TW-07 TEAM LEAD VERSION

TEAMWORK

CLARUSWAY
WAY TO REINVENT YOURSELF

Meeting Agenda

- Icebreaking
- Questions
- Interview Questions
- Coffee Break
- Coding Challenge
- Video of the week
- Retro meeting
- Case study / project

Teamwork Schedule

Ice-breaking 10m

- Personal Questions (Study Environment, Kids etc.)
- Any challenges (Classes, Coding, studying, etc.)
- Ask how they're studying, give personal advice.
- Remind that practice makes perfect.

Ask Questions 15m

1. What is the function to stop an interval timer?

A. stopTimer
B. clearInterval
C. shutdownTimer
D. clearTimer

Answer: B

2. What are 2 native functions to run code asynchronously in JavaScript ?

A. timeout - setTimeout
B. startInternal - setInterval
C. setTimeout - setInterval
D. interval - setInterval

Answer: C

3. What is the output of the code below?

```
let x = 0;

async function test() {
  x += await 2;
  console.log(x);
}

test();
x += 1;
console.log(x);
```

A. 2 3
B. 0 1
C. 1 2
D. 2 2

Answer: C

Explanation : When test() function called x was 0 and an asyncronous task started as x = x + 2. Once this promise fulfilled result of x will be 2 as 0 + 2 equals to 2.

4. Which method converts JSON data to a JavaScript object?

A. `JSON.fromString();`
B. `JSON.toObject()`
C. `JSON.stringify()`
D. `JSON.parse()`

Answer: D

5. Why would you choose an asynchronous structure for your code?

A. To use ES6 syntax
B. To ensure that parsers enforce all JavaScript syntax rules when processing your code
C. To ensure that tasks further down in your code aren't initiated until earlier tasks have completed
D. To start tasks that might take some time without blocking subsequent tasks from executing immediately

Answer: D

6. What will be logged to the console?

```
console.log("I");
setTimeout(() => {
  console.log("love");
}, 0);
console.log("Javascript!");
```

A.

```
I
love
Javascript!
```

B.

```
I
Javascript!
love
```

C.

```
love
I
Javascript!
```

D. The output may change with each execution of code and cannot be determined.

Answer: B

7. Which statement is true about the "async" attribute for the HTML script tag?

A.It can be used for both internal and external JavaScript code.
B.It can be used only for internal JavaScript code.
C.It can be used only for internal or external JavaScript code that exports a promise.
D. It can be used only for external JavaScript code.

Answer: D

8. When would 'results shown' be logged to the console?

```
let modal = document.querySelector("#result");
setTimeout(function () {
  modal.classList.remove("hidden");
}, 10000);
console.log("Results shown");
```

A. immediately
B. after 10 second
C. after 10000 seconds
D. after results are received from the HTTP request

Answer: A

9. Which Queue Is Executed First?

```
// Callback queue
setTimeout(() => console.log("timeout"), 0);

// Microtask queue
Promise.resolve().then(() => console.log("promise"));
```

A. Callback queue
B. Microtask queue
C. No priority
D. Depends on which one called first

Answer: B

10. How does a function create a closure?

A. It reloads the document whenever the value changes.
B. It returns a reference to a variable in its parent scope.
C. It completes execution without returning.
D. It copies a local variable to the global scope.

Answer: B

11. Consider the following async function and its output. What will be displayed to the console when calling the f() function?

```
async function f() {
  let result = 'first!';
  let promise = new Promise((resolve, reject) => {
    setTimeout(() => resolve('done!'), 1000);
  });

  result = await promise;

  console.log(result);
}

f();
```

A. first!
B. done!
C. JavaScript error
D. Something else

Answer: B

12. What will the output be?

```
Promise.resolve('Success!')
  .then(data => {
    return data.toUpperCase()
  })
  .then(data => {
    console.log(data)
  })
```

A. print "Success!" and "SUCCESS!"
B. print "Success!"
C. print "SUCCESS!"
D. nothing prints

Answer: C

Interview Questions 15m

1. Can you explain what asynchronous programming is?

Answer: Asynchronous programming is a form of programming that allows for tasks to be completed out of order. This means that a program can start a task and then move on to other tasks before the first one is completed. This can be helpful in situations where a task might take a long time to complete, but the program doesn't need to wait for it to finish before moving on.

2. How can you convert a callback-based function to a Promise-based function?

Answer: To convert a callback-based function to a Promise-based function, you can wrap the callback-based function with a Promise and pass the resolve and reject functions to it. You call the resolve function when the asynchronous operation is successful and the reject function when the operation fails. You return the Promise in the function and resolve or reject the Promise based on the result of the asynchronous operation. It is possible for callers of the Promise-based function to use the then method to access the resolved value and the catch method to handle errors.

3. How does async/await help with performance and scalability?

Answer :Async/await can help improve performance and scalability by allowing your application to do other work while it is waiting for a task to complete. This can help avoid bottlenecks and keep your application responsive. Additionally, using async/await can help reduce the overall amount of code needed to be written, making your application easier to maintain.

4. Can you explain the difference between an async function and a regular function in JavaScript?

Answer : Async functions are functions that allow you to use the await keyword to wait for a promise to resolve before continuing execution of the function. Regular functions do not have this ability, and will instead execute the code inside of them immediately.

Coding Challenge 15m

- JS-CC-07 iOS Calculator

Coffee Break 10m

Video of the Week 10m

- What the heck is the event loop anyway?

Case study/Project 15m

- JS-05 Crypto-Coin-App

Retro Meeting on a personal and team level 10m

Ask the questions below:

- What went well?
- What could be improved?
- What will we commit to do better in the next week?

Closing 5m

- Next week’s plan
- QA Session
