10X - Midterm

Total points 49/52





The respondent's email (nkyx1987@gmail.com) was recorded on submission of this form.

| პირადი ინფორმაცია | 0 of 0 points |
|-------------------|---------------|
| პირადი ნომერი * | |
| 61001071949 | |
| | |

ტელეფონის ნომერი *

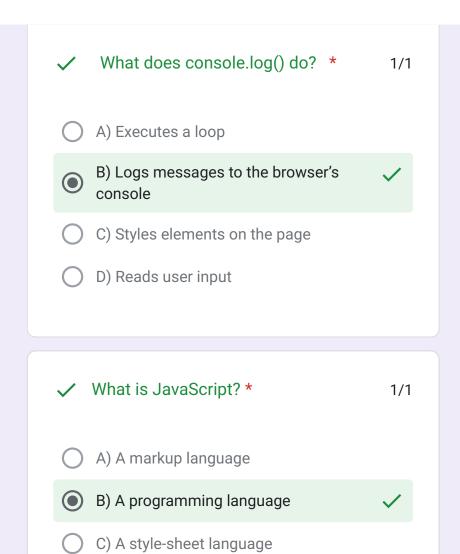
593212279

სახელი, გვარი *

ნიკა ბერიძე

Core JS foundations

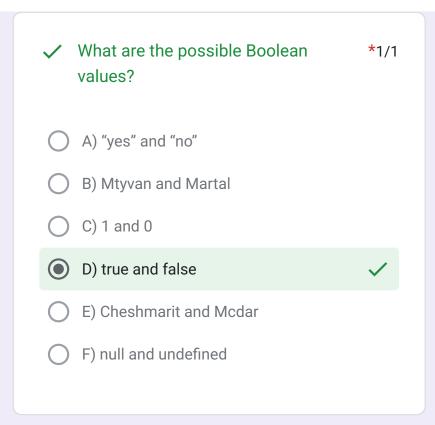
3 of 4 points

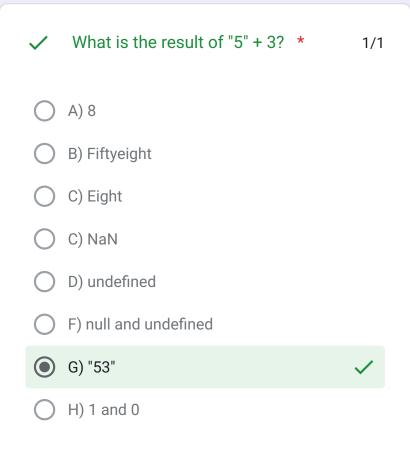


O D) A database query language

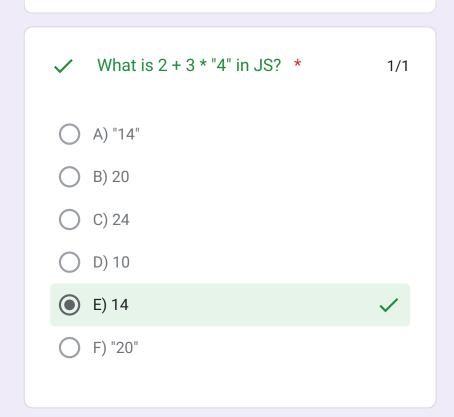
| Which description best fits *0/1 JavaScript's typing? |
|---|
| A) Statically typed |
| B) Dynamically and weakly typed |
| C) Strongly typed |
| D) Compiled language |
| Correct answer |
| B) Dynamically and weakly typed |
| |
| ✓ Where can JavaScript run? * 1/1 |
| A) Only in the browser |
| B) Only on the server |
| C) Both in the browser and on the server |
| O) Only in text editors |
| |
| Variables & Data Types 5 of 5 points |

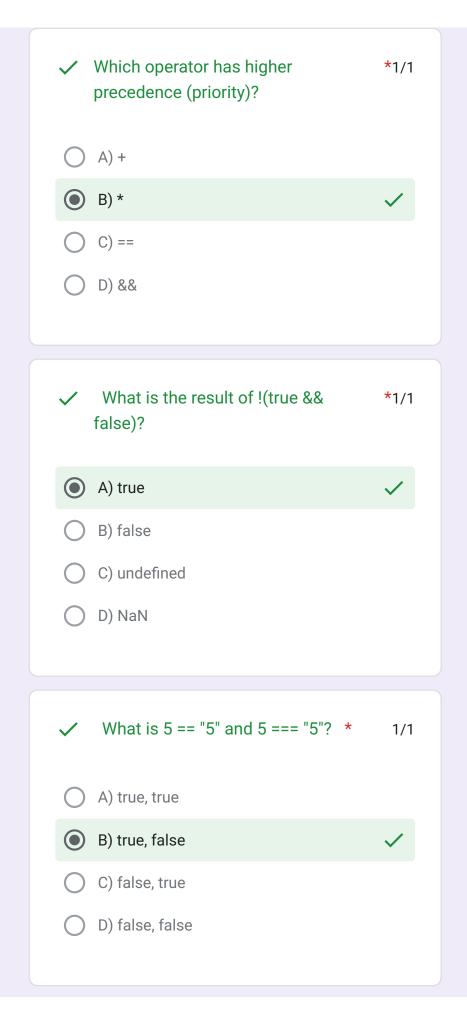
| A) "null" B) "undefined" C) "object" D) "string" ✓ What's the difference between let and const? A) let is hoisted; const is not B) let can be reassigned; const cannot C) const is block-scoped; let is function-scoped D) They're identical ✓ Which data type is used for text? * 1/1 A) Number B) Boolean C) String ✓ D) Object | ✓ | What is the output of typeof null? * | 1/1 |
|--|----------|--|----------|
| C) "object" D) "string" What's the difference between let and const? A) let is hoisted; const is not B) let can be reassigned; const cannot C) const is block-scoped; let is function-scoped D) They're identical Which data type is used for text? * 1/1 A) Number B) Boolean C) String | 0 | A) "null" | |
| ✓ What's the difference between let and const? ✓ A) let is hoisted; const is not ⑤ B) let can be reassigned; const cannot ✓ C) const is block-scoped; let is function-scoped ⑥ D) They're identical ✓ Which data type is used for text? * 1/1 ⑥ A) Number ⑥ B) Boolean ⑥ C) String | 0 | B) "undefined" | |
| ✓ What's the difference between let and const? ✓ A) let is hoisted; const is not ⑥ B) let can be reassigned; const cannot ✓ C) const is block-scoped; let is function-scoped ⑥ D) They're identical ✓ Which data type is used for text? * 1/1 ⑥ A) Number ⑥ B) Boolean ⑥ C) String | | C) "object" | ✓ |
| and const? A) let is hoisted; const is not B) let can be reassigned; const cannot ✓ C) const is block-scoped; let is function-scoped D) They're identical Which data type is used for text? ★ 1/1 A) Number B) Boolean C) String ✓ | 0 | D) "string" | |
| B) let can be reassigned; const cannot C) const is block-scoped; let is function-scoped D) They're identical Which data type is used for text? * 1/1 A) Number B) Boolean C) String | ✓ | | *1/1 |
| C) const is block-scoped; let is function-scoped D) They're identical Which data type is used for text? * 1/1 A) Number B) Boolean C) String | 0 | A) let is hoisted; const is not | |
| Scoped D) They're identical Which data type is used for text? * 1/1 A) Number B) Boolean C) String | | B) let can be reassigned; const cannot | ✓ |
| ✓ Which data type is used for text? * 1/1 A) Number B) Boolean C) String | 0 | | |
| A) NumberB) BooleanC) String | 0 | D) They're identical | |
| ○ B) Boolean○ C) String | ~ | Which data type is used for text? * | 1/1 |
| C) String | 0 | A) Number | |
| | 0 | B) Boolean | |
| O D) Object | | C) String | ✓ |
| | 0 | D) Object | |





| ✓ | What does x += 5 do? * | 1/1 |
|----------|--|----------|
| 0 | A) Sets x to 5 | |
| • | B) Adds 5 to x and assigns the result to x | ✓ |
| 0 | C) Increments x by 1 five times | |
| 0 | D) Throws an error | |





✓ Which of these correctly checks if *1/1 age is between 18 and 30 inclusive?

- A) if (age >= 18 && age <= 30) {}</p>
- B) if (18 <= age <= 30) {}
- O if (18 < age < 30) {}
- D) if (age > 18 || age < 30) {}</p>

✓ Evaluate this condition: * 1/1

- A) X
- B) Y
- C) No output
- D) Error

```
    ✓ Which is correct if syntax? * 1/1
    ○ A) if x > 5 {}
    ○ B) if (x > 5) {}
    ○ C) if x > 5 then {}
    ○ D) if [x > 5] {}
```

```
What will this output? * 1/1

let x = 2;
switch(x) {
   case 1: console.log("One"); break;
   case 2: console.log("Two"); break;
   default: console.log("Other");
  }

A) One

B) Two

C) Other

D) Error
```

```
    ✓ What does let a = (true) ? 1 : 2; *1/1 assign to a?
    △ A) 0
    ♠ B) 1
    ✓ C) 2
    ♠ D) undefined
```

Functions & Scope 8 of 10 points ✓ Which is an example of a callback? * 1/1 A) setTimeout(() => {}, 1000) B) console.log("Hi") C) let a = 5; D) function x() {}

| ★ Which are hoisted? * | 0/1 |
|-------------------------------|----------|
| A) Function declarations | ✓ |
| B) Function expressions | |
| C) let and const declarations | |
| D) Arrow functions | |
| E) All | |
| Correct answer | |
| A) Function declarations | |
| C) let and const declarations | |
| | |

How do you declare a named function?
A) let foo = => {}
B) function foo() {}
C) foo => {}
D) def foo() {}

```
✓ What does this code output?*

37     function outer() {
38         let count = 0;
39         return function() {
40              count++;
41               return count;
42         };
43         }
44         const c = outer();
45         console.log(c(), c());
46

○ A)11

○ B)01

○ C)12

✓ D)22
```

| ✓ | Which is a higher-order function? * | 1/1 |
|----------|--|----------|
| 0 | A) Math.max() | |
| | B) arr.map(fn) | ✓ |
| 0 | C) 5 + 2 | |
| 0 | D) const a = 1; | |
| | | |
| ~ | What's the difference between var and let? | *1/1 |

- A) var is block-scoped; let is functionscoped
- B) var is scoped B) var is function-scoped; let is block-
- O They're identical
- O) let is hoisted; var is not

| ✓ What is returned if a function has return? | s no *1/1 |
|--|-----------|
| A) null | |
| (B) 0 | |
| © C) undefined | ✓ |
| O D) false | |
| O E) ააფერი | |
| | |

- ✓ Which is valid arrow function syntax? * 1/1
- A) const f = function=>{}
- B) const f = () => {}
- C) function => {}
- D) () => function {}

```
X What does this log? *

31
32    function greet(name) {
33        return "Hi " + name;
34    }
35        console.log(greet("Ana"));
36

A) undefined

B) Hi Ana

C) greet Ana

D) Error

Correct answer

B) Hi Ana

B) Hi Ana

O B) Hi Ana

O D) Error

Correct answer

B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

Correct answer

O B) Hi Ana

O D) Error

O D) Error
```

Loops

7 of 7 points

```
✓ What does this log? *

49
50     for (let i = 0; i < 3; i++) {
51          console.log(i);
52     }
53

○ A) 1 2 3

○ B) 0 1 2

○ C) 0 1 2 3

○ D) Error
</pre>
```

```
✓ What does this log? *

49
50     for (let i = 0; i < 3; i++) {
        console.log(i);
52     }
53

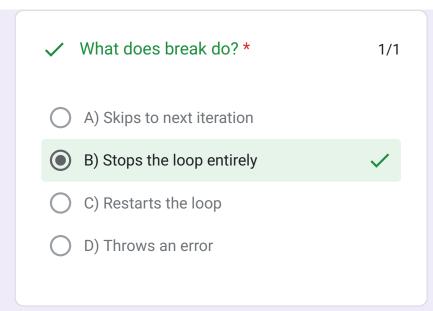
○ A) 1 2 3

○ B) 0 1 2

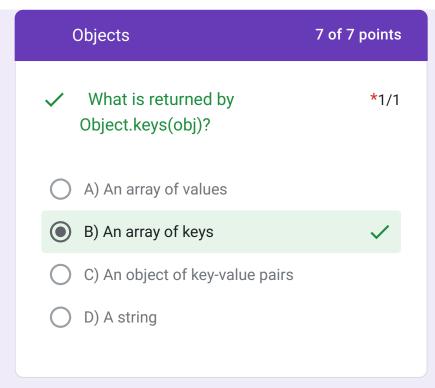
○ C) 0 1 2 3

○ D) Error

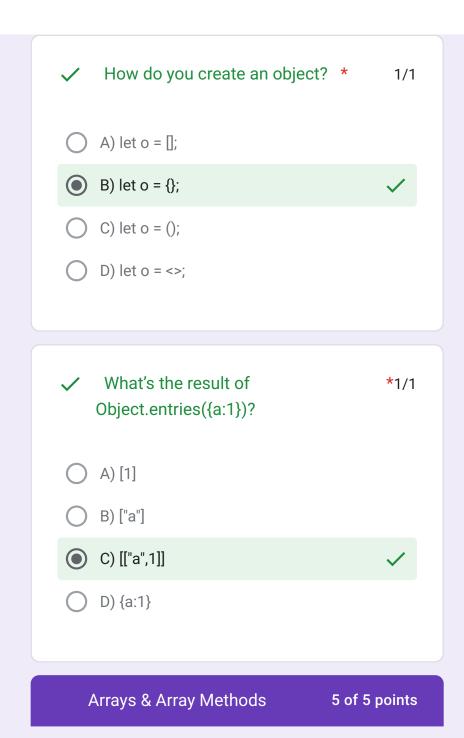
✓</pre>
```



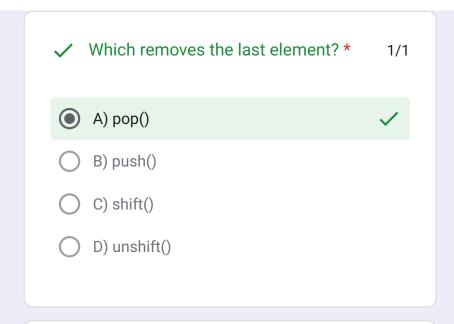
| ✓ | What does continue do? * | 1/1 |
|------------|------------------------------------|----------|
| \bigcirc | A) Stops loop | |
| | B) Skips to the next iteration | ✓ |
| 0 | C) Exits program | |
| 0 | D) Resets counter | |
| | | |
| ✓ | Which loop iterates object keys? * | 1/1 |
| \bigcirc | A) for | |
| | B) forin | ✓ |
| 0 | C) forof | |
| 0 | D) while | |
| | | |
| ✓ | Which always runs at least once? * | 1/1 |
| \bigcirc | A) for | |
| \circ | B) while | |
| | C) dowhile | ✓ |
| 0 | D) forof | |
| | | |



| A) obj.name |
|---|
| |
| B) obj->name |
| C) obj[name] |
| D) obj{"name"} |
| |
| ✓ What does delete obj.key do? * 1/1 |
| A) Sets property to null |
| B) Removes the property |
| C) Throws an error |
| D) Renames the key |
| |
| ✓ What does <u>this</u> refer to in a method? * 1/1 |
| A) Global object |
| B) The object calling the method |
| C) Function definition |
| D) Window only |
| |



| / | How do you create an empty array? | * 1/1 |
|----------|-----------------------------------|--------------|
| • | A) [] | ✓ |
| 0 | B) {} | |
| 0 | C) () | |
| 0 | D) <> | |
| ✓ | What does arr.map(fn) return? * | 1/1 |
| 0 | A) Single value | |
| • | B) New array of results | ✓ |
| 0 | C) Modifies original array | |
| 0 | D) Undefined | |
| ✓ | Which extracts a subarray? * | 1/1 |
| 0 | A) splice() | |
| • | B) slice() | ✓ |
| 0 | C) merge() | |
| 0 | D) split() | |



- ✓ Which filters even numbers from *1/1 [1,2,3,4]?
- A) [1,2,3,4].filter(n => n % 2 === 0)
- B) [1,2,3,4].map(n => n % 2 === 0)
- C) [1,2,3,4].reduce(...)
- D) [1,2,3,4].forEach(...)

DOM manipulation

2 of 2 points

| ~ | / | Which creates a new <div>? *</div> | 1/1 |
|---|------------|------------------------------------|----------|
| (| \subset | A) document.create("div") | |
| (| \bigcirc | B) document.makeElement("div") | |
| | • | C) document.createElement("div") | ✓ |
| (| \bigcirc | D) document.new("div") | |
| | | | |

✓ How do you select an element by *1/1 ID?

- A) document.query("#id")
- B) document.getElementById("id")
- C) document.find("id")
- D) document.id("id")

This content is neither created nor endorsed by Google. - <u>Terms of Service</u> - <u>Privacy Policy</u>

Does this form look suspicious? Report

Google Forms