✓ **Congratulations! You passed!**

Grade received 97.43% **To pass** 80% or higher

Go to next item

Module 3 Quiz

Latest Submission Grade 97.43%

1. Identify true statements concerning JavaScript and the DOM. Select all that apply.

1 / 1 point

☐ JavaScript only recently developed features for accessing elements on webpages.

☐ Some elements on web pages can not be accessed by JavaScript, so they should be avoided.

☐ JavaScript sees the webpage as just a string of characters.

☑ JavaScript includes a number of document methods and properties that allow developers to access content within the DOM and manipulate it.

✓ **Correct**
This relationship between JavaScript and the DOM is one of the features that sets this language apart from other programming languages.

☑ JavaScript recognizes a parent/child relationship between various elements on the web page.

✓ **Correct**
This relationship between JavaScript and the DOM is one of the features that sets this language apart from other programming languages.

☑ JavaScript sees a web page as a tree of branches with nodes.

✓ **Correct**
This relationship between JavaScript and the DOM is one of the features that sets this language apart from other programming languages.

2. What is the dot, or period character (.), used for in JavaScript? Select all that apply.

1 / 1 point

☐ To signify the end of a logical statement.

☑ To separate a property from an object.

✓ **Correct**
The dot or period character in JavaScript plays a special role in the language.

☐ To concatenate one statement to the next.

☑ To separate an object from its properties.

✓ **Correct**
The dot or period character in JavaScript plays a special role in the language.

☐ To signify classes.

3. Select all the true statements about properties and methods.

1 / 1 point

☑ Properties are distinguished by the dot and then followed with an equal sign and then a value.

✓ **Correct**
It is easy to get confused between properties and methods, but as you use them, you will become more clear on which is which.

☑ Methods have parentheses and sometimes parameters are passed into the parentheses.

✓ **Correct**
It is easy to get confused between properties and methods, but as you use them, you will become more clear on which is which.

☐ Document methods only work when combined with event handlers.

☐ Document properties include parentheses at the end of the property name.

☐ Document properties and methods do the same things and are interchangeable.

4. Identify the document methods that return an array of elements, even if only one element or zero elements are in the array. Select all that apply.

1 / 1 point

□ document.innerHTML

☑ document.getElementsByClassName();

✓ **Correct**
If you use this method, the returned result is always an array.

☑ document.getElementsByTagName();

✓ **Correct**
If you use this method, the returned result is always an array.

□ document.querySelector()

☑ document.querySelectorAll()

✓ **Correct**
If you use this method, the returned result is always an array.

□ document.getElementByID();

5. Identify all the event properties. Select all that apply.

1 / 1 point

□ className

□ innerHTML

☑ onclick

✓ **Correct**
Event properties are fine to use, but they are used less frequently and it is more likely that you will see event methods instead.

□ addEventListener()

☑ onmouseover

✓ **Correct**
Event properties are fine to use, but they are used less frequently and it is more likely that you will see event methods instead.

☑ onsubmit

✓ **Correct**
Event properties are fine to use, but they are used less frequently and it is more likely that you will see event methods instead.

6. Identify all the true statements about event methods. Select all that apply.

0.8333333333333334 / 1 point

☑ You have to initialize the event module in JavaScript before using event methods.

⊗ **This should not be selected**
Please revisit the lectures related to: **Capturing Events with JavaScript**.

☑ You can remove an event listener with the removeEventListener() method.

✓ **Correct**
Combined with DOM methods and properties, event methods become a powerful tool for creating interactive web pages.

☑ Event methods can be attached to DOM elements, the document or even the window itself.

✓ **Correct**
Combined with DOM methods and properties, event methods become a powerful tool for creating interactive web pages.

□ Event methods can only be accessed through a computer and not mobile devices.

□ You can only attach one event method on an element at a time.

☑ addEventListener() is a commonly used event method.

✓ **Correct**
Combined with DOM methods and properties, event methods become a powerful tool for creating interactive web pages.

7. Identify the problems with the event handler below. Select all that apply.

1 / 1 point

```
var myBtn = document.querySelector('button');

myBtn.addeventListener('click', function(event){

    evt.preventDefault();

    alert('you pressed me!');

});
```

☐ myBtn is a reserved word and can not be used.

☐ There is an extra set of parentheses.

☑ The event method is spelled incorrectly.

✓ **Correct**

☑ The event object was passed in as "event", but "evt" was used.

✓ **Correct**

☐ There are no errors in the code.

8. Why are anonymous functions frequently used with event handlers? Select all that apply.

1 / 1 point

☐ Most event handlers will only work with anonymous functions.

☐ Anonymous functions perform better than named functions.

☑ The syntax is shorter and a little more concise.

✓ **Correct**
It is frequently the case that functions in event handlers are used only in that context. The purpose of a named function is to make it reusable.

☑ Because these functions are usually only used in the context of an event, and rarely reused elsewhere in the program.

✓ **Correct**
It is frequently the case that functions in event handlers are used only in that context. The purpose of a named function is to make it resuable.

☐ Named functions can cause problems with variable scope.

9. If you create a new variable inside a function and forget to use the "var" keyword, what does JavaScript do?

1 / 1 point

● JavaScript creates the variable and that variable will belong to the window, or be in the global scope (unless the use strict directive has been invoked).

○ JavaScript will create the variable, but that variable will only be available inside the function.

○ JavaScript will create the variable, but it will hold a null value.

○ JavaScript always throws an error.

✓ **Correct**
Accidentally putting variables in the global scope can lead to unexpected values inside variables.

10. What does the "use strict" directive do?

1 / 1 point

○ JavaScript skips the variable declaration and continues with the program.

○ JavaScript creates the variable but gives it a null value.

○ JavaScript creates the variable, but makes it local in scope.

● JavaScript will throw an error if there is a variable declared in a function without the use of a keyword such as var, let or const.

✓ **Correct**
This is a best practice for keeping accidental global variable declaration from happening.

11. With the addition of the "let" keyword, how many different scopes does JavaScript recognize and what are they?

1 / 1 point

○ Because JavaScript was written to be a simple scripting language, there is only one global scope for variables.

○ JavaScript has two levels of scope, global and local. The "let" keyword does the same thing as the "var" keyword

○ JavaScript has two levels of scope, global and local. The "let" keyword does the same thing as the "var" keyword.

○ JavaScript has three scopes:

1. Global,
2. The window, and
3. Local

● JavaScript has three scopes:

1. Global scope, or the window,
2. Local scope, within a function,
3. Block level scope within a block if the "let" keyword is used.

✓ Correct
The addition of block level scope will help minimize scope problems in JavaScript programs.

12. Which of the following statements are true about hoisting in JavaScript? Select all that apply.

0.8333333333333334 / 1 point

☑ If you define a variable using the "const" or "let" keyword, it will not be hoisted.

✓ Correct
Avoiding hoisting is one more way to keep JavaScript from having variables with unexpected values.

□ Hoisting is only an issue if the program is affecting the DOM.

□ Variables defined with the keyword "let" are hoisted.

☑ Hoisting affects which variables are recognized by JavaScript, based on where in the script file they are defined.

✓ Correct
Avoiding hoisting is one more way to keep JavaScript from having variables with unexpected values.

□ Hoisting in JavaScript can cause problems because it is possible for a program to try to access the value of a variable before that variable has a value assigned to it.

□ Hoisting is good because it allows developers to write sloppy code that still works.

You didn't select all the correct answers

13. Identify the errors in the code below.

1 / 1 point

```
const counter = 1;

for( let i=0; i<10; i++){

    console.log(`this is loop number ${counter}`);

    counter = counter +1;

}
```

○ There are no errors.

● The expression **counter = counter + 1;** will generate an error.

○ The "let" keyword can not be used inside a loop.

○ The loop will run forever because the condition to end it is never met.

○ The variable "counter" is not defined because of hoisting.

✓ Correct
The variable "counter" can not be changed within the program because it was defined with the "const" keyword.