W03 Learning Activities

Score for this attempt: **15** out of 15

Submitted Jul 17 at 4:06pm

This attempt took 2 minutes.

**Question 1**

**3 / 3 pts**

I completed all the assigned learning activities for this week.

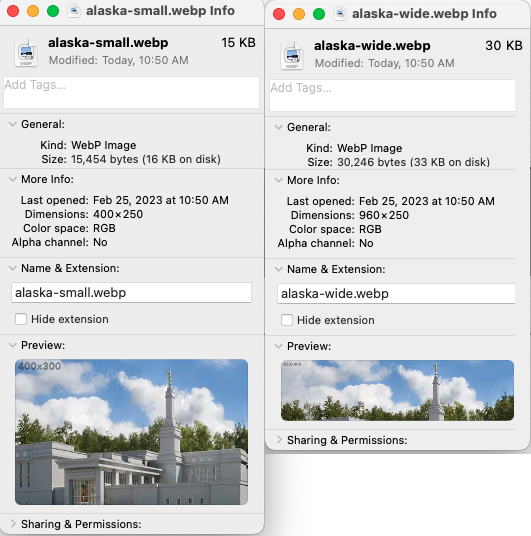
True

False

**Question 2**

**1 / 1 pts**

A member of your group has used these two images in their responsive image assignment. What feedback would you give them?  
Check all that apply.



The file size (KB) for these images are way too big.

The files are saved using the image file types that are not supported on the web.

Your images are too large, try reducing them using an image editing program.

Everything looks great, keep up the good work.

**Question 3**

**1 / 1 pts**

A student has this code for their responsive image. What feedback would you give them?  
Check all the apply.

<picture>  
 <source srcset="images/hero-small.webp" media="(max-width: 500px)">  
 <source srcset="images/hero-medium.webp" media="(max-width: 1000px)">  
 <img srcset="images/hero-large.webp" alt="Hero Photo" width="500" height="250">  
</picture>

Your source references are missing alt attributes.

You forgot to add the units (px) to the width and height attribute values.

Good job, the code looks good.

You are not allowed to use max-width with the <source> element's media attribute.

You have the two source element lines backwards.

**Question 4**

**1 / 1 pts**

One of your peers is confused as to why these JavaScript statements do not work as intended. They report that when a valid chapter is entered into the HTML input field with an id of "favchap" the message on the screen is wrong saying that the favorite chapter is something like [object HTMLInputElement]. And when they do not enter anything into the input field, the message is still outputting the "Thank you. your favorite chapter is [object HTMLInputElement]".  
What advice would you give them?  
Check all the apply.

const message = document.querySelector('#message');  
const input = document.querySelector('#favchap');  
if (input !== "") {  
 message.innerHTML = `Thank you. Your favorite chapter is ${input}.`;  
} else {  
 message.innerHTML = `Please enter a chapter.`;  
}

Template literal strings (`...`) cannot be used to produce output to the screen.

Use single quotes and not double quotes in the if block condition.

The HTML input element must have an id of "input" to match the variable identifier used.

Reference the value property of the #favchap input element, not just the element.

**Question 5**

**1 / 1 pts**

Another student has reviewed your favorite chapter application and commented that you must use **innerHTML** versus **textContent** when providing the list item's content.

li.innerHTML = item;

versus

li.textContent = item;  
  
What would be your reply to them?  
Mark all that apply.

Using innerText would be the best option in this case.

That would be true if the item variable contained HTML tags.

Using the textContent property is the only option when working with HTML list item content.

Either property will work in this case given that the item variable is a simple string.

**Question 6**

**1 / 1 pts**

JavaScript uses type conversion to coerce any value to a Boolean in contexts that require it, such as conditionals and loops. A **falsy** value is a value that is considered false when encountered in a Boolean context. Which of the following are JavaScript falsy values?

999

'', "", or ``

"False"

null

NaN (not a number)

undefined

"0"

0

**Question 7**

**1 / 1 pts**

Which of the following methods can be used to add elements to the end of an **array**in JavaScript?

unshift()

push()

add()

spice()

**Question 8**

**1 / 1 pts**

Given the following code snippet, which of the following is the output to the console?

let scores = [100, 72, 83, 94, 88, 87];  
let accumulator = 0;  
let count = 0;  
scores.forEach(score => {  
 if (score > 87) {  
 accumulator += score;  
 count ++;  
 }  
});  
if ( count > 0 ) {  
 console.log(accumulator/count);   
} else {  
 console.log("No scores reported.");  
}

92.25

No scores reported.

87.5

94

**Question 9**

**1 / 1 pts**

Which of the following code expressions produces the last element (index) of any **array**?

array.length - 1

array.length

array.lastIndex()

array.lastIndex

**Question 10**

**1 / 1 pts**

Rewrite (refactor) this function declaration by using an arrow function instead of a function declaration.

function fullName(first, last) {  
 return `${first} ${last}`;  
}

const fullName = (first, last) => `${first} ${last}`;

const fullName = function (first, last) { `${first} ${last}`;}

function = first + " " + last`

const generateFullName = (first, last) {first.concat(' ', last);}

**Question 11**

**1 / 1 pts**

In a JavaScript function declaration, which of the following is the keyword used to declare a function?

functionBlock

def

func

function

**Question 12**

**1 / 1 pts**

*Finish this statement.*  
The                               pseudo-class is useful for declaring global CSS variables and represents the <html> element and is identical to the selector html, except that its specificity is higher.

global

specific

html

root

**Question 13**

**1 / 1 pts**

Which of the following statements regarding pseudo-elements in CSS is correct?

Pseudo-elements allow you to create virtual elements that do not exist in the HTML document.

Pseudo-elements are used to style elements based on their position in the document tree.

Pseudo-elements are used to select and style elements based on their attributes.

Pseudo-elements are represented by double colons (::) and are used to style specific parts of an element.