Dashboard / My courses / Year 2022 / Atidim / Atidim 2023 / Atidim 2024 / Full Stack Developer & Specializations Nov24(Zim) / Jav
/ javaScript Exam

Started on	Monday, 2 December 2024, 9:25 AM
State	Finished
Completed on	Monday, 2 December 2024, 11:50 AM
Time taken	2 hours 24 mins
Grade	Not yet graded

Question **1**
Correct
Mark 6.00 out of 6.00

Which of the following will result in 'true'?

○ 1. '0' === false

● 2. '1' == 1 ✓

○ 3. '1' === 1

○ 4. null === undefined

Your answer is correct.

The correct answer is:
'1' == 1

Question **2**
Correct
Mark 6.00 out of 6.00

What will be the output of the following code?

`console.log(true ? 'Yes' : 'No');`

○ 1. 'Undefined'

● 2. 'Yes' ✓

○ 3. 'No'

○ 4. 'Error'

Your answer is correct.

The correct answer is:
'Yes'

Question **3**

Correct

Mark 6.00 out of 6.00

Which of the following loops executes at least once?

○ 1. foreach

○ 2. for loop

○ 3. while

● 4. do-while ✓

Your answer is correct.

The correct answer is:
do-while

Question **4**

Incorrect

Mark 0.00 out of 6.00

Which of the following statements is true about the 'for...in' loop?

○ 1. It can iterate over properties of an object

● 2. It iterates over numerical indices of an array ✗

○ 3. It creates a new scope

○ 4. It should be used for arrays instead of objects

Your answer is incorrect.

The correct answer is:
It can iterate over properties of an object

Question **5**

Correct

Mark 6.00 out of 6.00

What does the following code log to the console?

```
console.log(1)
console.log(2)
setTimeout(() => console.log(3), 0)

console.log(4)
console.log(5)
```

○ 1. 1
2
3
5
4

○ 2. 3
5
4
2
1

● 3. 1 ✓
2
4
5
3

○ 4. 1
2
3
4
5

Your answer is correct.

The correct answer is:
1
2
4
5
3

Question **6**

Correct

Mark 6.00 out of 6.00

What is the correct syntax for declaring an async arrow function in JavaScript?

● 1. const func = async () => {} ✓

○ 2. async function: myFunction() => {}

○ 3. function = () => async {}

○ 4. async function myFunction(){}

Your answer is correct.

The correct answer is:
const func = async () => {}

Question **7**

Complete

Marked out of 10.00

Create a function that receives an array of numbers as a parameter and returns the number of even numbers in the array.

```
function countEven(arr) {
if (arr === null) return -1;
let c = 0;
arr.forEach(item => {
if (item % 2 == 0) {
c += 1;
}
});
return c;
}

console.log(countEven([1, 2, 6, 9, 4, 0])); //includes 0 as even
```

Question **8**

Complete

Marked out of 10.00

Create an HTML page with an input field, a button, and a result display area. When the button is clicked, a JavaScript function should receive a string as a parameter from the input field. The function should then find the longest word in the string and display it in the result area.

Example:

Given input: 'The quick brown fox jumps over the lazy dog'

If the button is clicked, the result should display 'jumps' as it's the longest word in the input string.

SCRIPT:

```
document.addEventListener('DOMContentLoaded', () => {


const input = document.getElementById('input');
const find_btn = document.getElementById('find_btn');
const result = document.getElementById('result');

// Button event
find_btn.addEventListener('click', () => {
if (input) {
result.value = findLargestWord(input.value);
}
else {
result.value = 'missing input';
}
});

function findLargestWord(inputText) {
let words = inputText.split(' ');
let max = 0;
let largest = '';

words.forEach(word => {
let wordLength = word.length;
if (wordLength > max) {
largest = word;
max = wordLength;
}
});

return largest;
}


});
```

HTML:

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>FIND</title>
<link rel="stylesheet" href="styles.css">
</head>

<body>
```

```
<div class="input-area">
<input type="text" id="input" placeholder="Enter input">
<button id="find_btn">Find Larget Word</button>
<input type="text" id="result" placeholder="awaiting result">
</div>

<script src="script.js"></script>
</body>

</html>
```

Question **9**

Complete

Marked out of 10.00

Create a function that receives a parameter, checks if the parameter is a number, and then prints all the prime numbers up to the given limit from the parameter number.

```
function primesUpto(upto) {
if (typeof upto !== 'number' || upto < 2) {
console.log("invalid input. must be >= 2.");
return;
}

function isPrime(num) {
if (num < 2) return false;
for (let i = 2; i <= Math.sqrt(num); i++) {
if (num % i === 0) return false;
}
return true;
}

//print promes up to upto:
for (let i = 2; i <= upto; i++) {
if (isPrime(i)) {
console.log(i);
}
}
}

console.log(primesUpto(15));
```

Question **10**

Complete

Marked out of 10.00

Write a JavaScript function that takes a string as input and returns the string with each word reversed while maintaining the order of the words.

Example:

Input: Hello world, how are you?

Output: olleH dlrow, woh era ?uoy

```
function revWordsNotOrder(str) {
let words = str.split(' ');
let after_rev = words.map(word => reverseString(word));
// console.log(after_rev);
return after_rev.join(' ');
}

function reverseString(string) {
let newString = '';
for (let i = string.length - 1; i >= 0; i--) {
newString += string.charAt(i);
}
return newString;
}

console.log(revWordsNotOrder("Hello world, how are you?"));
```

Question **11**

Complete

Marked out of 12.00

Create an html page with a button and a list, when clicking on the button, perform a fetch request to the following API - ***https://jsonplaceholder.typicode.com/***
And display 5, random emails from the API in a list, on the html page.

Make sure that every click, you display 5 different emails.

SCRIPT:

```
const fetchBtn = document.getElementById('btn');
const emailList = document.getElementById('list');

fetchBtn.addEventListener('click', () => {
// fetch
emailList.innerHTML = 'Emails wiill display here:';
let fetchedEmails = emailsFetcher();
console.log(fetchedEmails);// pricessed

// show in list
fetchedEmails.forEach(email => {
const listItem = document.createElement('li');
listItem.textContent = email;
emailList.appendChild(listItem);
});
});

async function emailsFetcher() {
try {
const response = await fetch('https://jsonplaceholder.typicode.com/users');

if (!response.ok) {
throw new Error('Network response was not ok');
}

const data = await response.json();

// console.log(data);// all of json

const emails = data
.sort(() => 0.5 - Math.random()) // randomize
.slice(0, 5) // minimize to 5 emails
.map(user => user.email); // select

// console.log(emails);// pricessed

return emails;
} catch (error) {
console.error('Error fetching data:', error);
}
}
```

HTML:

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>FIND</title>
<link rel="stylesheet" href="styles.css">
</head>

<body id="main_cont">
<div>
<button id="btn">FETCH</button>
<ul id="list">LIST:</ul>

</div>
<script src="./email.js"></script>
</body>

</html>
```

Question **12**

Complete

Marked out of 12.00

Write a function that receives an array as a parameter,
Returns an array with only the numbers and sorted.

```
function sortRem(arr) {
// filter alphas
let nums = arr.filter(item => typeof item === 'number');
nums.sort((a, b) => a - b);

return nums;
}
```

◄ Weather app - Exercise

Jump to...

Specifications example