Solutions to Quick Checks

Table of Contents

[Quick Check Answers 2](#_Toc49426986)

[Quick Check 1 2](#_Toc49426987)

[Quick Check 2 2](#_Toc49426988)

[Quick Check 3 4](#_Toc49426989)

# Quick Check Answers

Quick Check 1

1. Show how to create an array named foodMenu containing the text strings “Breakfast”, “Lunch”, and “Dinner” as an array literal and using the new Array() object constructor.

let foodMenu = ["Breakfast", "Lunch", "Dinner"];  
let foodMenu = new Array("Breakfast", "Lunch", "Dinner");

**Feedback**: An array literal is defined using the expression [*value1*, *value2*, *value3*, …]. An array can also be defined using the new Array() object constructor as new Array(*value1*, *value2*, *value3*, …)

1. Provide a command to return the size of the array customerOrders.

customerOrders.length

**Feedback**: To determine an array’s current size, use the property:

array.*length*

1. Provide a command to return the 10th entry in the customerOrders array.

customerOrders[9]

**Feedback**: The first array element has an index value of zero, so the tenth element would have an index value of 9.

1. Provide the expression to reference to fifth inline image in the document.

document.images[4]

**Feedback**: The first element in a HTML Element collection has an index value of zero, so the fifth element would have an index value of 4.

1. Provide the expression to reference the third element belonging to the blogpost class An HTML file contains the following tag

document.getElementsByClassName("blogpost")[2]

**Feedback:** The first element in a HTML Element collection has an index value of zero, so the third element would have an index value of 2.

Quick Check 2

1. Show how to use a while loop to write the HTML code <td>*counter*</td> for integer values of *counter* ranging from 1 to 100 by 1.

let i = 1;  
while(i <= 100) {  
 document.write("<td>" + i + "</td>");  
 i++;  
}

**Feedback:** In a while loop, a command block is executed while a given condition is true but stops once that condition is no longer true. The syntax of a while loop is:  
while (*condition*) {  
 *statements*;  
}   
where *condition* is conditional expression that either true or false and *statements* are the statements within the command block which are repeatedly executed as long as that conditional expression is true.

1. What is the most important difference between a while loop and a do while loop?

The do while loop is evaluated after the command block has been executed at least once.

**Feedback:** The while loop is an example of a pretest loop in that the stopping condition is evaluated before each iteration of the command block. Because of this, it is possible that command block will never be executed. Another type of program loop, called the do while loop, is a posttest loop in which the stopping condition is evaluated after the command block has been executed at least once.

1. Provide code for a for loop that writes the following HTML code:  
   <td>3</td> <td>6</td> <td>12</td> <td>24</td> <td>48</td> <td>96</td>

for (let i = 3; i < 100; i \*= 3) {  
 document.write("<td>" + i + "</td>");  
}

**Feedback:** Answers may vary, but the important point is that the counter starts with an initial value 3 and increases by the assignment operator i \*= 3.

1. What JavaScript method can be used to insert HTML code just after an element’s opening tag?

document.insertAdjacentHTML("afterBegin", *text*)

**Feedback:** Use the following insertAdjacentHTML() method to insert additional content into an element:  
*element*.insertAdjacentHTML(*position*, *text*)   
where *element* is the element into which the new content is inserted, *position* is the location of the new content, and *text* is the text of the content.

1. What JavaScript method can be used to apply a function to each element of an array without writing a program loop?

forEach()

Feedback: The forEach() method which calls a function for each element within an array:

array.forEach(callback, thisArg)

where array is a reference to array, callback is the function called for each array element, and thisArg is an optional parameter containing a value that can be passed to the callback function.

Quick Check 3

1. Provide the code for an if statement that displays an alert window with the message "You passed with an A" if the value of the exam variable is greater than 90.

if (exam > 90) {  
 window.alert("You passed with an A");  
}

Feedback: With only one outcome, you can create an if statement with a single command block.

1. Provide the code for an if else statement that displays an alert window with the message "You passed with an A" if the value of the exam variable is greater than 90 and the message "Not an A" if otherwise.

if (exam > 90) {  
 window.alert("You passed with an A");  
} else {  
 window.alert("Not an A");  
}

Feedback: With two possible outcomes, you need an else condition with its own command block if the if condition is not true.

1. Provide the code for an else if statement that displays the message "You passed with an A" if exam is greater than 90, else if exam is greater than 80 the browser displays the message "You passed with a B", else if exam is greater than 70, the message "You passed with an C" is displayed, else the message "You did not pass" is displayed.

if (exam > 90) {  
 window.alert("You passed with an A");  
} else if (exam > 80) {  
 window.alert("You passed with a B");  
} else if (exam > 70) {  
 window.alert("You passed with a C");  
} else {  
 window.alert("You did not pass");  
}

Feedback: With multiple possible outcomes, you need an else if condition with each command block if the if condition is not true.

1. Provide the general code for a browser test that tests whether or not the browser supports the findIndex() method when applied to an array named xValues.

A falsy value is a value that is treated in comparison operations as the Boolean value false. The six falsy values in JavaScript are "", -0, 0, NaN, null, and undefined.

if (xValues.findIndex) {  
 *// Statements using the findIndex() method*  
} else {  
 // Statements using an alternate solution  
}

Feedback: The general syntax is:  
if (*feature*) {

*statements that use the feature*

} else {

*statements that use replacement code*

} ,

where *feature* is JavaScript object, property, or method that should be tested for browser support.

1. How should you write the code for a switch statement to allow more than one condition to be run by the JavaScript interpreter?

Omit the break command from the different case statements.

Feedback: The break statement, marking the end of each case, is an optional keyword that halts the execution of the switch statement once a matching case has been found. For programs in which more than one label might match the expression, omit the break statements and the JavaScript interpreter will continue moving through the case labels, running all statements in which a match has been found.