✓ **Congratulations! You passed!**

Grade received 100%   **To pass** 80% or higher

Retake the assignment in **7h 59m**

Go to next item

Module 4 Quiz

Latest Submission Grade 100%

1. Identify true statements about asynchronous functions in JavaScript. (Select all that apply.)   1 / 1 point

☐ Asynchronous functions in JavaScript provide ways of performing several tasks at once.

☑ There is more than one API for using JavaScript asynchronously.

✓ **Correct**
Asynchronous functions allow for more sophisticated web applications.

☐ Asynchronous functions in JavaScript must always complete successfully.

☐ Since JavaScript is single threaded, asynchronous JavaScript is a workaround.

☑ Asynchronous functions allow users to interact with the web application while waiting for the asynchronous function to complete.

✓ **Correct**
Asynchronous functions allow for more sophisticated web applications.

2. Identify true statements about XMLHttpRequest API. (Select all that apply.)   1 / 1 point

☑ This API was cumbersome to work with so many developers used libraries to make working with the API easier.

✓ **Correct**
Understanding the history of JavaScript can help when working with legacy code.

☐ This API fell out of favor because it only worked with data in XML format.

☑ This API has largely been replaced by the more robust Fetch API in modern JavaScript.

✓ **Correct**
Understanding the history of JavaScript can help when working with legacy code.

☐ This API was first supported in Netscape Navigator 4.0.

☑ This API was originally supported in Microsoft Internet Explorer browser.

✓ **Correct**
Understanding the history of JavaScript can help when working with legacy code.

☐ You have always needed a browser plugin in order to properly use the XMLHttpRequest API.

3. Identify true statements about the Fetch API. (Select all that apply.)   1 / 1 point

☑ The Fetch API is a modern, more robust, replacement for the older XMLHttpRequest API.

✓ **Correct**
The Fetch API provides a modern, up-to-date way of getting and sending data asynchronously in JavaScript.

☐ The Fetch API doesn't provide ways of accessing remote URLs without a browser plugin.

☐ The Fetch API is an older method for writing asynchronous functions and was replaced by the XMLHttpRequest API.

☑ The Fetch API makes use of promises, which are either pending, fulfilled or rejected.

✓ **Correct**
The Fetch API provides a modern, up-to-date way of getting and sending data asynchronously in JavaScript.

☐ The Fetch API was first implemented in jQuery to make asynchronous functions easier to write.

4. Identify advantages of using the async and await keywords over using the promises and chaining then statements. (Select all that apply.)   `1 / 1 point`

☐ The keywords async and await are more robust than promises.

☐ Chained then statements are more prone to network errors

☐ Resources are fetched faster using async and await.

☑ Functions that use the async and await keywords create code that is easier to read and maintain.

> ⊘ **Correct**
> Async and await are syntactic sugar that makes working with promises easier.

☑ The functions that use async and await are less likely to lead to callback hell.

> ⊘ **Correct**
> Async and await are syntactic sugar that makes working with promises easier.

5. Identify potential issues when working with asynchronous functions in JavaScript. (Select all that apply.)   `1 / 1 point`

☑ It is easy to end up in "callback hell" where it is hard to follow the flow of the function.

> ⊘ **Correct**
> Asynchronous functions can be confusing, but they can lead to robust web applications when handled properly.

☑ Using the await keyword in a function that does not have the async keyword will not work properly.

> ⊘ **Correct**
> Asynchronous functions can be confusing, but they can lead to robust web applications when handled properly.

☐ Asynchronous functions only work in certain web browsers.

☑ You can't assume that retrieving external data will always succeed.

> ⊘ **Correct**
> Asynchronous functions can be confusing, but they can lead to robust web applications when handled properly.

☐ Modern JavaScript is multithreaded and that has made asynchronous functions irrelevant.

☐ Asynchronous functions will slow down your web application.

☑ The async and await keywords can make your code look more like non asynchronous functions, leading to potential confusion.

> ⊘ **Correct**
> Asynchronous functions can be confusing, but they can lead to robust web applications when handled properly.

6. When will the callback function inside the setTimeout function below run?   `1 / 1 point`

```
setTimeout(function () { console.log('this has run') }, 1000);
```

○ It will run exactly 1 second later.

○ The callback function will fire 1000 seconds later.

⦿ The callback function will run after at least a second has elapsed.

○ There is a syntax error and the callback function will never run.

> ⊘ **Correct**
> Yes, the call to the console log goes into the queue and runs when it makes it into the stack will be at least a second later, but it could be longer.

7. What is the difference between the stack and the queue within the JavaScript event model?   `1 / 1 point`

○ The stack is free unstructured memory, and the queue is where events line up to be processed by the JavaScript engine.

○ There is no difference between the stack and the queue. The terms are interchangeable.

○ The stack is where asynchronous events wait to be processed, while the queue is where synchronous events are processed.

⦿ The stack is the set of procedures the JavaScript engine is working on completing, one at a time until the stack is empty. The queue is where asynchronous tasks wait until they are returned and enter the stack.

> ⊘ **Correct**
> Yes, the queue is only used for asynchronous events.

8. If you're repeating an asynchronous task on a timer, why is it important to use recursion and setTimeout instead of setInterval?

`1/1 point`

○ Using setTimeOut is better supported across web browsers than setInterval.

○ Using setTimeOut with a recursion makes code more readable and maintainable.

◉ Using setTimeout with recursion ensures that the repetition happens after the asynchronous task has completed.

○ Using setTimeOut and recursion is more efficient than using setInterval.

⊘ **Correct**
You don't want to start repeating the task until the first one has completed.

---

9. What does this line of code do?

`1/1 point`

```
$('#receivingDiv').load('http://website.com/somedata.php');
```

○ This code uses the JavaScript load method to prepare data for transmission to a webpage.

○ This code is using plain JavaScript to load somedata.php into a variable called receivingDiv.

○ This code has a syntax error and will not run without creating an error.

◉ This is using the jQuery library to load data from a specific URL endpoint and place it into an element in the DOM with an ID set to receivingDiv.

⊘ **Correct**
jQuery is recognizable by the leading $, which is a shortcut that stands in for the jQuery keyword.

---

10. Identify true statements about the jQuery ajax method. (Select all that apply.)

`1/1 point`

☐ The jQuery ajax method is obsolete and will not function anymore.

☐ The jQuery ajax method is faster than using the JavaScript Fetch API.

☑ The jQuery ajax method is really using the older XMLHttpRequest API under the hood.

⊘ **Correct**
The jQuery ajax method still works, but it's better to use the Fetch API, where possible.

☑ The jQuery ajax method takes an object as an argument which includes key/value pairs.

⊘ **Correct**
The jQuery ajax method still works, but it's better to use the Fetch API, where possible.

☑ The jQuery ajax method can send data to the server using POST.

⊘ **Correct**
The jQuery ajax method still works, but it's better to use the Fetch API, where possible.

☐ The jQuery ajax method can get data from the server, but it can not send data to the server.

---

11. In the contact form script, what does the focus() method do?

`1/1 point`

○ The focus method is a way for JavaScript to make sure the data is handled quickly.

◉ If the user has neglected to put required data into a field, on submission of the form, the focus method will automatically place the cursor in that field.

○ The focus method keeps users focused on completing the form.

○ The focus method is a jQuery method and can not be used without the jQuery library loaded.

⊘ **Correct**
It is good practice to make it as easy as possible for users to fill out information correctly on forms.