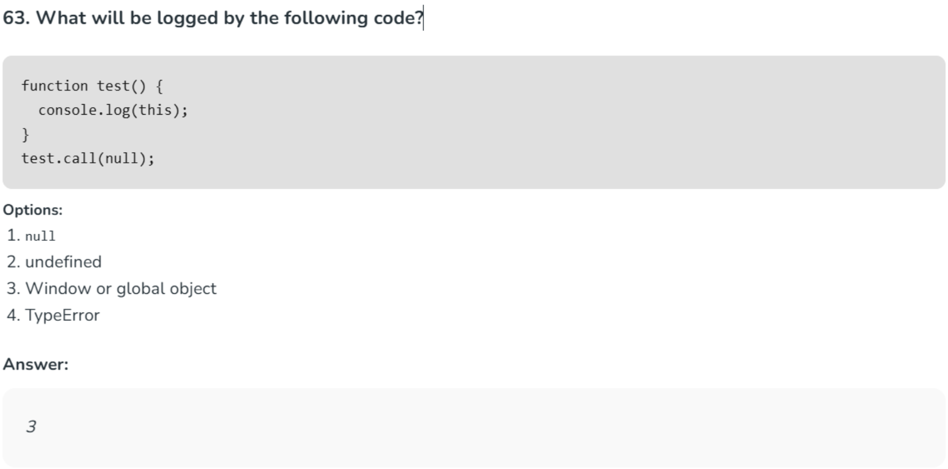
```
let a = [1, 2, 3];
let b = a;
b.push(4);
console.log(a);
console.log(b);
```

Options:

1. [1, 2, 3]
   [1, 2, 3, 4]
2. [1, 2, 3, 4]
   [1, 2, 3, 4]
3. [1, 2, 3]
   [1, 2, 3]
4. [1, 2, 3, 4]
   [1, 2, 3]

Answer:

2



63. What will be logged by the following code?

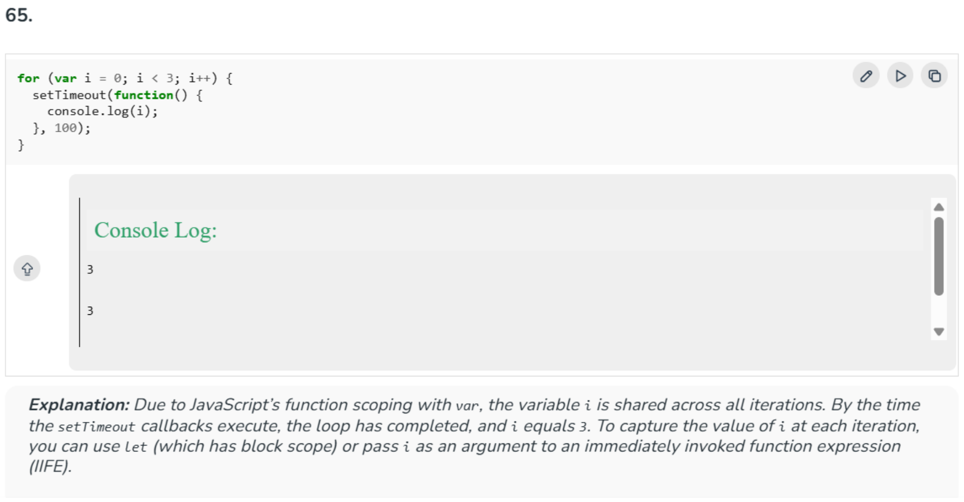
```
function test() {
  console.log(this);
}
test.call(null);
```

Options:

1. null
2. undefined
3. Window or global object
4. TypeError

Answer:

3



65.

```
for (var i = 0; i < 3; i++) {
  setTimeout(function() {
    console.log(i);
  }, 100);
}
```

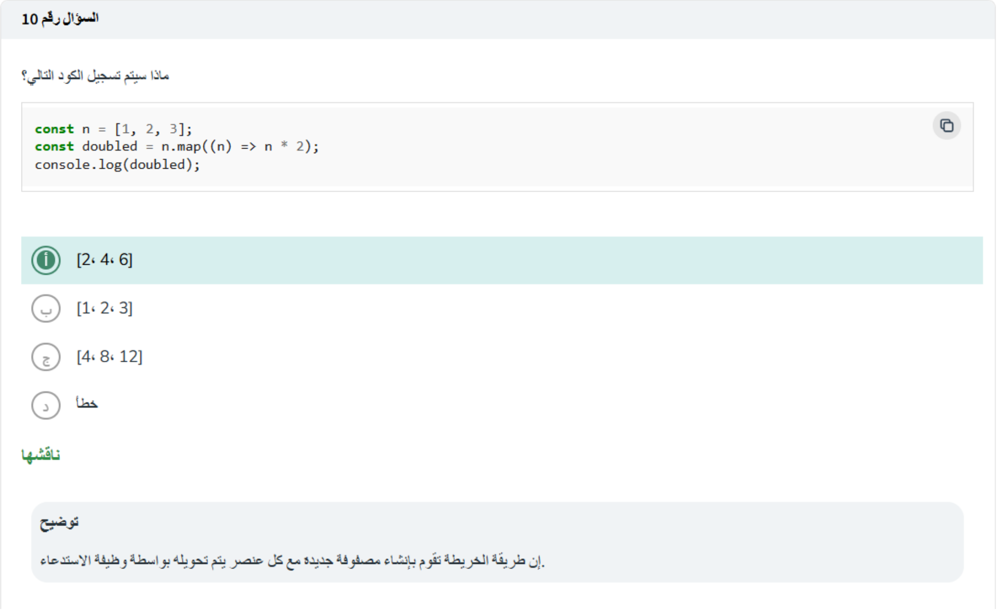
Console Log:

```
3

3
```

Explanation: Due to JavaScript's function scoping with `var`, the variable `i` is shared across all iterations. By the time the `setTimeout` callbacks execute, the loop has completed, and `i` equals `3`. To capture the value of `i` at each iteration, you can use `let` (which has block scope) or pass `i` as an argument to an immediately invoked function expression (IIFE).

لو صار التعريف let ماذا سيحدث هل نفس المخرجات



السؤال رقم 10

ماذا سيتم تسجيل الكود التالي؟

```
const n = [1, 2, 3];
const doubled = n.map((n) => n * 2);
console.log(doubled);
```

أ [2، 4، 6]

ب [1، 2، 3]

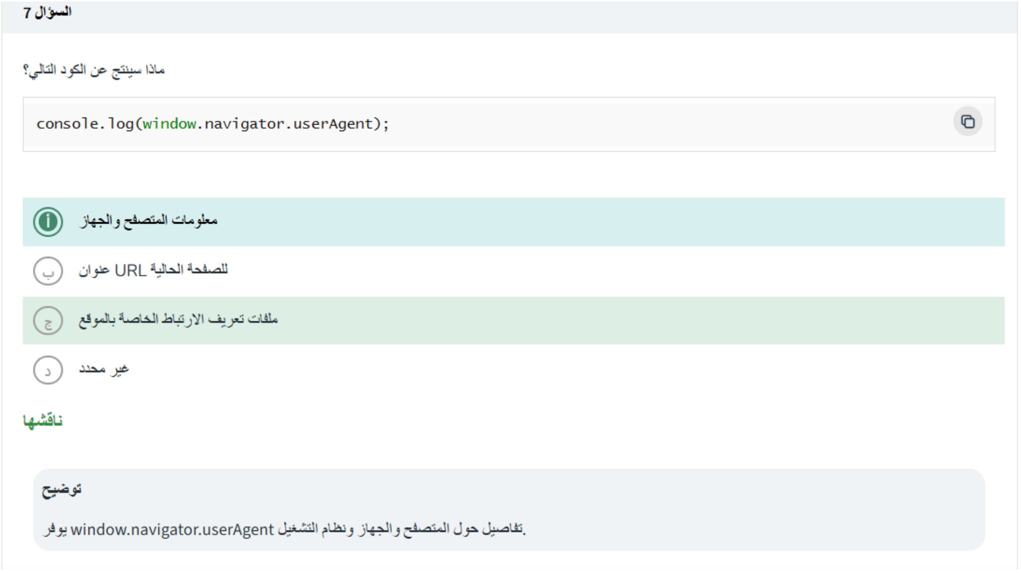
ج [4، 8، 12]

د خطأ

ناقشها

توضيح

إن طريقة الخريطة تقوم بإنشاء مصفوفة جديدة مع كل عنصر يتم تحويله بواسطة وظيفة الاستدعاء.



السؤال 7

ماذا سينتج عن الكود التالي؟

```
console.log(window.navigator.userAgent);
```

أ معلومات المتصفح والجهاز

ب عنوان URL للصفحة الحالية

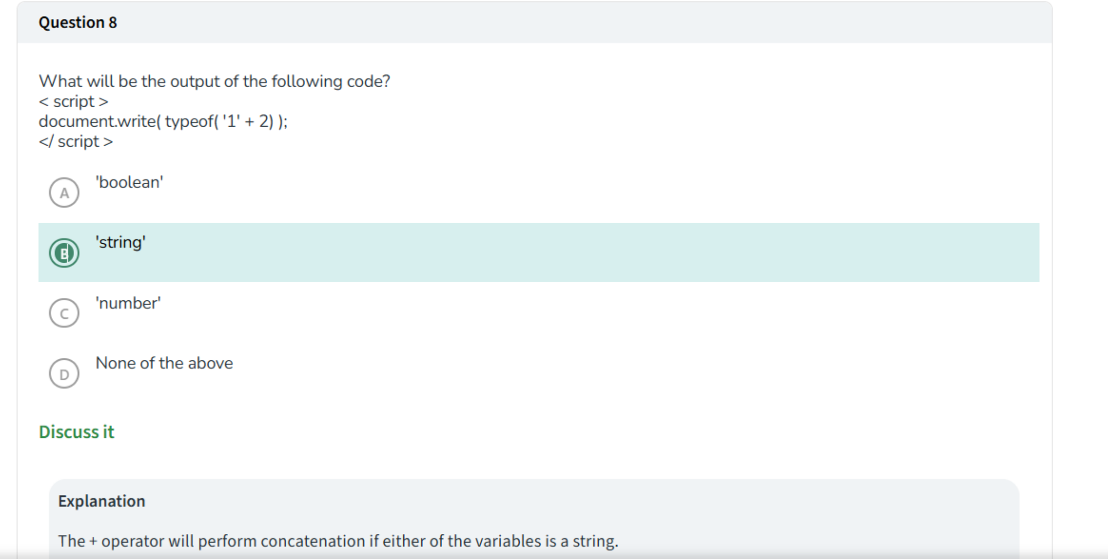
ج ملفات تعريف الارتباط الخاصة بالموقع

د غير محدد

ناقشها

توضيح

يوفر window.navigator.userAgent تفاصيل حول المتصفح والجهاز ونظام التشغيل.



Question 8

What will be the output of the following code?
< script >
document.write(typeof('1' + 2));
</ script >

A 'boolean'

[✓ B] 'string'

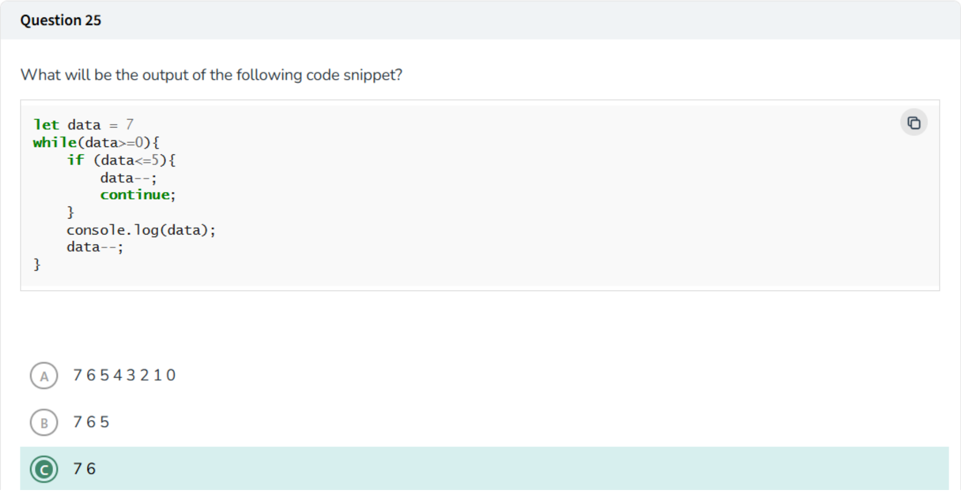
C 'number'

D None of the above

Discuss it

Explanation

The + operator will perform concatenation if either of the variables is a string.



Question 25

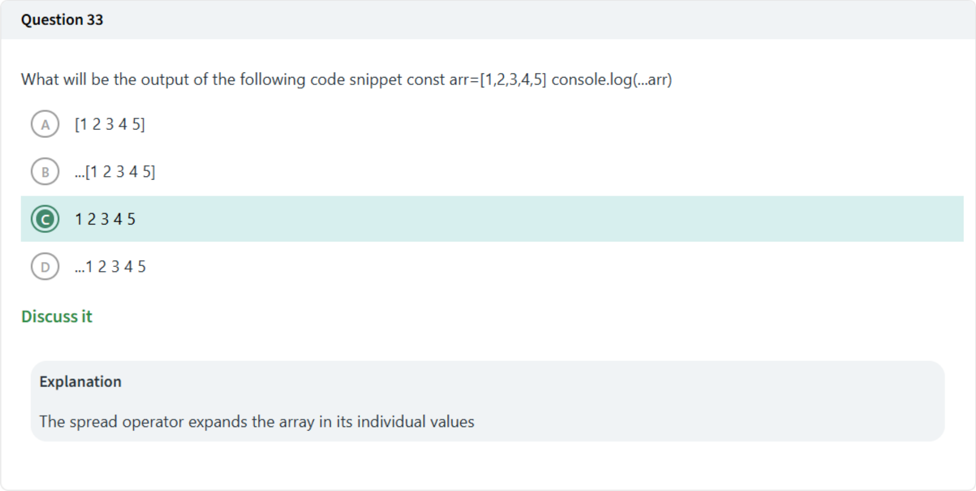
What will be the output of the following code snippet?

```
let data = 7
while(data>=0){
    if (data<=5){
        data--;
        continue;
    }
    console.log(data);
    data--;
}
```

A 7 6 5 4 3 2 1 0

B 7 6 5

[✓ C] 7 6



Question 33

What will be the output of the following code snippet const arr=[1,2,3,4,5] console.log(...arr)

A [1 2 3 4 5]

B ...[1 2 3 4 5]

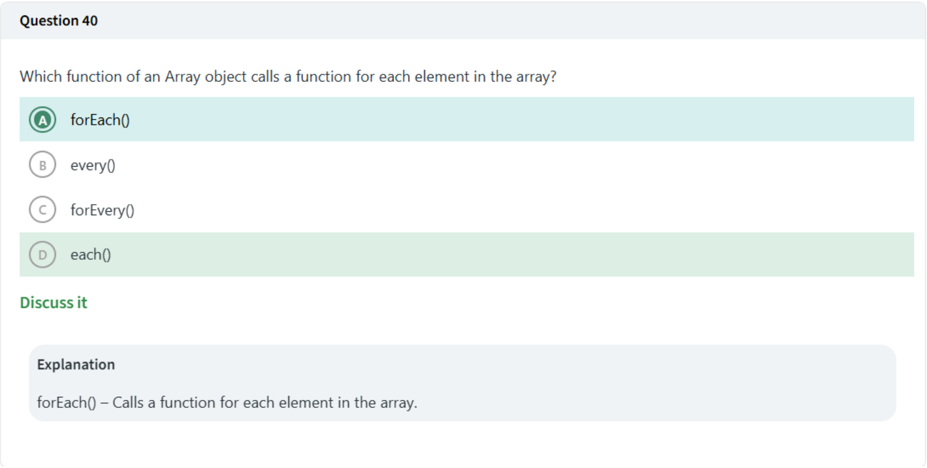
[✓ C] 1 2 3 4 5

D ...1 2 3 4 5

Discuss it

Explanation

The spread operator expands the array in its individual values



Question 40

Which function of an Array object calls a function for each element in the array?

A forEach()

B every()

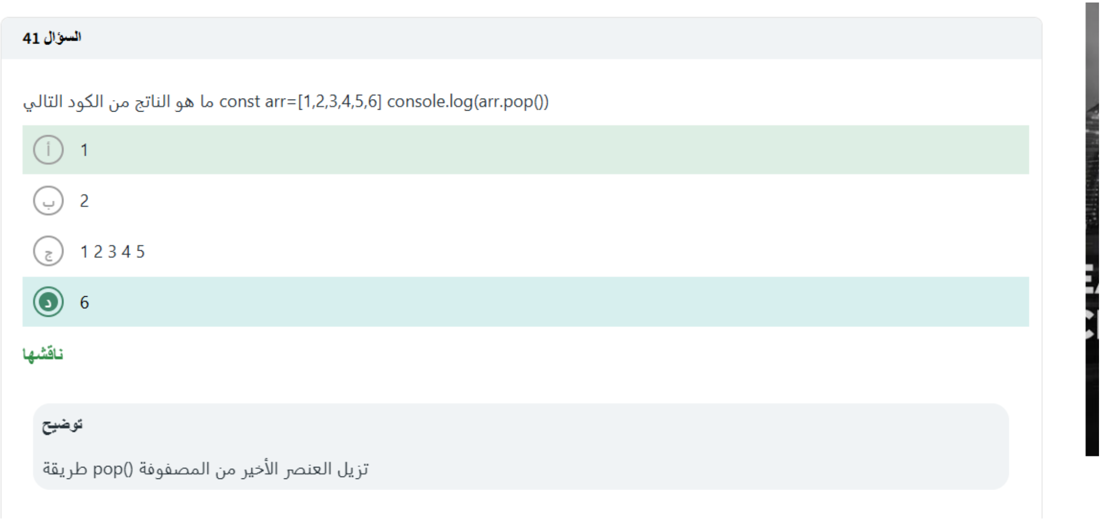
C forEvery()

D each()

Discuss it

Explanation

forEach() – Calls a function for each element in the array.



السؤال 41

ما هو الناتج من الكود التالي const arr=[1,2,3,4,5,6] console.log(arr.pop())

أ 1

ب 2

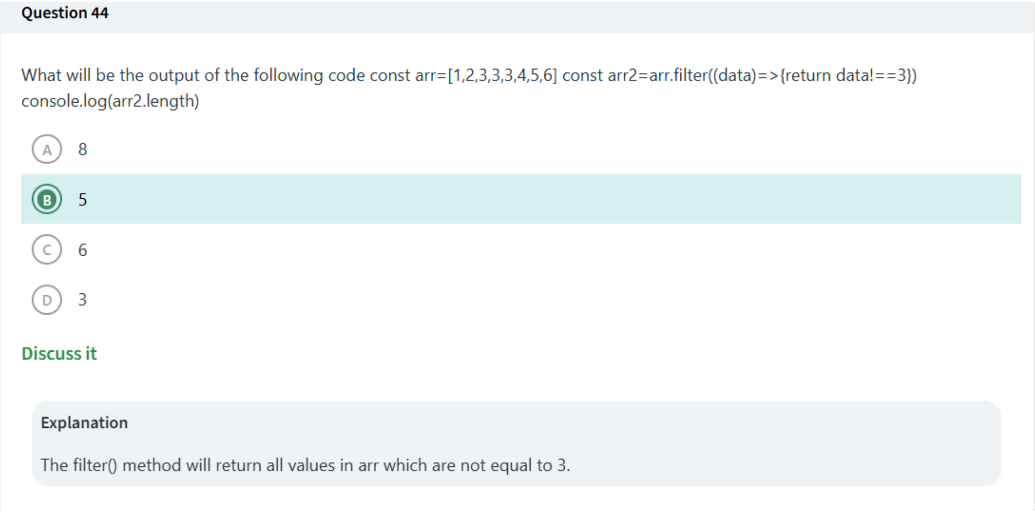
ج 1 2 3 4 5

د 6

ناقشها

توضيح

طريقة pop() تزيل العنصر الأخير من المصفوفة



Question 44

What will be the output of the following code const arr=[1,2,3,3,3,4,5,6] const arr2=arr.filter((data)=>{return data!==3}) console.log(arr2.length)

A 8

B 5

C 6

D 3

Discuss it

Explanation

The filter() method will return all values in arr which are not equal to 3.