*const atomicInteger* = *createAtomicInteger*(0);  
*let sumDisassembled* = 0;  
*let sumNotDisassembled* = 0;  
  
*function createAtomicInteger*(initialValue) {  
 *let* value = initialValue || 0;  
 *return* {  
 get() {  
 *return* value;  
 },  
  
 increment() {  
 *return* ++value;  
 },  
 };  
}  
  
*document*.addEventListener('change', *exampleJavaScriptQuestion*);  
  
*function exampleJavaScriptQuestion*() {  
 *const* list = ["Explain event delegation.",  
 "Explain how this works in JavaScript.",  
 "Can you give an example of one of the ways that working with this has changed in ES6?",  
 "Explain how prototypal inheritance works.",  
 "What's the difference between a variable that is: null, undefined or undeclared?",  
 "How would you go about checking for any of these states?",  
 "What is a closure, and how/why would you use one?",  
 "What language constructions do you use for iterating over object properties and array items?",  
 "Can you describe the main difference between the Array.forEach() loop and Array.map() methods and why you would pick one versus the other?",  
 "What's a typical use case for anonymous functions?",  
 "What's the difference between host objects and native objects?",  
 "Explain the difference between: function Person(){}, var person = Person(), and var person = new Person()?",  
 "Explain the differences on the usage of foo between function foo() {} and var foo = function() {}",  
 "Can you explain what Function.call and Function.apply do? What's the notable difference between the two?",  
 "Explain Function.prototype.bind.",  
 "What's the difference between feature detection, feature inference, and using the UA string?",  
 "Explain \"hoisting\".",  
 "Describe event bubbling.",  
 "Describe event capturing.",  
 "What's the difference between an \"attribute\" and a \"property\"?",  
 "What are the pros and cons of extending built-in JavaScript objects?",  
 "What is the difference between == and ===?",  
 "Explain the same-origin policy with regards to JavaScript.",  
 "Why is it called a Ternary operator, what does the word \"Ternary\" indicate?",  
 "What is strict mode? What are some of the advantages/disadvantages of using it?",  
 "What are some of the advantages/disadvantages of writing JavaScript code in a language that compiles to JavaScript?",  
 "What tools and techniques do you use debugging JavaScript code?",  
 "Explain the difference between mutable and immutable objects.",  
 "What is an example of an immutable object in JavaScript?",  
 "What are the pros and cons of immutability?",  
 "How can you achieve immutability in your own code?",  
 "Explain the difference between synchronous and asynchronous functions.",  
 "What is event loop?",  
 "What is the difference between call stack and task queue?",  
 "What are the differences between variables created using let, var or const?",  
 "What are the differences between ES6 class and ES5 function constructors?",  
 "Can you offer a use case for the new arrow => function syntax? How does this new syntax differ from other functions?",  
 "What advantage is there for using the arrow syntax for a method in a constructor?",  
 "What is the definition of a higher-order function?",  
 "Can you give an example for destructuring an object or an array?",  
 "Can you give an example of generating a string with ES6 Template Literals?",  
 "Can you give an example of a curry function and why this syntax offers an advantage?",  
 "What are the benefits of using spread syntax and how is it different from rest syntax?",  
 "How can you share code between files?",  
 "Why you might want to create static class members?",  
 "What is the difference between while and do-while loops in JavaScript?",  
 "What is a promise? Where and how would you use promise?",  
 "Discuss how you might use Object Oriented Programming principles when coding with JavaScript.",  
 "Coding questions: Make this work: duplicate([1,2,3,4,5]); // [1,2,3,4,5,1,2,3,4,5]",  
 "Coding questions: Create a for loop that iterates up to 100 while outputting \"fizz\" at multiples of 3, \"buzz\" at multiples of 5 and \"fizzbuzz\" at multiples of 3 and 5",  
 "Coding questions: What will be returned by each of these?\n console.log(\"hello\" || \"world\")\n console.log(\"foo\" && \"bar\")",  
 "Coding questions: Write an immediately invoked function expression (IIFE)"];  
 *console*.log(list.length);  
 *let* disassembled = *document*.getElementById("disassembled").checked;  
 *let* noDisassembled = *document*.getElementById("noDisassembled").checked;  
 *let* resultDisassembled = *sumDisassembled* / list.length;  
 *let* resultNotDisassembled = *sumNotDisassembled* / list.length;  
 event.preventDefault();  
 *console*.log(*sumDisassembled*);  
 *console*.log(*sumNotDisassembled*);  
 *if* (*atomicInteger*.get() === 0) {  
 *document*.getElementById("firstQuestion").innerHTML = "";  
 }  
 *switch* (*true*) {  
 *case* (disassembled && noDisassembled && *atomicInteger*.get() < list.length - 1):  
 *document*.querySelector('.card-header-title').innerHTML = `YOU HAVE TO CHOOSE ONE STATUS GAME OVER!\n  
 Percentage of understanding questions: ${resultDisassembled.toFixed(3)} Percentage of not understanding questions: ${resultNotDisassembled.toFixed(3)}`;  
 *updateCheckAfterHalfSeconds*();  
 *reloadAfter4Seconds*();  
 *break*;  
 *case* (disassembled && !noDisassembled && *atomicInteger*.get() < list.length - 1):  
 *document*.querySelector('.card-header-title').innerHTML = `${list[*atomicInteger*.increment()]}`;  
 *document*.getElementById('questionNumber').innerHTML = `Question ${*atomicInteger*.get()}`;  
 *sumDisassembled*++;  
 *updateCheckAfterHalfSeconds*();  
 *break*;  
 *case* (!disassembled && noDisassembled && *atomicInteger*.get() < list.length - 1):  
 *document*.querySelector('.card-header-title').innerHTML = `Okay, here's another question ${list[*atomicInteger*.increment()]}`;  
 *document*.getElementById('questionNumber').innerHTML = `Question ${*atomicInteger*.get()}`;  
 *sumNotDisassembled*++;  
 *updateCheckAfterHalfSeconds*();  
 *break*;  
 *case* (*atomicInteger*.get() >= list.length - 1):  
 *document*.querySelector('.card-header-title').innerHTML = `GAME OVER\n  
 Percentage of understanding questions: ${resultDisassembled.toFixed(3)} Percentage of not understanding questions: ${resultNotDisassembled.toFixed(3)}`;  
 *updateCheckAfterHalfSeconds*();  
 *reloadAfter4Seconds*();  
 *break*;  
 }  
}  
  
*function reloadAfter4Seconds*() {  
 *return new Promise*(resolve => {  
 *setTimeout*(() => {  
 *window*.location.reload();  
 }, 4000);  
 });  
}  
  
*function updateCheckAfterHalfSeconds*() {  
 *return new Promise*(resolve => {  
 *setTimeout*(() => {  
 *document*.getElementById("disassembled").checked = *false*;  
 *document*.getElementById("noDisassembled").checked = *false*;  
 }, 500);  
 });  
}