**JavaScript Fundamentals**

1. History JavaScript

HTML dan CSS adalah bahasa markup. Bahasa markup digunakan untuk mendeskripsikan dan mendefinisikan elemen-elemen di dalam dokumen. JavaScript adalah bahasa pemrograman. Bahasa pemrograman digunakan untuk mengomunikasikan instruksi ke mesin. Bahasa pemrograman dapat digunakan untuk mengontrol perilaku mesin dan mengekspresikan algoritme.

JavaScript pertama kali dikembangkan oleh Brendan Eich pada tahun 1995 ketika ia bekerja di Netscape Communications Corporation. Awalnya dikenal dengan nama Mocha, kemudian LiveScript, dan akhirnya disebut JavaScript. Tujuannya adalah untuk memberikan interaktivitas pada halaman web. Saat ini, JavaScript adalah salah satu bahasa pemrograman paling populer di dunia, digunakan untuk mengembangkan aplikasi web, aplikasi seluler, permainan, dan banyak lagi.

1. The JavaScript Console

Konsol JavaScript adalah alat debugging yang tersedia di browser web modern seperti Google Chrome, Mozilla Firefox, dan lain-lain. Konsol ini memungkinkan Anda untuk mengevaluasi kode JavaScript, memeriksa variabel, memecahkan masalah, dan melakukan debugging pada halaman web. Anda dapat membukanya dengan menekan tombol F12 pada keyboard atau klik kanan pada halaman web dan pilih opsi "Inspect" atau "Inspect Element".

1. Developer Tools on Different Browsers

* Chrome DevTools adalah sekumpulan alat pembuat web dan debug yang terpasang di Google Chrome. Gunakan DevTools untuk mengulangi, men-debug, dan membuat profil situs Anda.
* Alat Pengembang Firefox memungkinkan Anda memeriksa, mengedit, dan men-debug HTML, CSS, dan JavaScript di desktop dan seluler. Selain itu, Anda dapat mengunduh versi Firefox yang disebut Edisi Pengembang Firefox yang disesuaikan untuk pengembang, menampilkan fitur Firefox terbaru dan alat pengembang eksperimental.
* Jika Anda menggunakan Internet Explorer, Anda dapat mengakses alat pengembang F12 hanya dengan menekan F12. Fitur bervariasi antar versi, tetapi mulai dari Internet Explorer 8 tetap cukup konsisten.
* Microsoft Edge memperkenalkan peningkatan baru yang hebat pada alat pengembang F12 yang terlihat di Internet Explorer. Alat baru dibangun di TypeScript, dan selalu berjalan, jadi tidak perlu memuat ulang. Selain itu, dokumentasi alat pengembang F12 kini tersedia sepenuhnya di GitHub .
* Untuk setiap pengguna Mac, Safari menyertakan Web Inspector, alat canggih yang memudahkan untuk memodifikasi, men-debug, dan mengoptimalkan situs web untuk performa puncak dan kompatibilitas di kedua platform.
* Opera adalah peramban web yang cepat, ramping, dan andal.

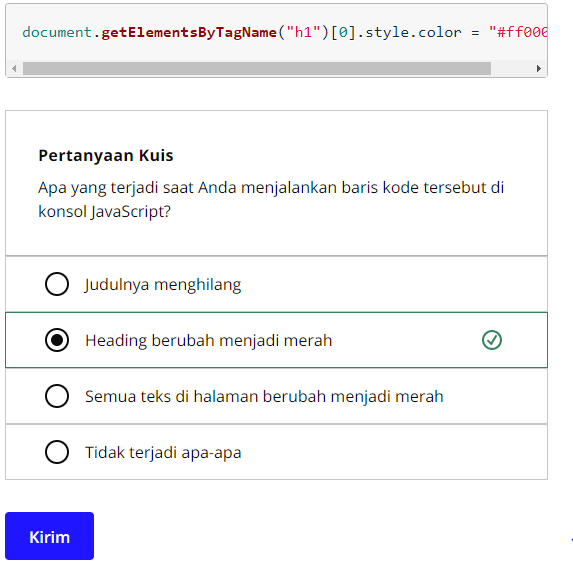
1. console.log

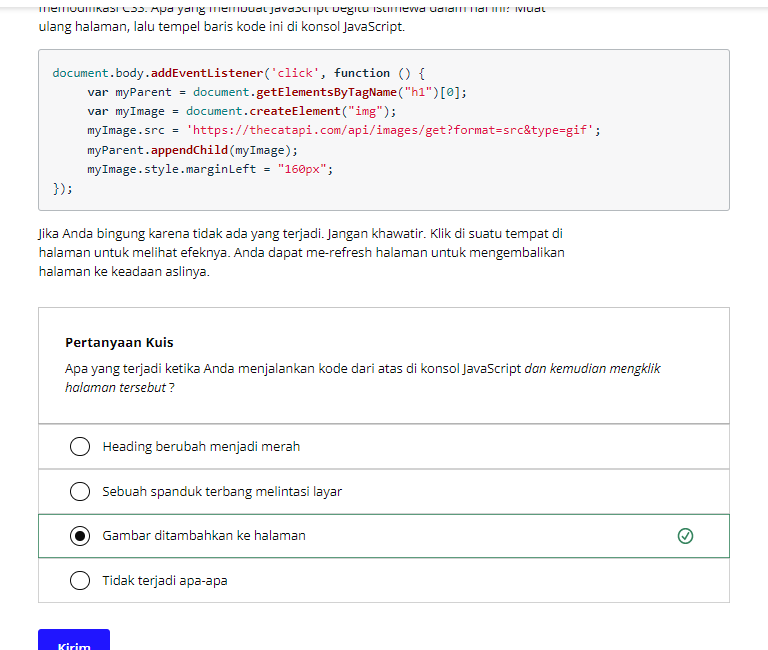
console.logdigunakan untuk menampilkan konten ke konsol JavaScript. Contoh Jalankan kode berikut di konsol:

code: console.log("hiya friend!");

Cetakan: "hai teman!"

1. Demo JavaScript





1. Summary

Ringkasan dibangun di dalam mesin JavaScript, artinya bahwa konstruksi sintaksis ringkasan tersebut adalah fitur bawaan dari bahasa pemrograman JavaScript yang sudah terintegrasi ke dalam mesin JavaScript itu sendiri. Oleh karena itu, kita dapat menggunakan ringkasan di dalam kode JavaScript tanpa perlu menginstal atau menambahkan pustaka atau library tertentu. Fitur ini memudahkan pengembang dalam membuat kode JavaScript dengan lebih efisien dan mudah dibaca. Selain itu, penggunaan ringkasan juga membantu meminimalkan jumlah kode yang perlu ditulis, sehingga dapat meningkatkan produktivitas pengembang.

1. Intro to Data Types

Tipe data adalah konsep penting dalam pemrograman yang mengacu pada jenis nilai yang dapat disimpan dan diolah oleh program komputer. Dalam bahasa pemrograman, tipe data digunakan untuk memastikan bahwa variabel atau nilai yang digunakan dalam program memiliki format yang benar dan dapat diolah dengan benar oleh program.

Dalam bahasa pemrograman JavaScript, terdapat beberapa tipe data dasar yang meliputi:

Number: tipe data number digunakan untuk merepresentasikan angka, baik itu bilangan bulat atau desimal.

* String: tipe data string digunakan untuk merepresentasikan teks atau karakter.
* Boolean: tipe data boolean hanya memiliki dua nilai, yaitu true atau false.
* Null: tipe data null digunakan ketika suatu variabel tidak memiliki nilai.
* Undefined: tipe data undefined digunakan ketika suatu variabel belum diberikan nilai.
* Object: tipe data object digunakan untuk merepresentasikan objek dalam JavaScript.
* Array: tipe data array digunakan untuk merepresentasikan kumpulan nilai dalam satu variabel.

1. Angka

Mendefinisikan angka dalam JavaScript sebenarnya cukup sederhana. Tipe data Angka mencakup bilangan bulat positif atau negatif, serta desimal. Memasukkan nomor ke konsol akan mengembalikannya langsung kepada Anda.

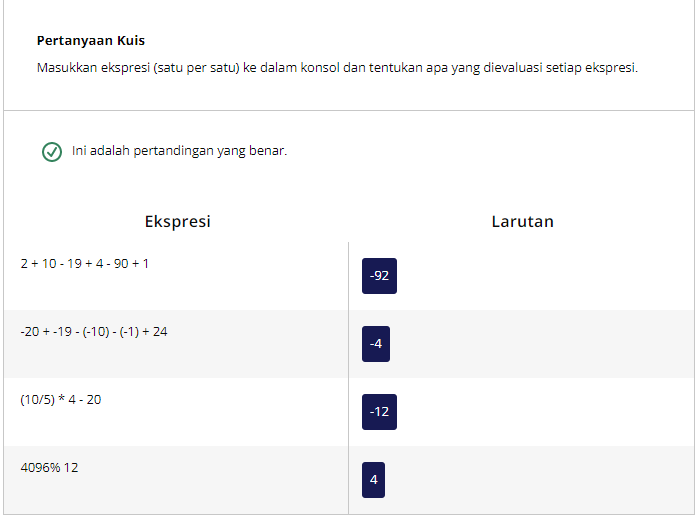
Code 3

Pengembalian: 3

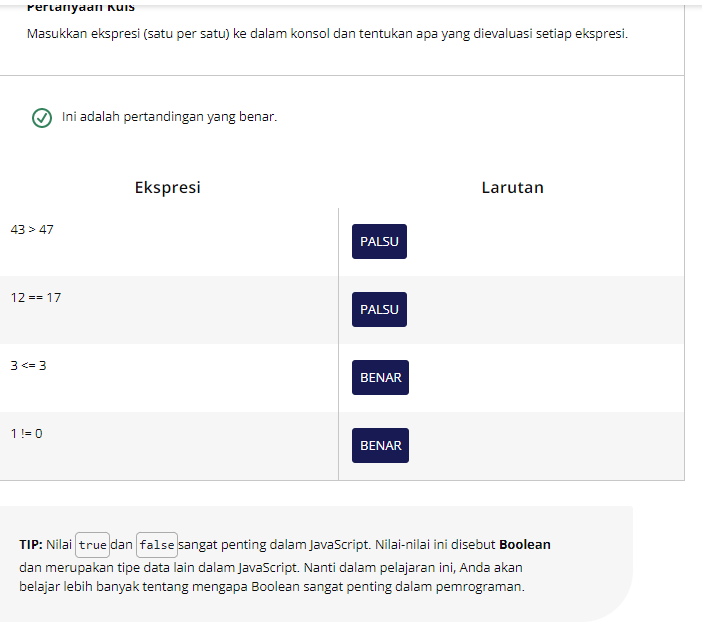
Operasi aritmatika Anda juga dapat melakukan perhitungan dengan angka dengan cukup mudah. Pada dasarnya ketikkan ekspresi seperti yang Anda ketikkan di kalkulator.

Code 3 + 2.1

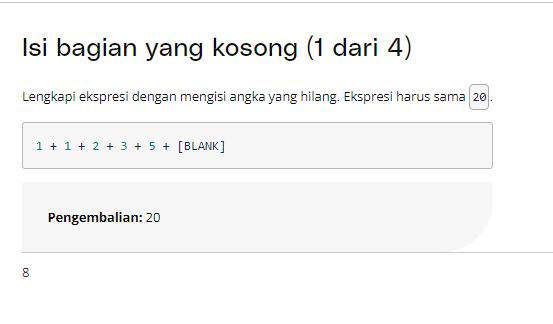
Pengembalian: 5.1

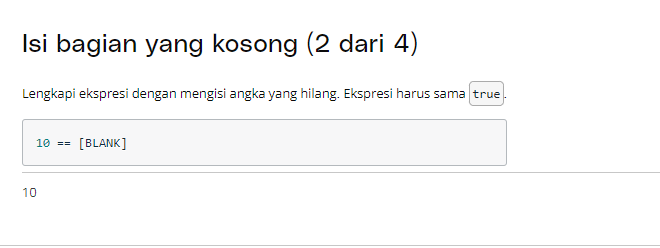


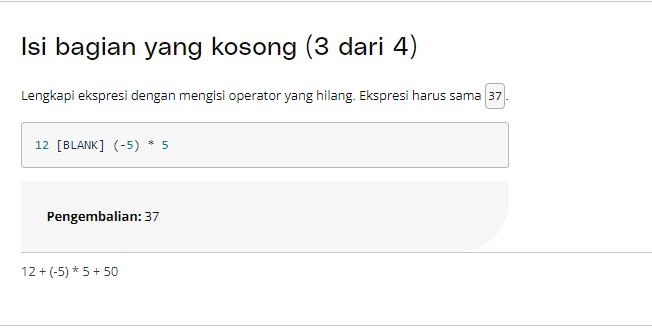
|  |  |
| --- | --- |
| Operator | Arti |
| < | Kurang dari |
| > | Lebih besar dari |
| <= | Kurang dari atau Sama dengan |
| >= | Lebih dari atau sama dengan |
| == | Sama dengan |
| != | Tidak sama dengan |

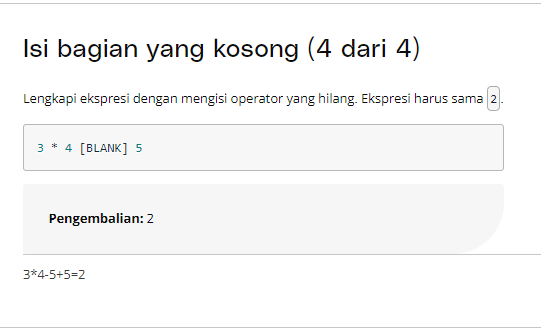


1. Nomor dan Operator – Latihan





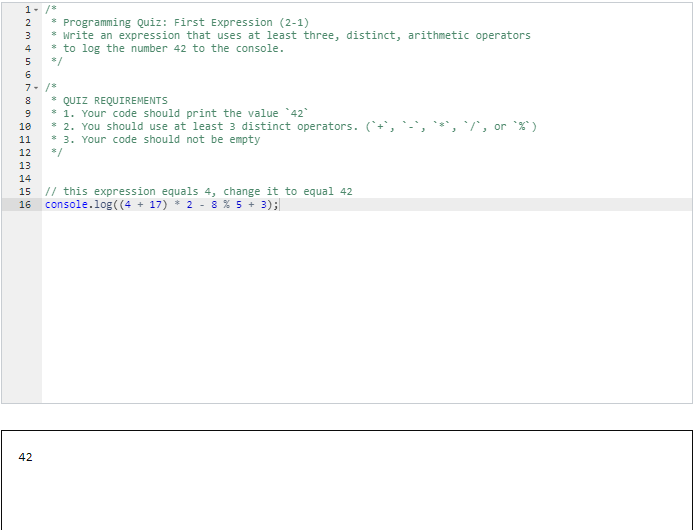




1. Comments

Anda dapat menggunakan komentar untuk membantu menjelaskan kode Anda dan membuatnya lebih jelas. Dalam JavaScript, komentar ditandai dengan garis miring ganda //. Apa pun yang ditulis pada baris yang sama setelahnya //tidak akan dieksekusi atau ditampilkan. Agar komentar mencakup beberapa baris, tandai awal komentar Anda dengan garis miring dan bintang, lalu sertakan komentar Anda di dalam bintang dan garis miring / \*…\* /.

1. Quiz: First Expression (2-1)

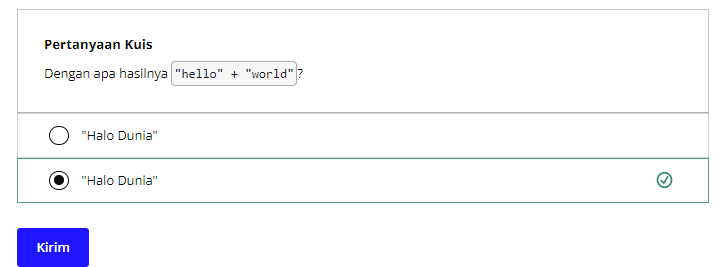


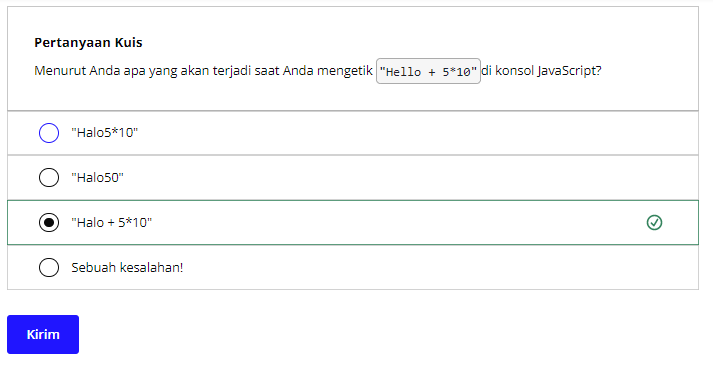
1. String

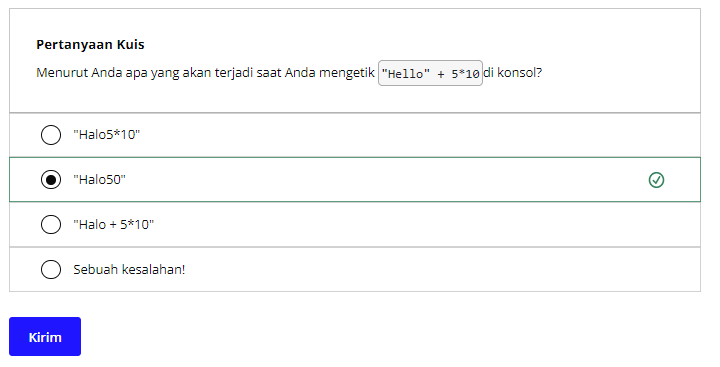
String pada JavaScript adalah tipe data yang digunakan untuk merepresentasikan teks atau karakter. String pada JavaScript dapat dibuat dengan menggunakan tanda kutip tunggal (') atau ganda ("), atau dengan menggunakan backtick ( ) untuk membuat template literals. Contohnya ‘h’, “h”, “123”.

1. String Concatenation

String adalah kumpulan karakter yang diapit di dalam tanda kutip ganda atau tunggal. Anda dapat menggunakan string untuk merepresentasikan data seperti kalimat, nama, alamat, dan lainnya.







1. Variabel

Variabel pada JavaScript adalah wadah untuk menyimpan nilai atau referensi ke nilai. Variabel dapat dideklarasikan dengan menggunakan kata kunci let, const, atau var.

Contohnya:

let nama = "John";

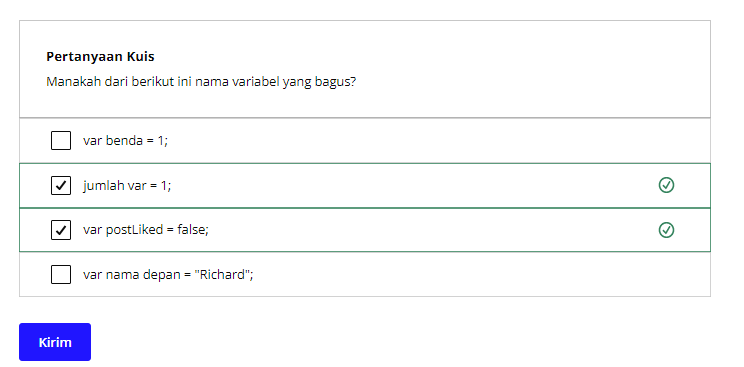
let umur = 25;

const PI = 3.14;

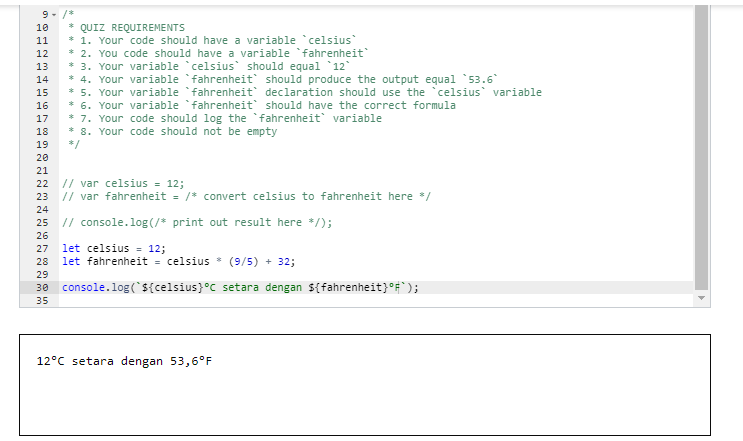
const hariDalamSeminggu = 7;

var x = 5;

var y = "Hello";



1. Mengonversi Suhu (2-2)



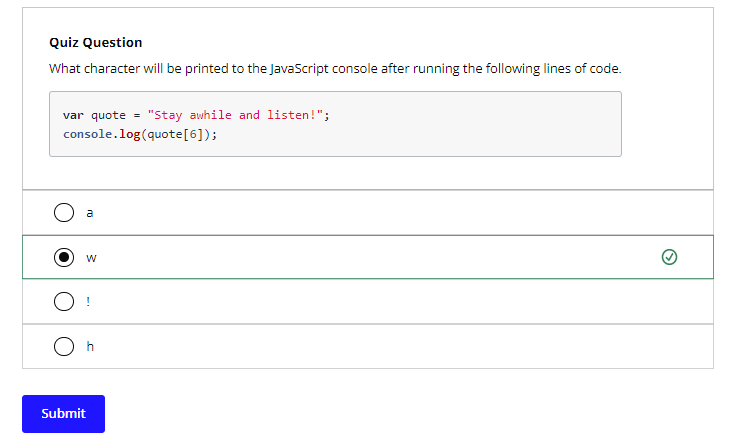
1. String Index

Indeks pada string di JavaScript dimulai dari 0 dan diurutkan dari kiri ke kanan. Dalam string, setiap karakter dapat diakses dengan menggunakan indeksnya. Indeks pada string pada JavaScript dapat diakses dengan menggunakan tanda kurung siku [].

Contohnya:

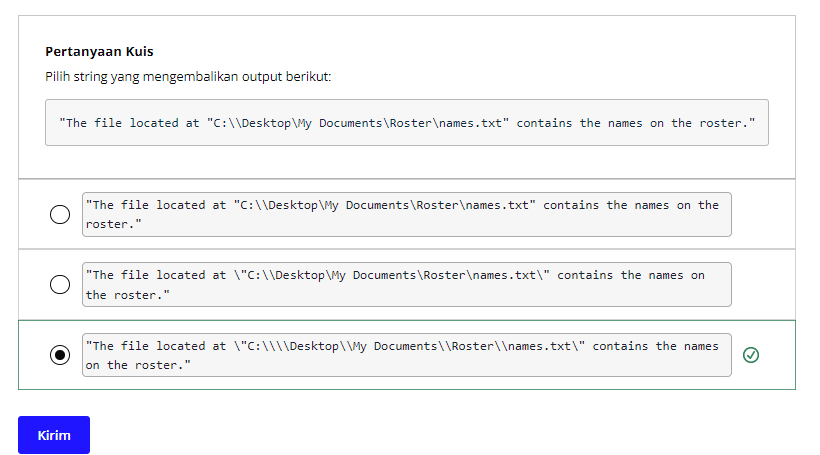
var name = "James";

name[0];



1. Escaping Strings

|  |  |
| --- | --- |
| Kode | Karakter |
| \\ | \ (garis miring terbalik) |
| " | '' (kutipan ganda) |
| ' | ' (kutipan tunggal) |
| \N | garis baru |
| \T | tab |



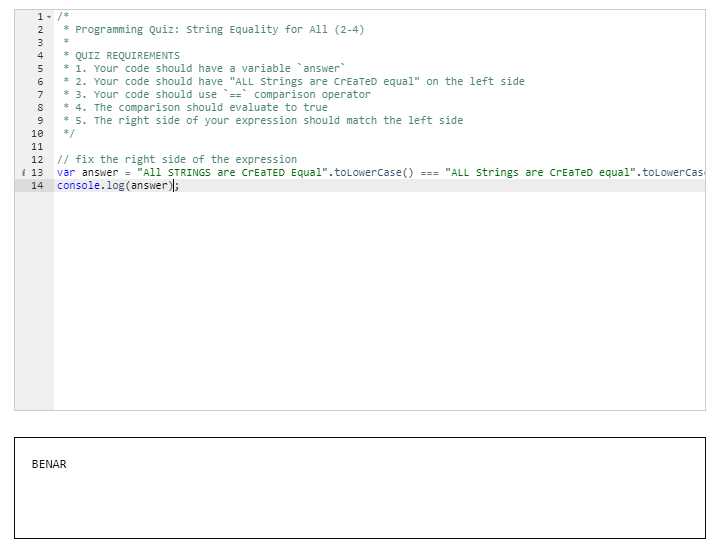
1. Comparing Strings

Cara lain untuk bekerja dengan string adalah dengan membandingkannya. Anda telah melihat operator perbandingan ==dan !=saat Anda membandingkan angka untuk kesetaraan. Anda juga dapat menggunakannya dengan string! Sebagai contoh, mari bandingkan string "Yes"dengan "yes".

1. Quiz: Favorite Food (2-3)



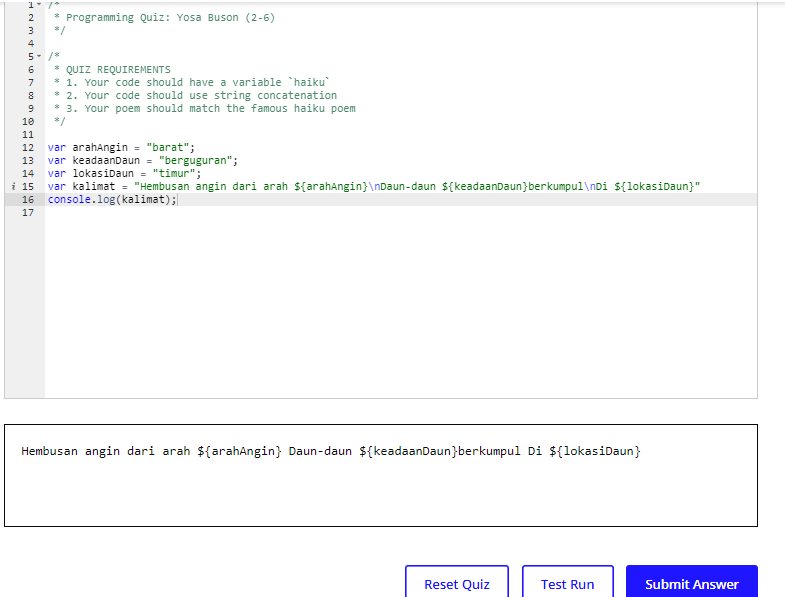
1. String Equality for All (2-4)



1. Quiz: All Tied Up (2-5)



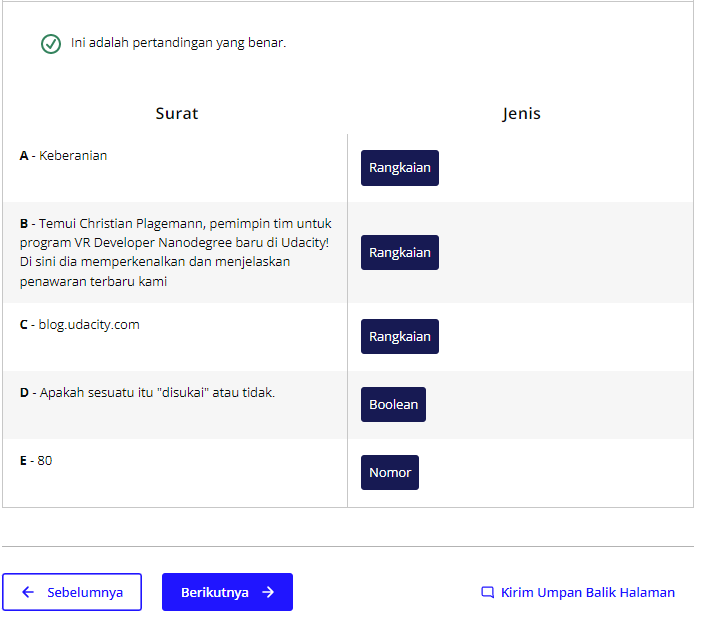
1. Quiz: Yosa Buson (2-6)



1. Booleans

Boolean adalah tipe data dalam pemrograman yang hanya memiliki dua nilai, yaitu true atau false. Fungsi dari boolean adalah untuk menyatakan nilai kebenaran dari suatu pernyataan atau kondisi.

1. Quiz: Facebook Post (2-7)



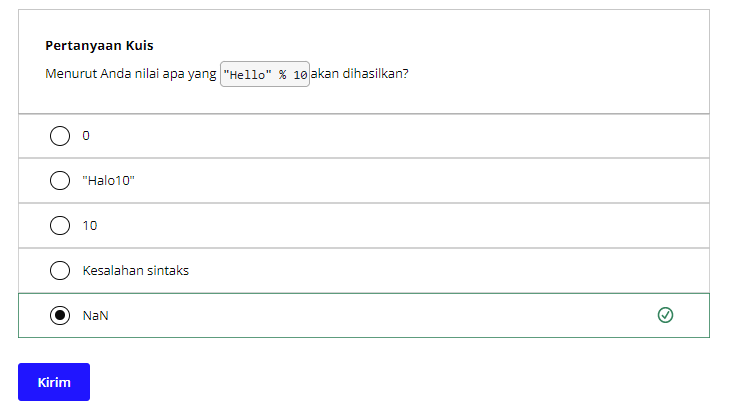
1. Null, Undefined, and NaN

NaNsingkatan dari "Not-A-Number" dan sering dikembalikan menunjukkan kesalahan dengan operasi angka. Misalnya, jika Anda menulis beberapa kode yang melakukan perhitungan matematika, dan perhitungan gagal menghasilkan angka yang valid, NaNmungkin dikembalikan.

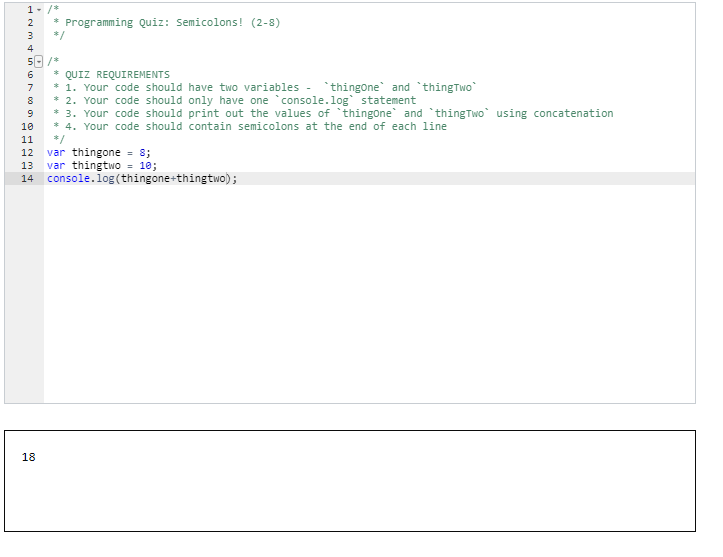


1. Equality

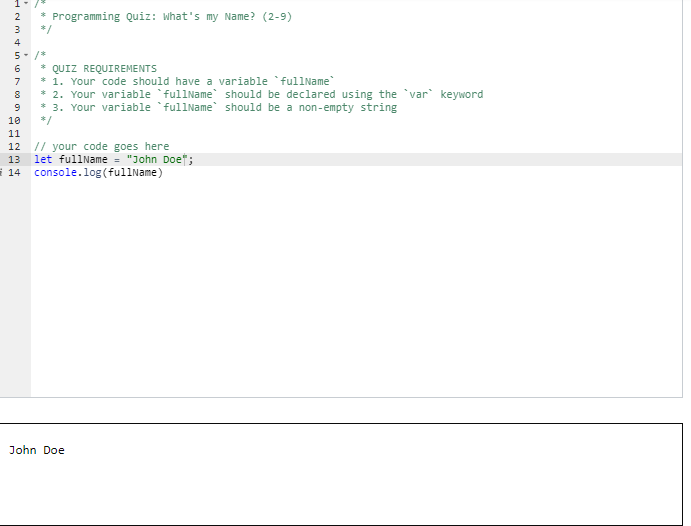
Dalam kasus perbandingan reguler, operan di kedua sisi operator ==pertama-tama diubah menjadi angka, sebelum dibandingkan. Oleh karena itu, a ' ', false, dan 0semuanya dianggap sama. Demikian pula, a '1'dan 1juga dianggap sama. Jika kita tidak ingin mengonversi operan, sebelum melakukan perbandingan, kita harus menggunakan strict comparison === , yang dijelaskan di bawah ini.



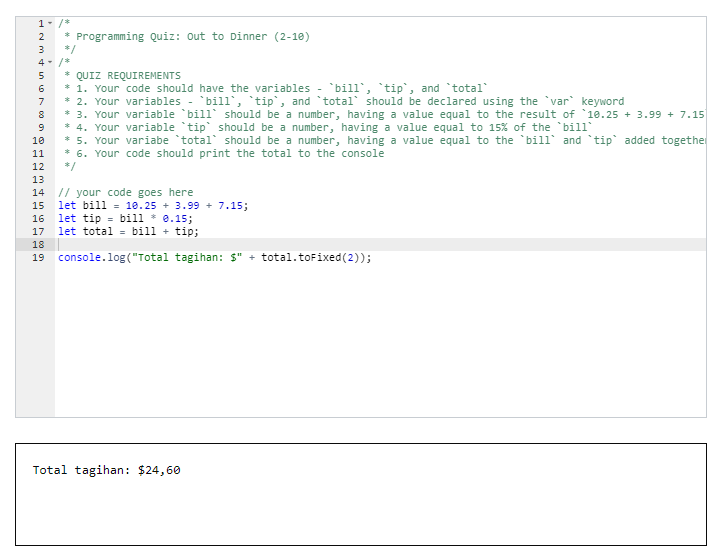
1. Quiz: Semicolons! (2-8)



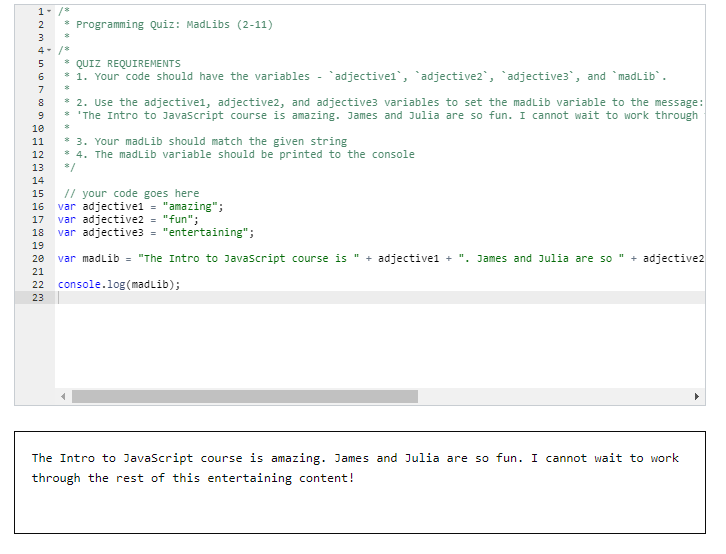
1. Quiz: What's my Name? (2-9)



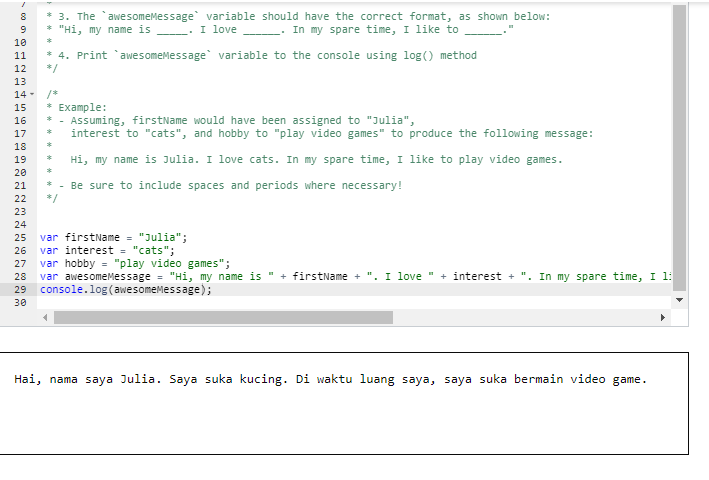
1. Quiz: Out to Dinner (2-10)



1. Quiz: Mad Libs (2-11)



1. One Awesome Message (2-12)



1. Lesson 2 Summary

Javascript Data Types

* Number
* Strings
* Booleans
* Null undefined

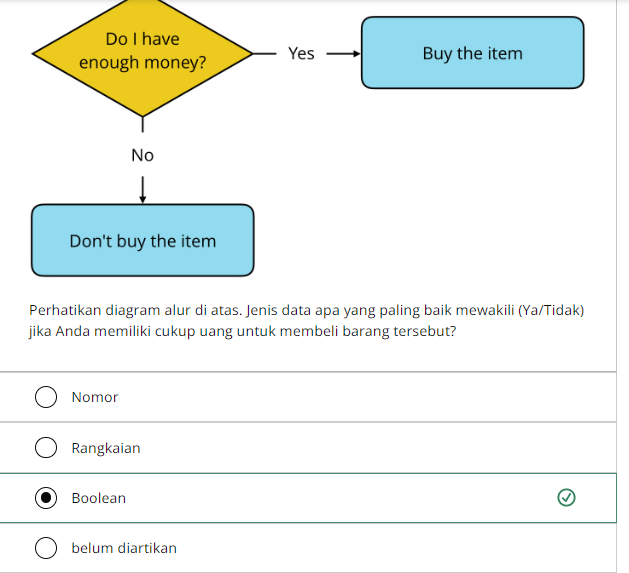
Storing Data in Variables

* Basic Operations

1. Intro to Conditionals

"Conditionals" adalah salah satu konsep dasar dalam pemrograman. Mereka memungkinkan program untuk membuat keputusan berdasarkan kondisi yang diberikan. Dalam JavaScript, terdapat beberapa bentuk conditional, tetapi yang paling umum adalah if statement.

1. Quiz: Flowcharts (3-1)





1. Flowchart to Code

Flowchart adalah sebuah diagram yang menggambarkan alur suatu proses atau program. Flowchart dapat digunakan sebagai alat bantu dalam merancang program dan juga dalam memahami alur suatu program.

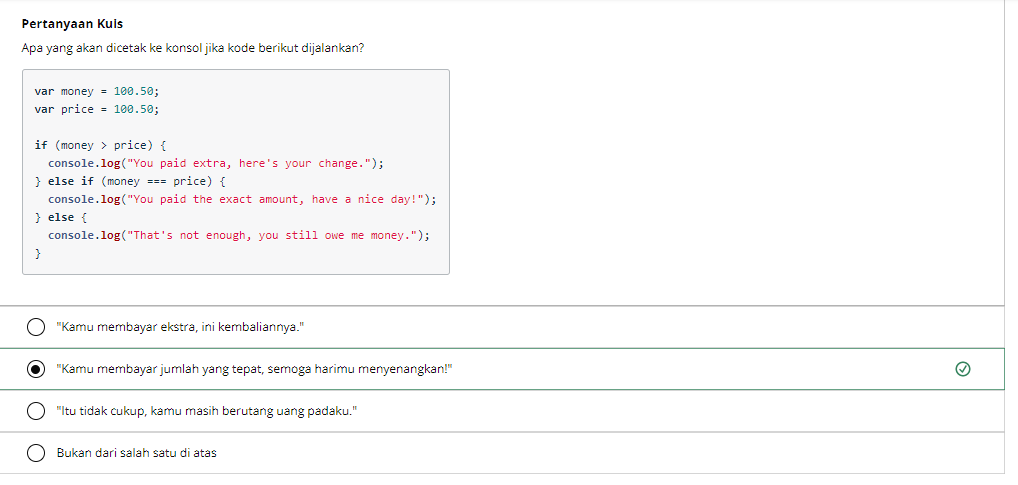
Untuk mengimplementasikan flowchart ke dalam bentuk kode program, kita dapat mengikuti langkah-langkah berikut:

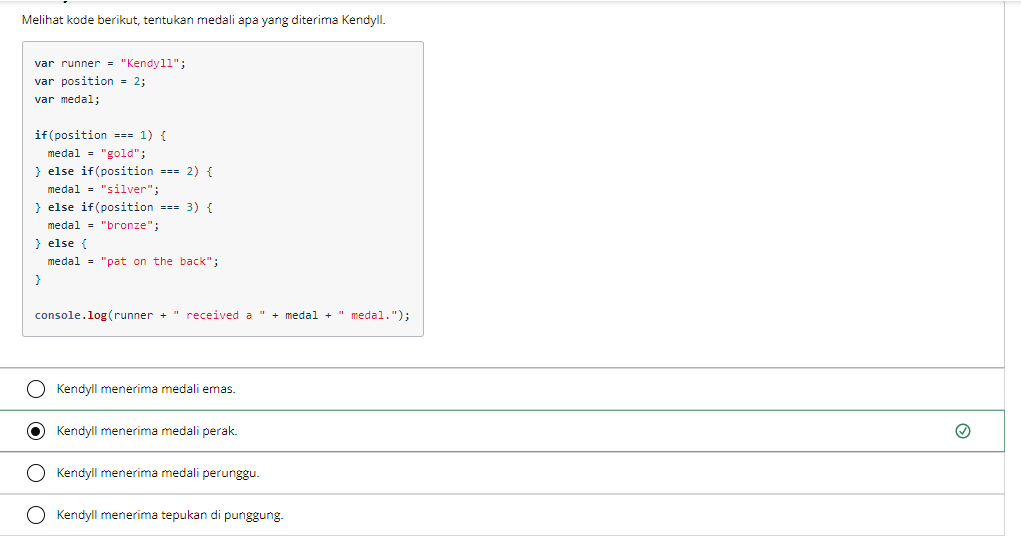
* Identifikasi kebutuhan program
* Gunakan kondisional
* Gunakan variable
* Gunakan loop
* Implementasikan operasi matematika
* Debugging
* Dokumentasi

1. If...Else Statements

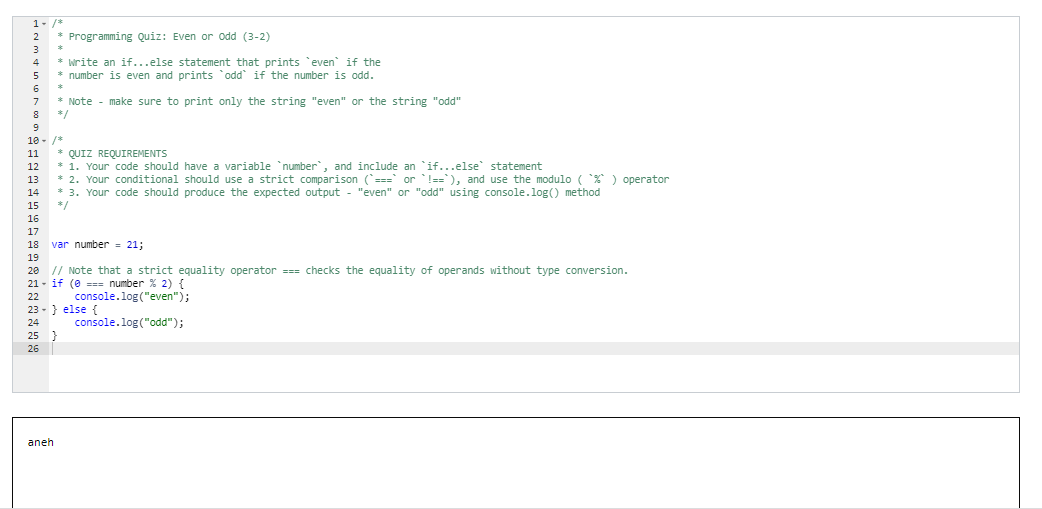
If...else statements adalah jenis kondisional di dalam bahasa pemrograman yang memungkinkan program untuk mengambil keputusan berdasarkan kondisi tertentu. Kondisi yang dievaluasi harus berupa Boolean, yaitu bernilai true atau false. Jika kondisi bernilai true, maka blok kode di dalam if statement akan dieksekusi. Jika kondisi bernilai false, maka blok kode di dalam else statement akan dieksekusi.

1. Else If Statements





1. Quiz: Even or Odd (3-2)



1. Quiz: Musical Groups (3-3)



1. Quiz: Murder Mystery (3-4)



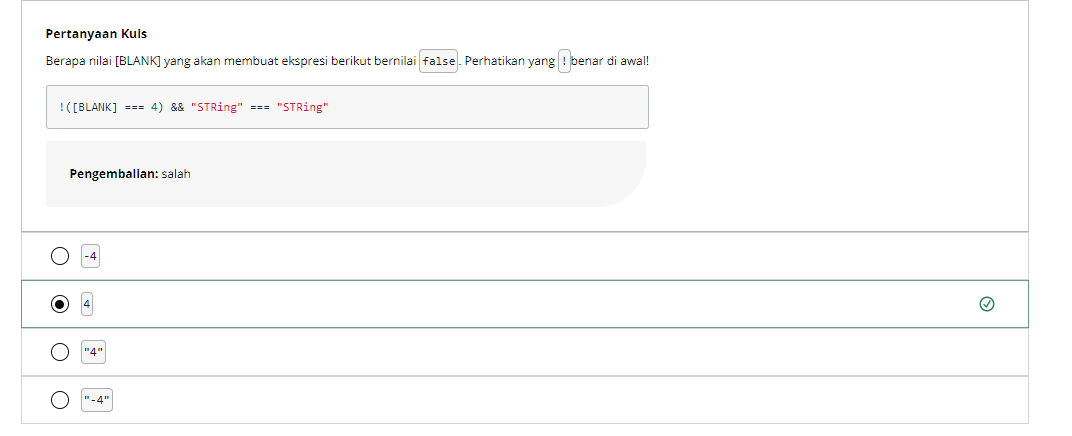
1. More Complex Problem

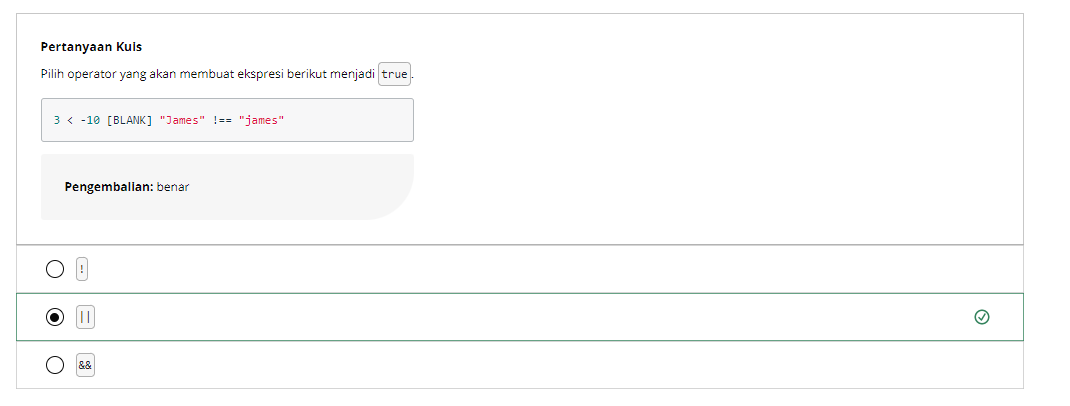
Masalah yang lebih kompleks di bidang JavaScript bisa mencakup beberapa area yang berbeda, termasuk:

* Aplikasi web yang kompleks: Aplikasi web modern yang kompleks memerlukan banyak kode JavaScript untuk menjalankan fungsionalitas yang berbeda, seperti manipulasi DOM, membuat animasi, pengiriman permintaan jaringan, dan banyak lagi.
* Optimisasi kinerja: Dalam aplikasi web yang besar, kinerja bisa menjadi masalah besar. Ini melibatkan optimisasi kode JavaScript untuk meningkatkan waktu respon dan meminimalkan waktu muat, serta menerapkan teknik seperti caching dan kompresi.
* Pengembangan lintas platform: Masalah ini muncul ketika Anda ingin membuat aplikasi yang dapat berjalan pada banyak platform, seperti desktop, web, dan seluler. Ini memerlukan pengetahuan tentang berbagai teknologi, seperti React Native, Angular, dan Cordova.
* Debugging masalah: Dalam kode JavaScript yang kompleks, debugging dapat menjadi sulit karena kompleksitas kode. Hal ini memerlukan keterampilan dan alat untuk membantu memecahkan masalah.
* Integrasi dengan layanan pihak ketiga: Integrasi dengan layanan pihak ketiga seperti API, database, dan platform lainnya, bisa menjadi masalah yang kompleks dan memerlukan pemahaman tentang protokol dan format data yang berbeda.
* Keamanan: Masalah keamanan bisa sangat kompleks di aplikasi web modern. Hal ini melibatkan melindungi data pengguna, mencegah serangan jaringan, dan menjaga kode JavaScript tetap aman dan terlindungi.

1. Logical Operators

Simbolnya &&adalah operator logika AND, dan digunakan untuk menggabungkan dua ekspresi logika menjadi satu ekspresi logika yang lebih besar. Jika kedua ekspresi yang lebih kecil adalah benar , maka seluruh ekspresi dievaluasi menjadi benar . Jika salah satu dari ekspresi yang lebih kecil adalah false , maka seluruh ekspresi logis adalah false .







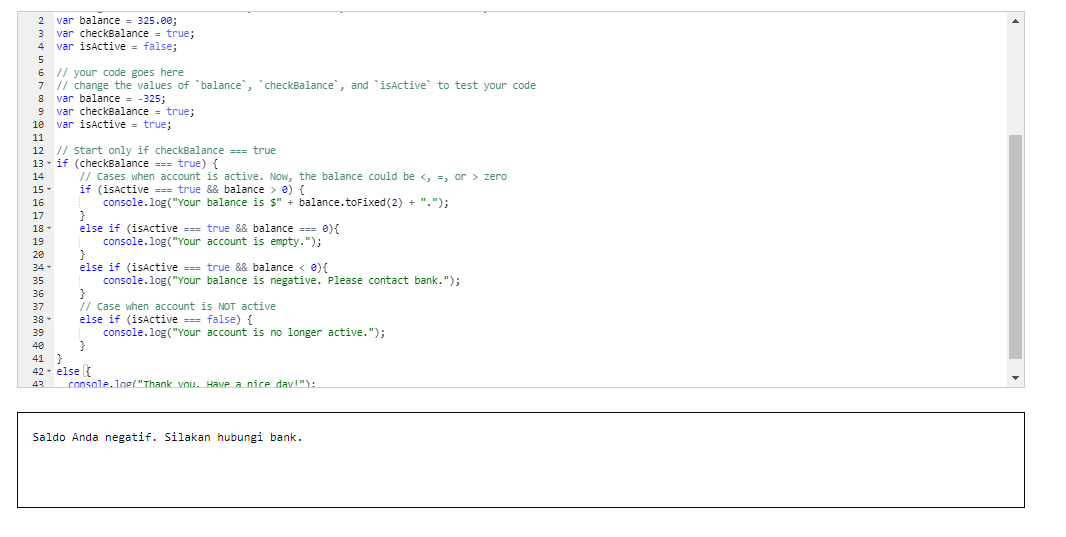
1. Logical AND and OR

Logika AND (&&) dan logika OR (||) adalah operator logika pada bahasa pemrograman yang digunakan untuk mengkombinasikan kondisi dalam suatu ekspresi boolean.

Operator logika AND (&&) digunakan untuk memeriksa apakah semua operand dalam ekspresi boolean bernilai true. Jika salah satu atau semua operand bernilai false, maka ekspresi akan menghasilkan nilai false. Namun jika semua operand bernilai true, maka ekspresi akan menghasilkan nilai true. Contohnya, ekspresi 5 > 3 && 2 < 4 akan menghasilkan nilai true karena kedua operand, yaitu 5 > 3 dan 2 < 4, bernilai true.

Operator logika OR (||) digunakan untuk memeriksa apakah setidaknya satu operand dalam ekspresi boolean bernilai true. Jika semua operand bernilai false, maka ekspresi akan menghasilkan nilai false. Namun jika setidaknya satu operand bernilai true, maka ekspresi akan menghasilkan nilai true. Contohnya, ekspresi 5 < 3 || 2 > 4 akan menghasilkan nilai false karena kedua operand, yaitu 5 < 3 dan 2 > 4, bernilai false.

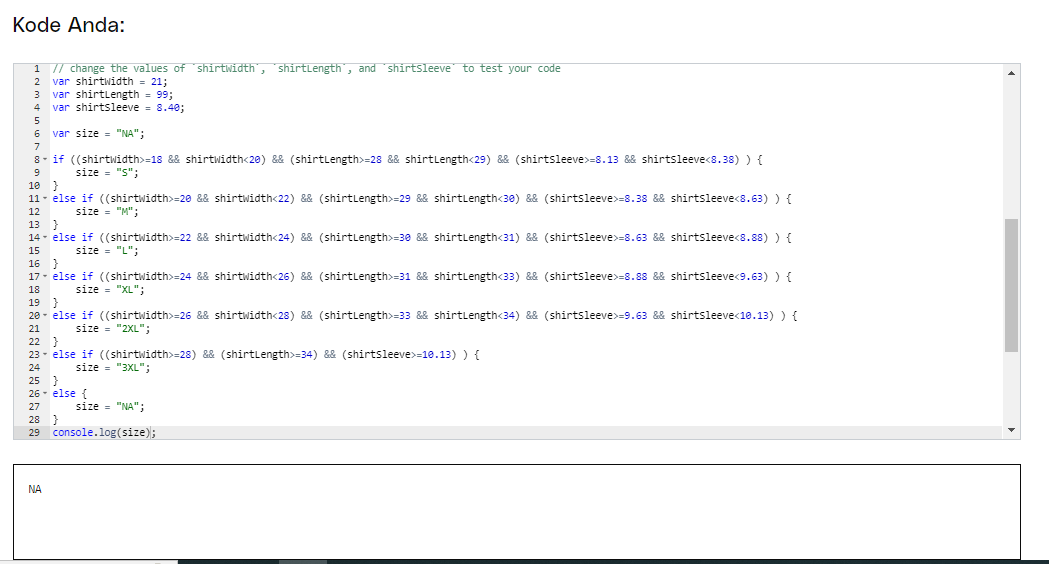
1. Quiz: Checking your Balance (3-5)



1. Quiz: Ice Cream (3-6)



1. Quiz: What do I Wear? (3-7)

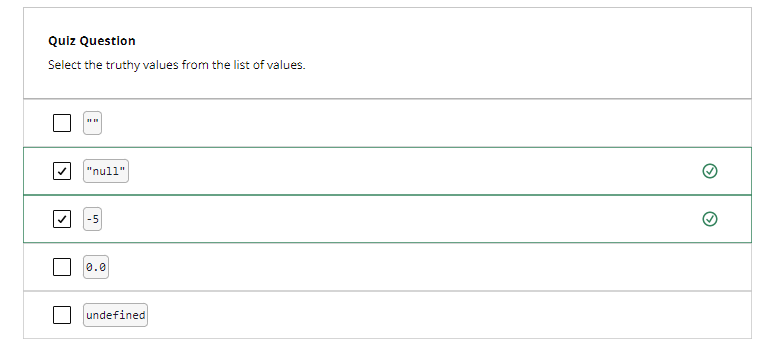


1. Advanced Conditionals

Kondisi lanjutan pada bahasa pemrograman digunakan untuk mengeksekusi suatu blok kode jika suatu kondisi tertentu terpenuhi, dan mengeksekusi blok kode yang berbeda jika kondisi tersebut tidak terpenuhi.

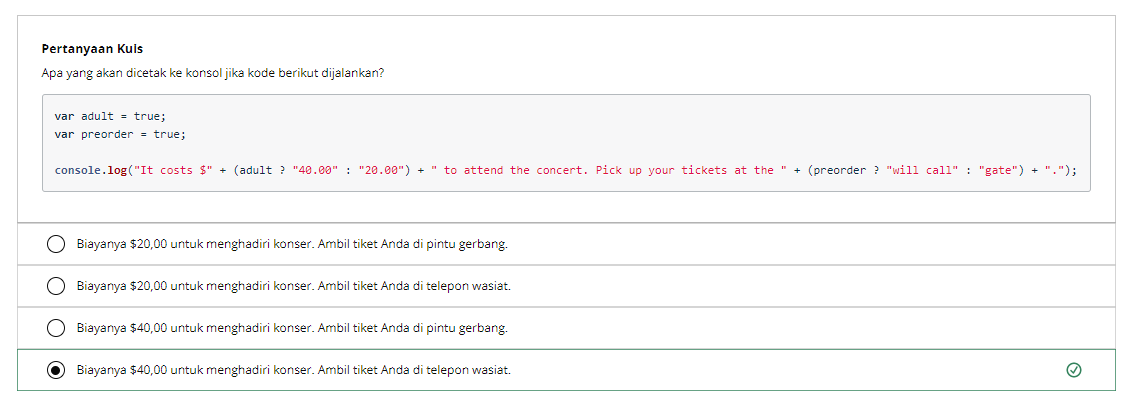
1. Truthy and Falsy

Truthy dan Falsy adalah istilah dalam JavaScript untuk menggambarkan nilai-nilai yang dianggap sebagai true atau false dalam konteks boolean.

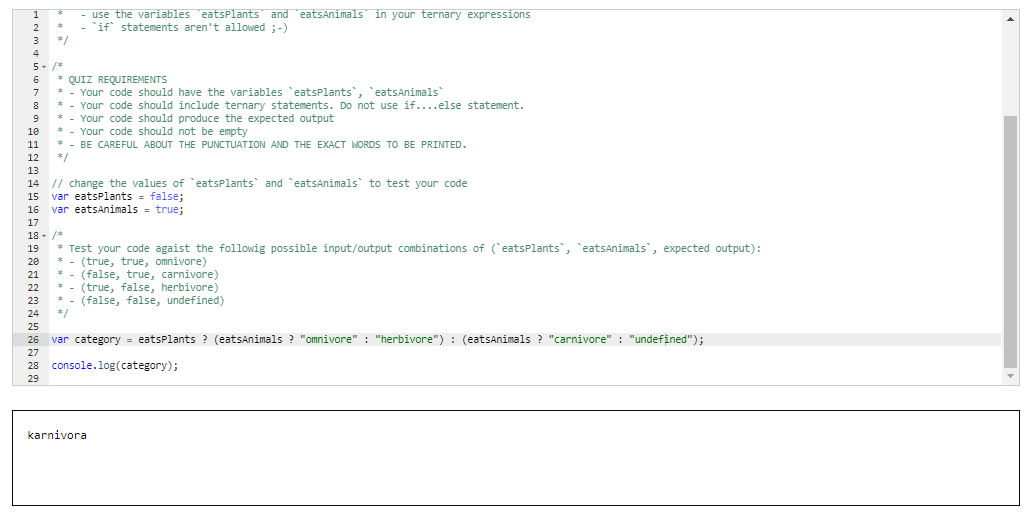


1. Ternary Operator

Untuk menggunakan operator ternary, pertama berikan pernyataan kondisional di sisi kiri ?. Kemudian, di antara ?dan :tulis kode yang akan dijalankan jika kondisinya adalah truedan di sisi kanan kode tuliskan :kode yang akan dijalankan jika kondisinya false.

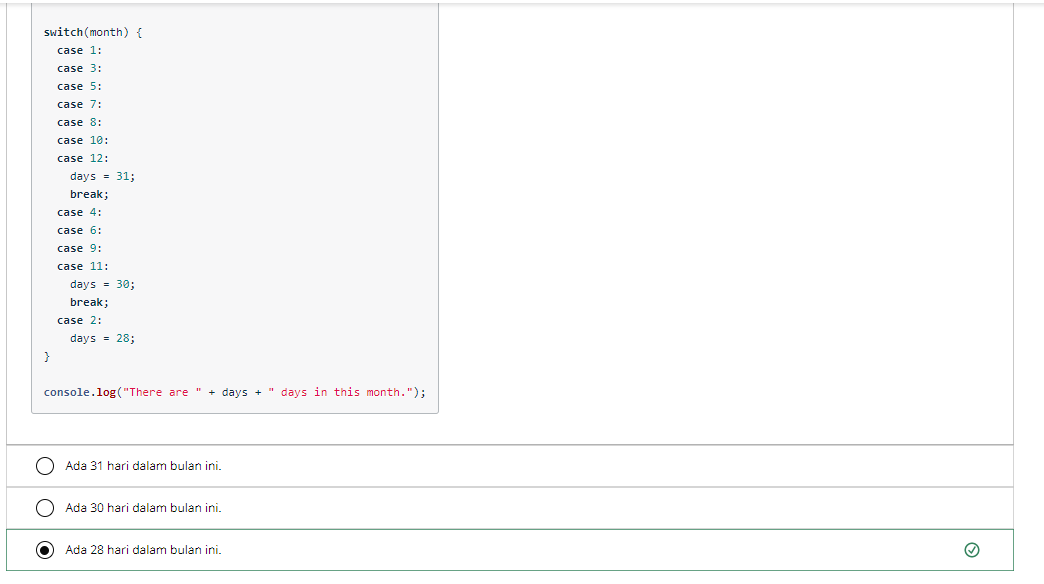


1. Quiz: Navigating the Food Chain (3-8)



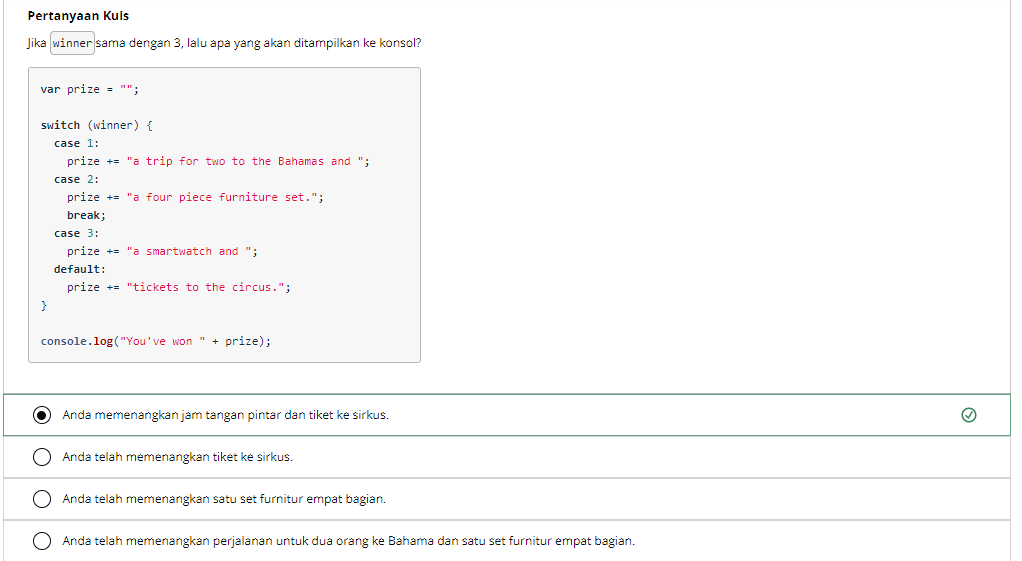
1. Switch Statement

Switch Statement adalah salah satu struktur kontrol pada bahasa pemrograman JavaScript yang digunakan untuk melakukan seleksi nilai atau ekspresi tertentu. Struktur kontrol ini akan mengevaluasi nilai atau ekspresi yang diberikan dan membandingkannya dengan beberapa nilai atau ekspresi lain yang sudah ditentukan sebelumnya.



1. Falling-through

Falling-through pada JavaScript terjadi ketika kasus (case) dalam blok switch statement tidak memiliki instruksi break atau return, sehingga kontrol aliran program akan terus berlanjut ke kasus selanjutnya tanpa menghentikan eksekusi



1. Lesson 3 Summary

Break Down Problems

Condirional Statements And Logical Operators

Advanced Techniques

* Truthy Falsy
* Ternary Operator
* Switch Statements

1. Intro to Loops

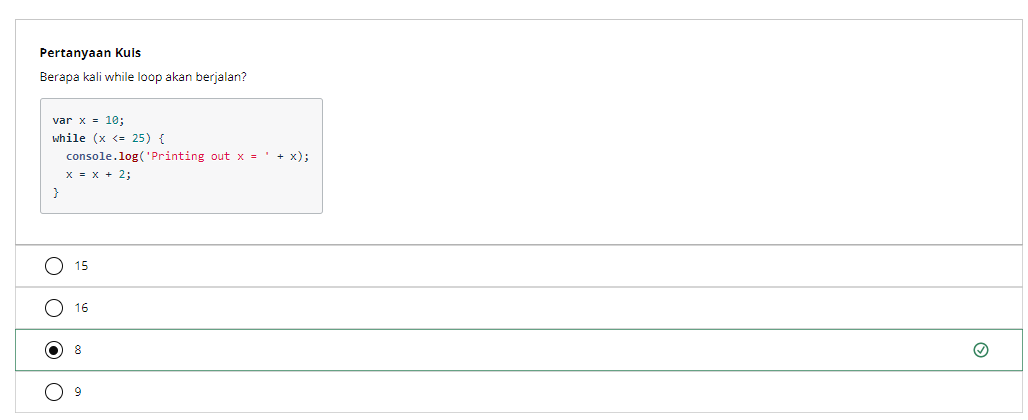
Loop pada JavaScript adalah struktur kontrol yang digunakan untuk mengulang eksekusi blok kode tertentu berulang kali selama kondisi tertentu masih terpenuhi.

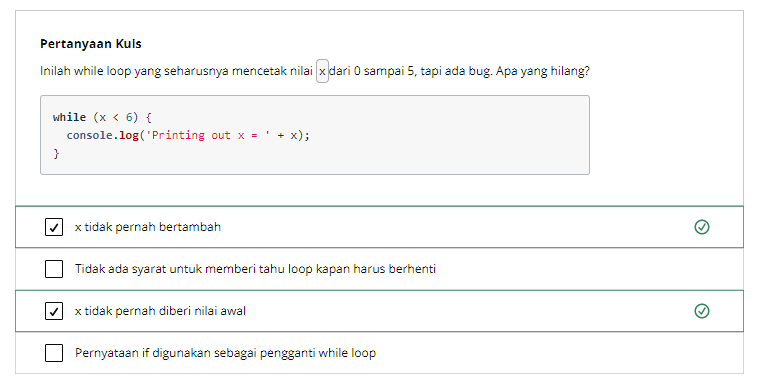
1. While Loops

While Loops pada JavaScript adalah jenis loop yang digunakan untuk mengulang eksekusi blok kode tertentu selama kondisi tertentu masih terpenuhi. Loop ini akan terus berjalan selama kondisi yang diberikan pada bagian while masih bernilai true.

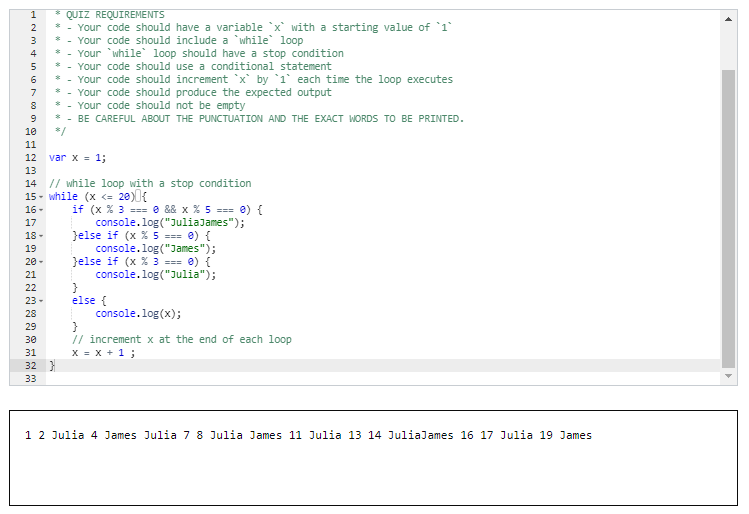
1. Parts of a While Loop

Tiga informasi utama yang harus dimiliki setiap loop adalah:

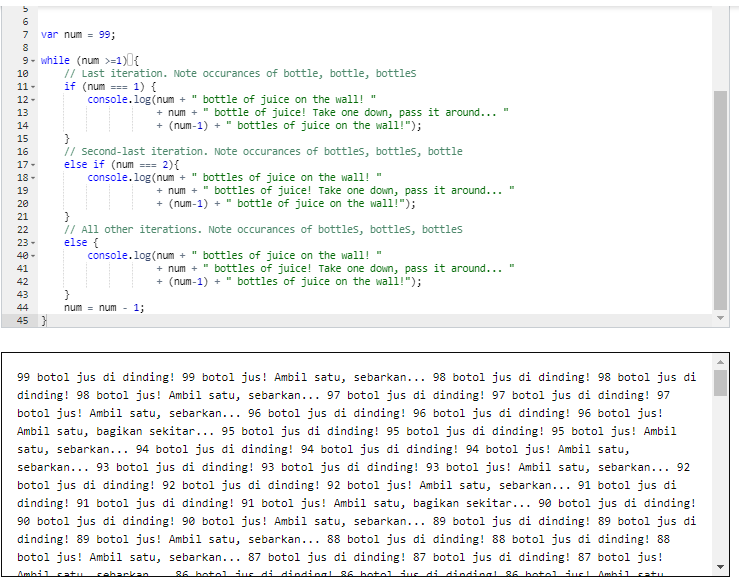
* Kapan memulai: Kode yang mengatur perulangan — misalnya menentukan nilai awal variabel.
* Kapan harus berhenti: Kondisi logis untuk menguji apakah perulangan harus dilanjutkan.
* Cara mendapatkan item berikutnya: Langkah penambahan atau pengurangan — misalnya, x = x \* 3ataux = x – 1



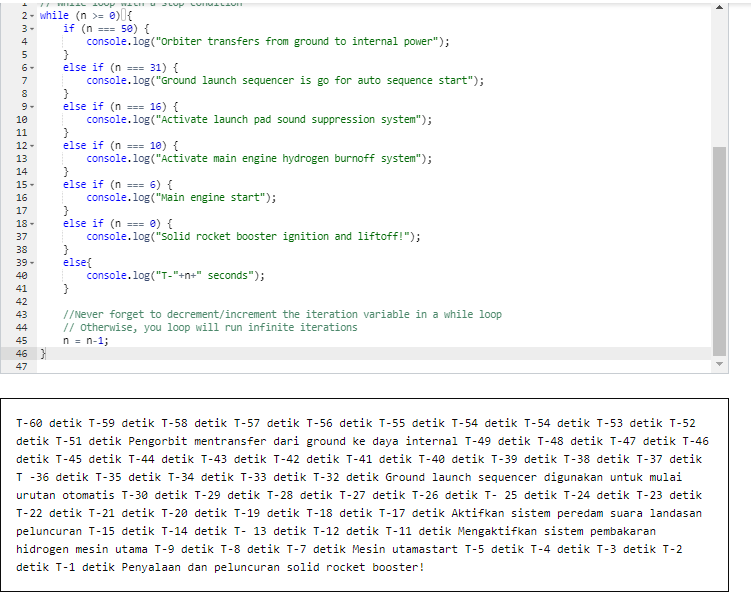
1. Quiz: JuliaJames (4-1)



1. Quiz: 99 Bottles of Juice (4-2)



1. Quiz: Countdown, Liftoff! (4-3)

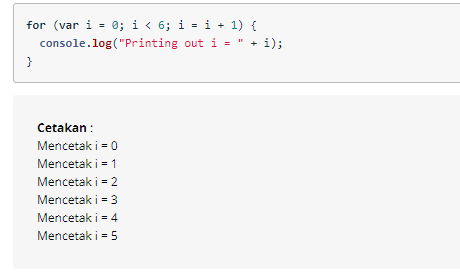


1. For Loops

For loop pada JavaScript digunakan untuk mengulang sebuah blok kode tertentu sejumlah kali. For loop terdiri dari tiga bagian utama yaitu inisialisasi variabel, kondisi looping, dan perubahan variabel setelah setiap iterasi.

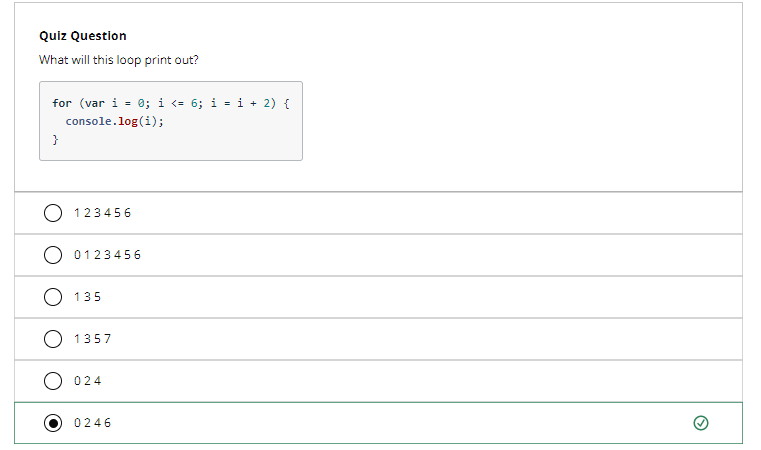
1. Parts of a For Loop

Berikut adalah contoh perulangan for yang mencetak nilai dari 0 hingga 5. Perhatikan titik koma yang memisahkan berbagai pernyataan perulangan for:var i = 0; i < 6; i = i + 1



1. Nested Loops

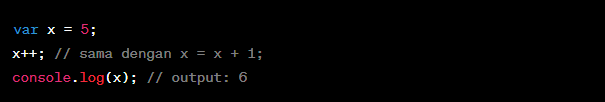
Nested loops pada JavaScript adalah sebuah konstruksi perulangan yang terdiri dari satu atau lebih loop di dalam loop lainnya. Dalam nested loops, setiap perulangan akan mengeksekusi seluruh iterasi yang ada di dalam perulangan lainnya.



1. Increment and Decrement

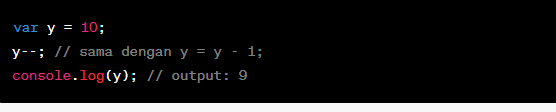
Increment dan Decrement adalah operator yang digunakan untuk menambah atau mengurangi nilai dari sebuah variabel dengan jumlah 1. Dalam JavaScript, terdapat dua jenis operator Increment dan Decrement, yaitu:

++ (Increment): operator ini digunakan untuk menambahkan nilai variabel dengan jumlah 1. Contohnya:

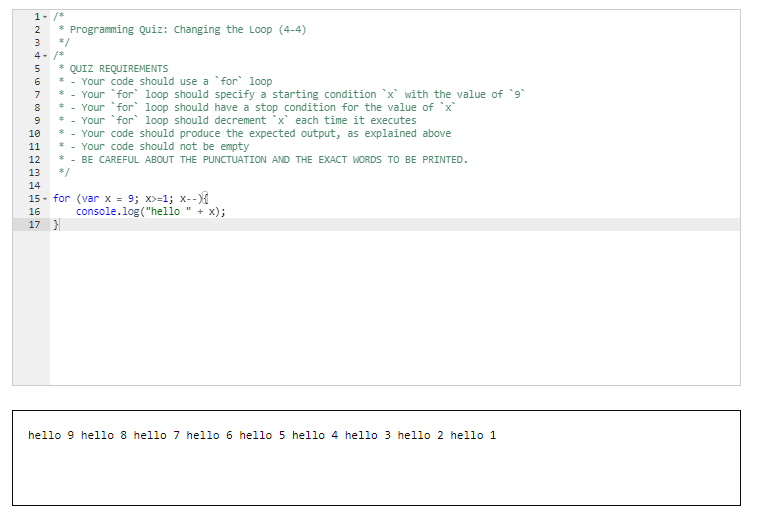


-- (Decrement): operator ini digunakan untuk mengurangi nilai variabel dengan jumlah 1.

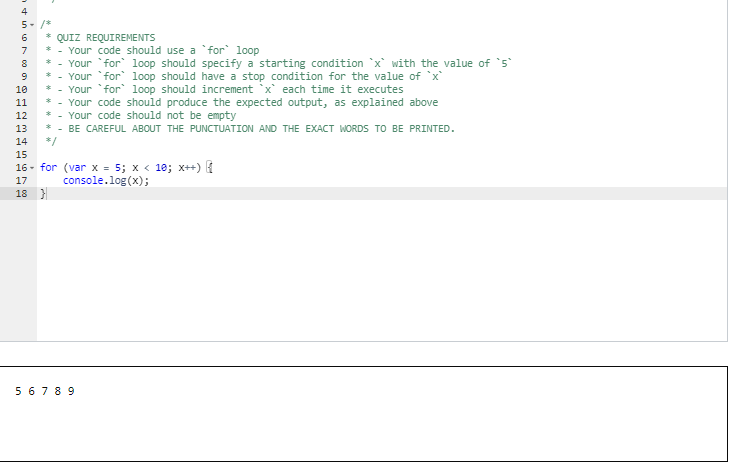
Contohnya:



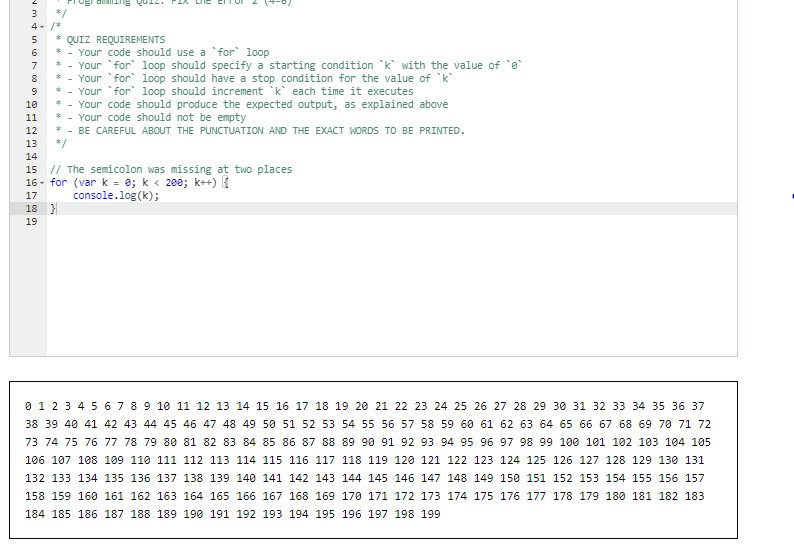
1. Quiz: Changing the Loop (4-4)



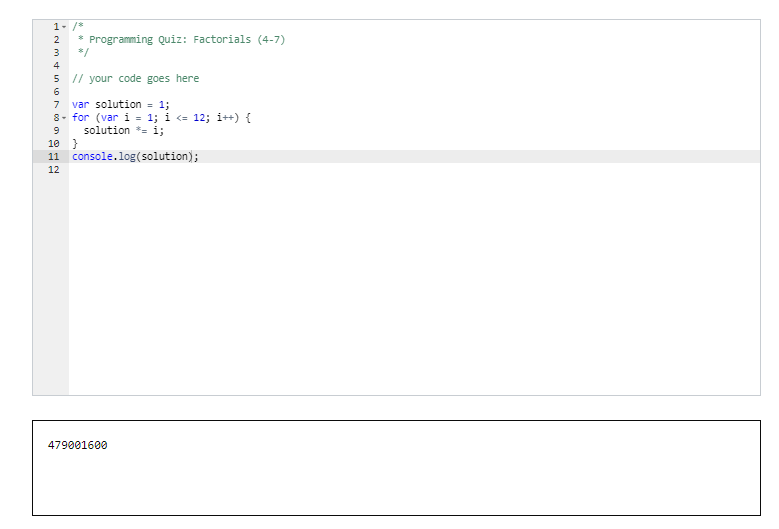
1. Quiz: Fix the Error 1 (4-5)



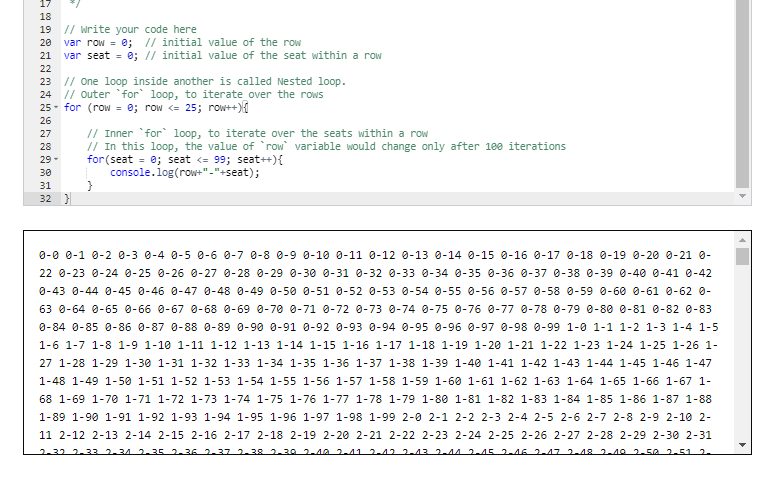
1. Quiz: Fix the Error 2 (4-6)



1. Quiz: Factorials! (4-7)



1. Quiz: Find my Seat (4-8)



1. Lesson 4 Summary

Loop adalah topik mendasar untuk Bahasa pemograman apa pun, pastikan anda merasa nyaman dengan loop karena anda akan menggunakan setiap saat. Loop adalah fitur yang sangat keren dari Bahasa tersebut, tetapi anda harus tahu apa yang mungkin menjadi fungsi yang lebih baik , fungsion sangat keren karena membiarkan anda mengelompokan baris kode bersama sama dan menjalankan semua.

1. Intro to Functions

Function adalah blok kode yang dapat digunakan kembali dan dijalankan ketika dibutuhkan. Dalam JavaScript, function adalah objek yang bisa dianggap sebagai "paket" yang terdiri dari beberapa pernyataan yang akan dieksekusi bersama-sama. Function biasanya digunakan untuk menyelesaikan suatu tugas tertentu, dan dapat menerima input (parameter) dan mengembalikan output.

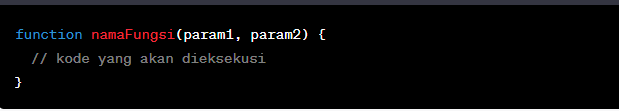
1. Function Example



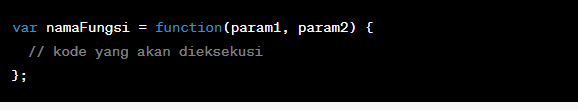
1. Declaring Functions

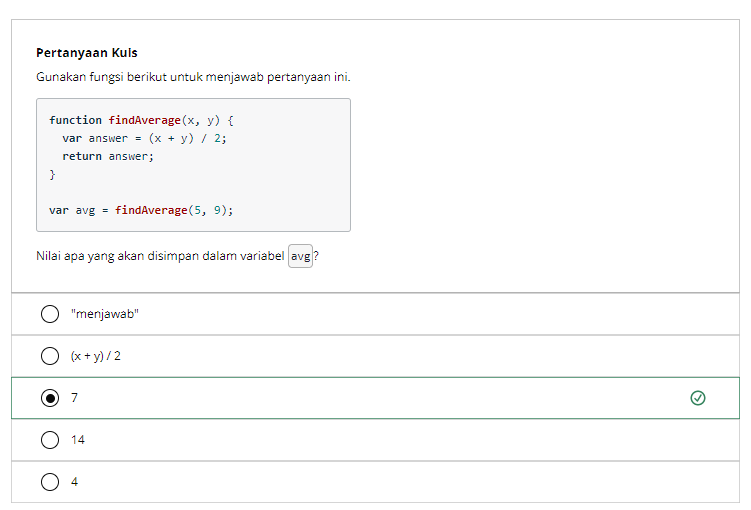
Anda dapat mendeklarasikan fungsi di JavaScript menggunakan function statement atau function expression.

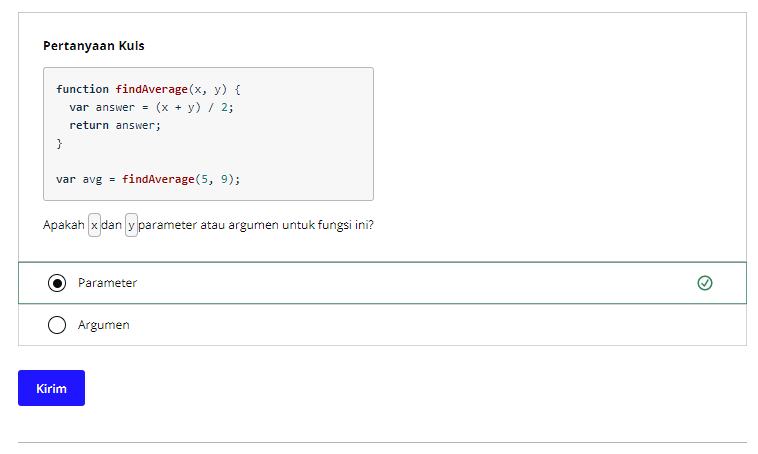
* Function statement dideklarasikan dengan menggunakan kata kunci function, diikuti dengan nama fungsi, daftar parameter dalam tanda kurung, dan kode yang akan dieksekusi di dalam kurung kurawal. Contoh deklarasi fungsi menggunakan function statement:



* Function expression adalah cara lain untuk mendeklarasikan fungsi di JavaScript. Dalam hal ini, fungsi dideklarasikan sebagai nilai variabel. Contoh deklarasi fungsi menggunakan function expression:



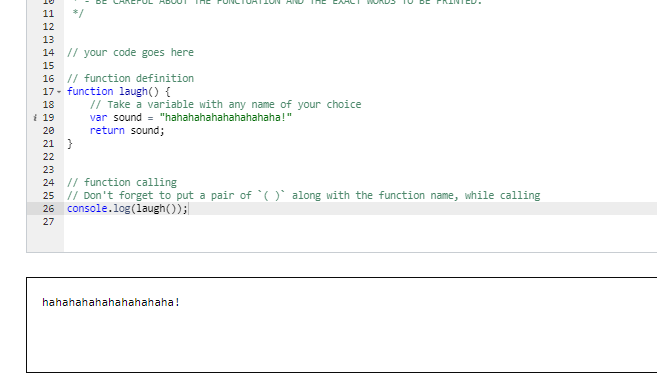




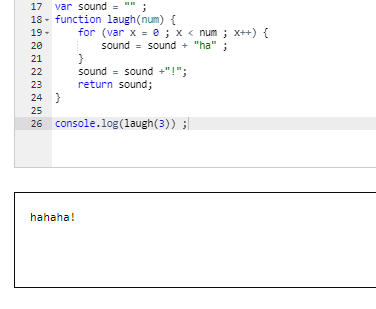
1. Function Recap

Fungsi mengemas kode sehingga Anda dapat dengan mudah menggunakan (dan menggunakan kembali) blok kode. Parameter adalah variabel yang digunakan untuk menyimpan data yang diteruskan ke fungsi agar fungsi tersebut dapat digunakan. Argumen adalah data aktual yang diteruskan ke suatu fungsi saat dipanggi

1. Quiz: Laugh it Off 1 (5-1)

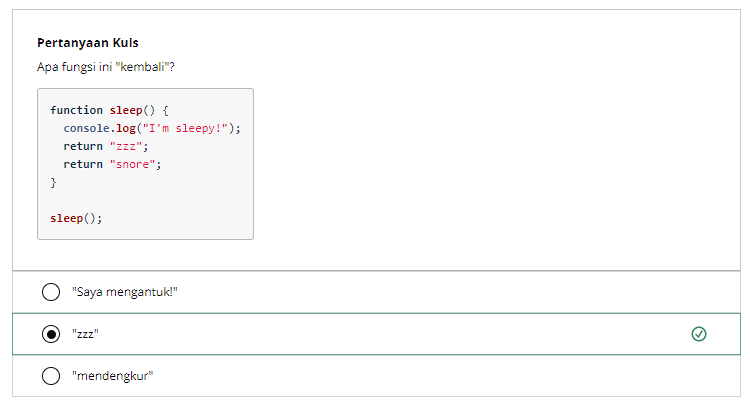


1. Quiz: Laugh it Off 2 (5-2)

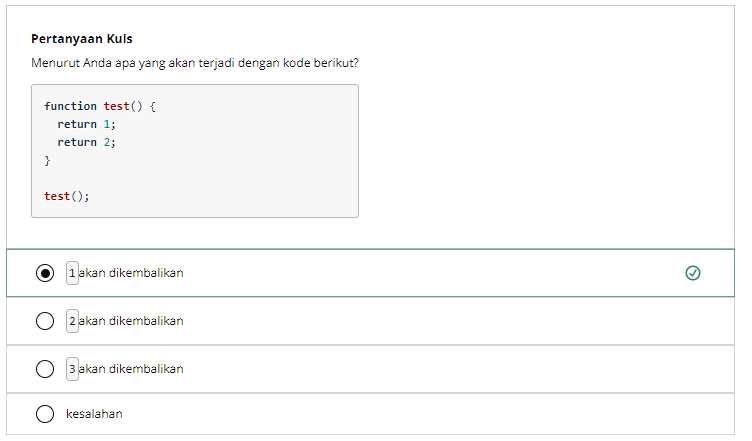


1. Return Values

Dalam JavaScript, sebuah fungsi dapat mengembalikan nilai dengan menggunakan kata kunci return. Ketika sebuah fungsi dipanggil, kode di dalam fungsi akan dijalankan dan menghasilkan nilai yang akan dikembalikan ke tempat pemanggilan fungsi.







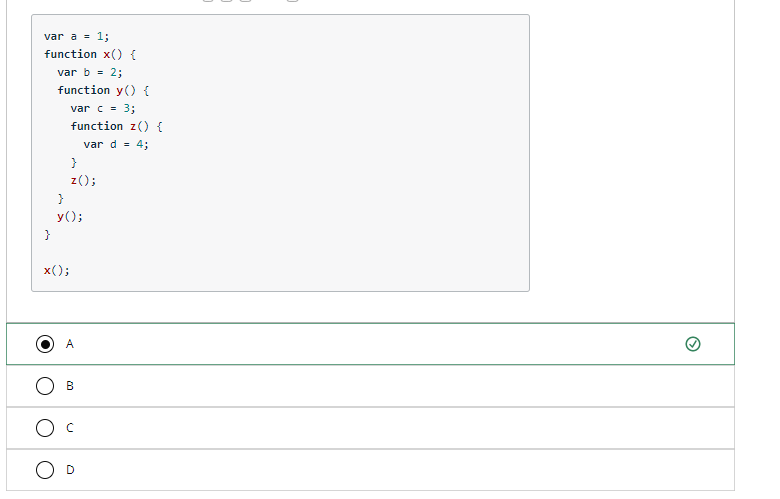
1. Using Return Value

Setelah sebuah fungsi mengembalikan nilai, nilai tersebut dapat digunakan oleh kode lain di program. Misalnya, nilai yang dikembalikan oleh fungsi dapat disimpan dalam variabel atau digunakan sebagai argumen untuk fungsi lain.



1. Coverage

Coverage atau istilah lengkapnya adalah code coverage adalah metrik yang digunakan untuk mengukur sejauh mana kode dalam suatu program telah dieksekusi selama proses pengujian (testing). Code coverage dapat digunakan untuk mengevaluasi kualitas pengujian suatu program, karena semakin banyak kode yang dieksekusi selama proses pengujian, semakin besar kemungkinan terdeteksinya bug atau kesalahan dalam program.





1. Imagery

Imagery dalam JavaScript merujuk pada kemampuan kita untuk membayangkan atau memvisualisasikan proses yang terjadi saat kode JavaScript dijalankan di browser atau lingkungan runtime JavaScript lainnya. Imagery membantu kita memahami bagaimana program berinteraksi dengan elemen HTML, mengambil data dari formulir, memanipulasi DOM, atau melakukan interaksi asinkron seperti mengambil data dari server atau menangani peristiwa.





1. Variabel Global

Variabel global dalam JavaScript adalah variabel yang dideklarasikan di luar fungsi atau blok kode apapun sehingga dapat diakses dari mana saja dalam kode, baik itu di dalam atau di luar fungsi. Variabel global dikenal sebagai variabel yang bersifat global karena dapat diakses dari lingkup yang lebih luas dari pada variabel lokal.

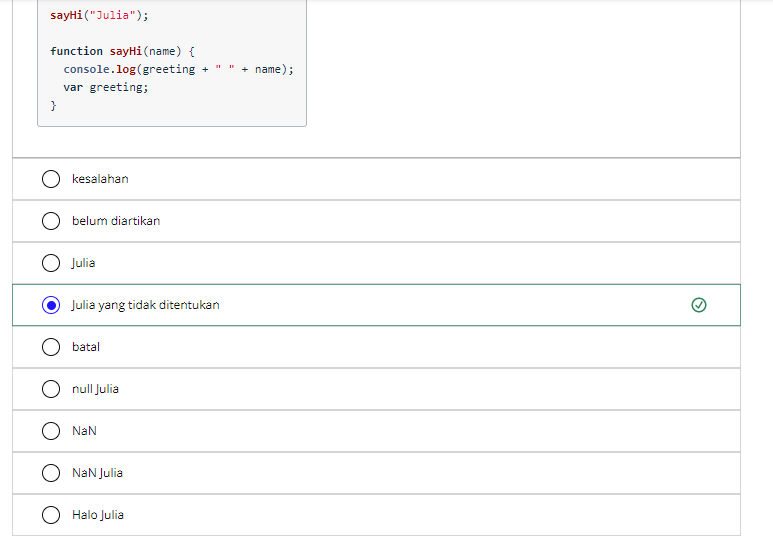
1. Scope Recap

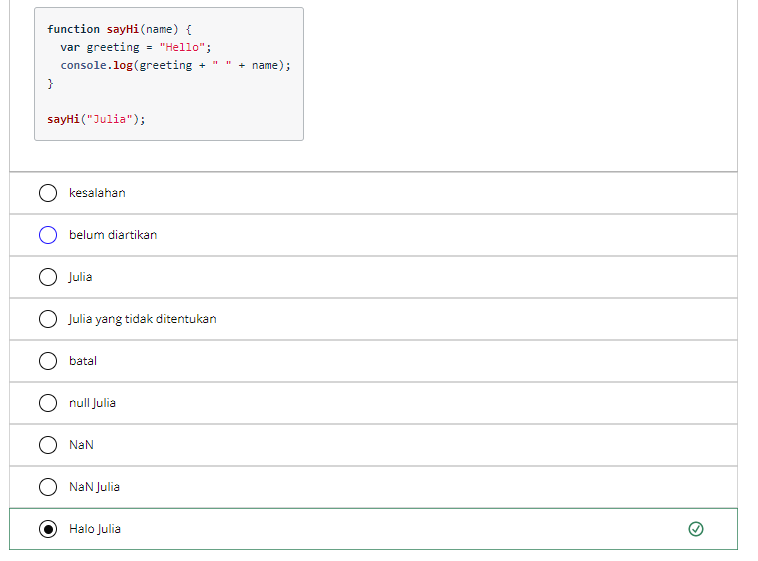
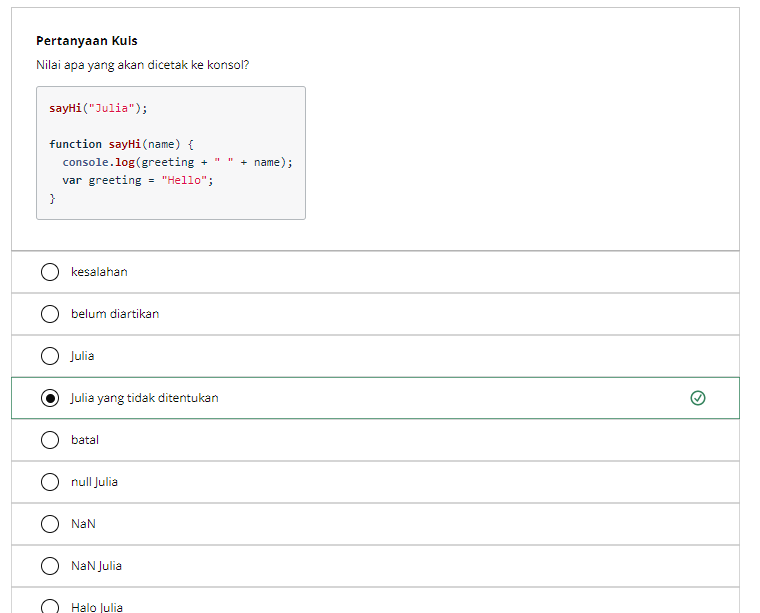
Apa yang telah Anda pelajari sejauh ini:

* Jika pengidentifikasi dideklarasikan dalam lingkup global , ini tersedia di mana saja .
* Jika pengidentifikasi dideklarasikan dalam function scope , itu tersedia dalam fungsi yang dideklarasikan (bahkan dalam fungsi yang dideklarasikan di dalam fungsi).
* Saat mencoba mengakses pengidentifikasi, JavaScript Engine pertama-tama akan melihat fungsi saat ini. Jika tidak menemukan apa pun, ia akan melanjutkan ke fungsi luar berikutnya untuk melihat apakah ia dapat menemukan pengidentifikasi di sana. Ini akan terus dilakukan hingga mencapai lingkup global.
* Pengidentifikasi global adalah ide yang buruk. Mereka dapat menyebabkan nama variabel yang buruk, nama variabel yang bertentangan, dan kode yang beran

1. Lifting

Terkadang kode JavaScript Anda akan menghasilkan kesalahan yang mungkin tampak berlawanan dengan intuisi pada awalnya. Mengangkat adalah salah satu topik yang mungkin menjadi penyebab beberapa kesalahan rumit yang Anda debug.



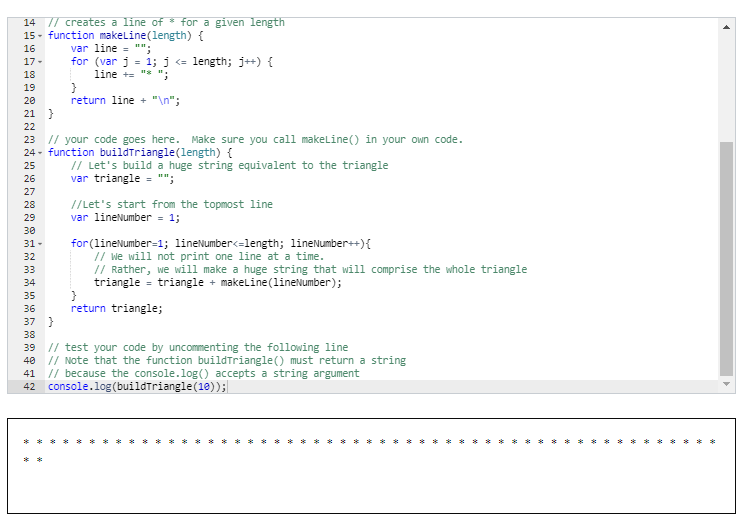


1. Lifting Recap

Apa yang telah Anda pelajari sejauh ini:

* JavaScript mengerek deklarasi fungsi dan deklarasi variabel ke bagian atas cakupan saat ini.
* Penugasan variabel tidak diangkat.
* Deklarasikan fungsi dan variabel di bagian atas skrip Anda, sehingga sintaks dan perilakunya konsisten satu sama lain.

1. Quiz: Build a Triangle (5-3)



1. Function Expressions

Function expression adalah cara untuk mendefinisikan fungsi dalam bentuk ekspresi yang dievaluasi saat kode dieksekusi. Dalam function expression, fungsi dianggap sebagai nilai dan dapat disimpan dalam variabel atau properti objek. Berbeda dengan function declaration, di mana fungsi didefinisikan dengan kata kunci function di awal kode.

1. Patterns with Function Expressions

"Patterns with Function Expressions" merujuk pada penggunaan fungsi sebagai ekspresi dalam pola-pola pemrograman. Hal ini dapat digunakan dalam berbagai konteks, seperti:

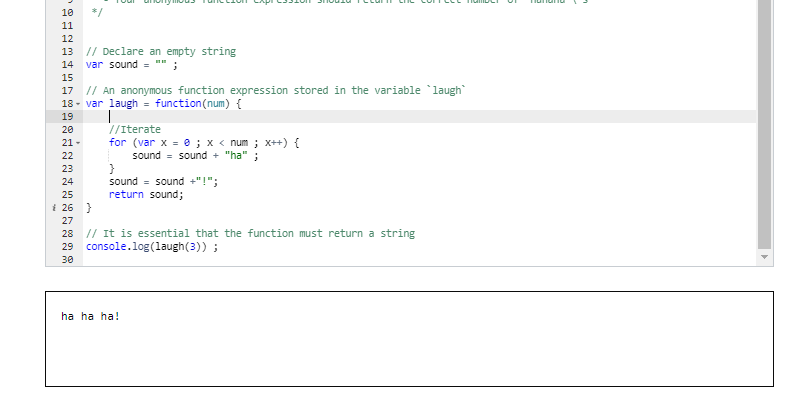
* Higher-Order Functions: Dalam pemrograman fungsional, fungsi dapat diperlakukan seperti tipe data lainnya. Dalam hal ini, fungsi dapat digunakan sebagai argumen untuk fungsi lain, atau dapat dikembalikan sebagai nilai dari fungsi itu sendiri.
* Callback Functions: Fungsi dapat digunakan sebagai callback function dalam JavaScript, yaitu fungsi yang dijalankan setelah suatu peristiwa terjadi. Contohnya, fungsi yang dijalankan ketika tombol di klik pada sebuah halaman web.
* Anonymous Functions: Fungsi juga dapat ditulis sebagai fungsi anonim atau tanpa nama, yang tidak perlu diberi nama dan dapat langsung digunakan sebagai argumen untuk fungsi lain atau ekspresi yang membutuhkannya.
* Arrow Functions: Fungsi panjang dapat ditulis secara lebih singkat menggunakan arrow function, yaitu fungsi yang ditulis dengan menggunakan tanda panah (=>) sebagai pengganti kata kunci function.

1. Function Expression Recap

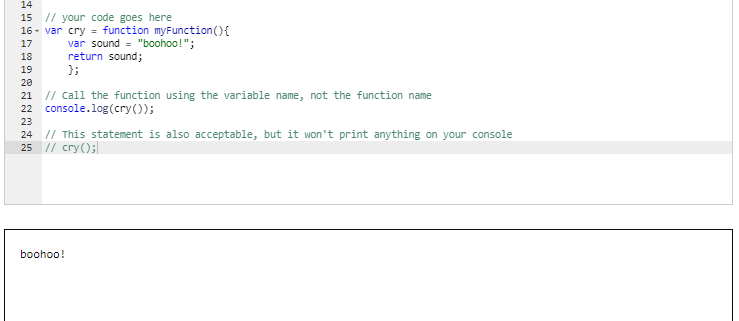
"Function Expression Recap" adalah ringkasan atau ulasan kembali tentang penggunaan fungsi sebagai ekspresi dalam pemrograman.

* Function Expression adalah cara untuk menetapkan fungsi ke dalam variable.
* Fungsi yang ditetapkan ke dalam variabel dengan Function Expression dapat digunakan di mana saja seperti variabel biasa.
* Function Expression dapat ditulis sebagai fungsi anonim atau dengan nama. Fungsi anonim tidak memiliki nama dan dapat digunakan langsung sebagai argumen untuk fungsi lain atau ekspresi yang membutuhkannya.
* Function Expression dapat digunakan dalam Higher-Order Functions dan sebagai callback function.
* Arrow Function adalah cara lain untuk menulis Function Expression secara lebih singkat dan mudah dibaca.
* Dalam JavaScript, hoisting (mengangkat) tidak berlaku untuk Function Expression, sehingga variabel yang menampung Function Expression harus dideklarasikan terlebih dahulu sebelum digunakan.

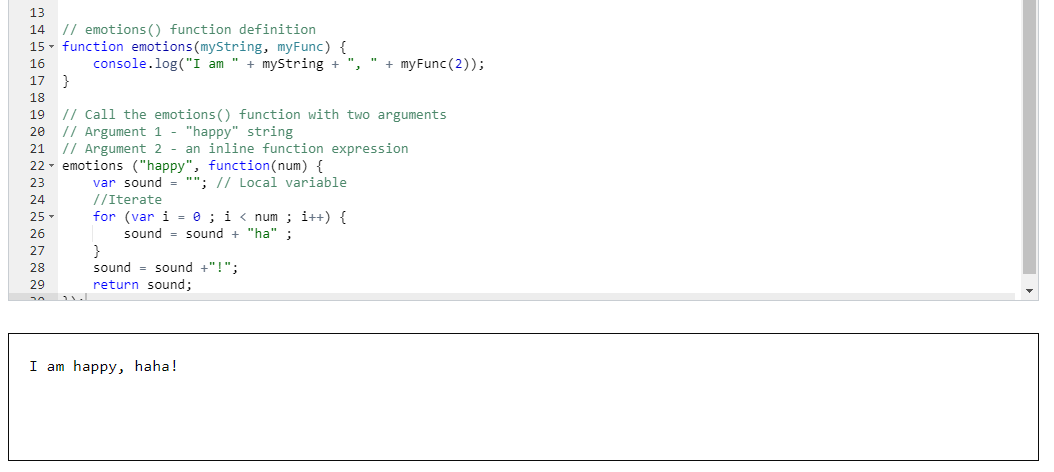
1. Quiz: Laugh (5-4)



1. Quiz: Cry (5-5)



1. Quiz: Inline (5-6)



1. Lesson 5 Summary

Setelah kita belajar Fungsin sekarang kita akan belajar array, lets’go.

1. Intro to Arrays

Array adalah salah satu struktur data yang penting dalam pemrograman. Array adalah kumpulan nilai atau objek yang disimpan dalam satu variabel dengan indeks yang teratur. Setiap nilai dalam array memiliki indeks numerik yang dimulai dari 0.

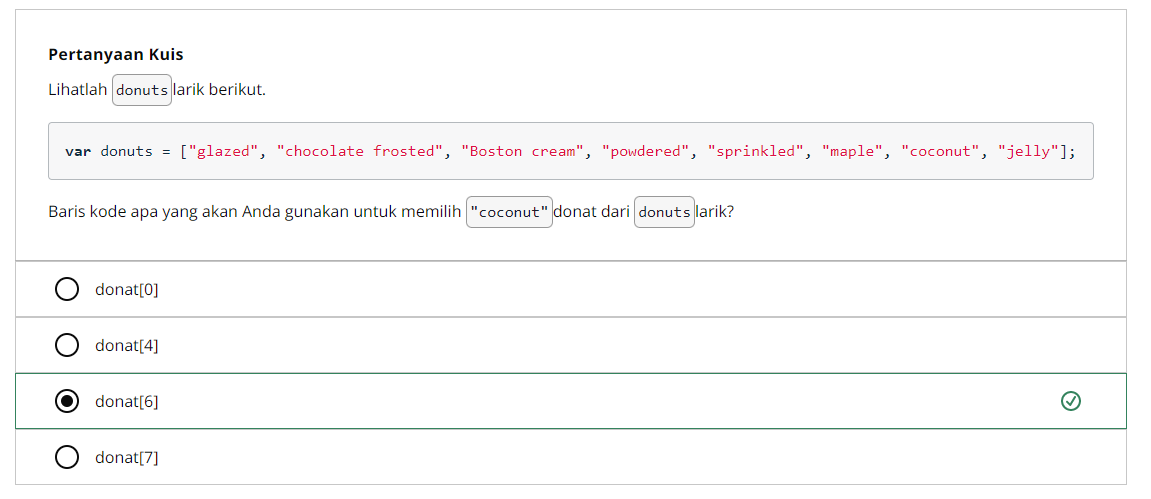
1. Membuat Array

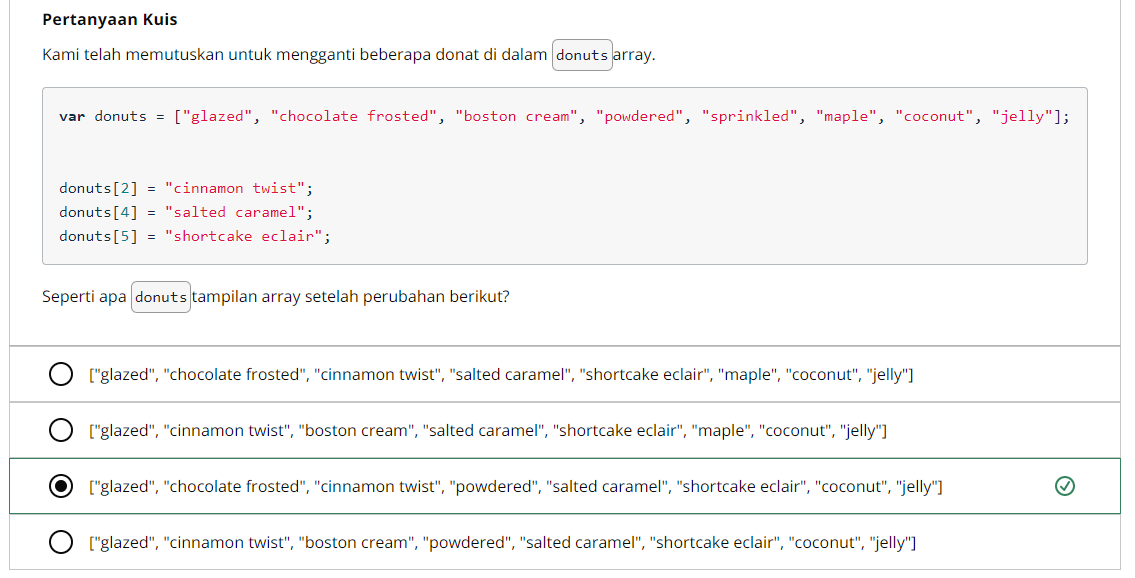
Array berguna karena menyimpan beberapa nilai ke dalam struktur data tunggal yang terorganisir . Anda dapat mendefinisikan larik baru dengan mencantumkan nilai yang dipisahkan dengan koma di antara tanda kurung siku [].



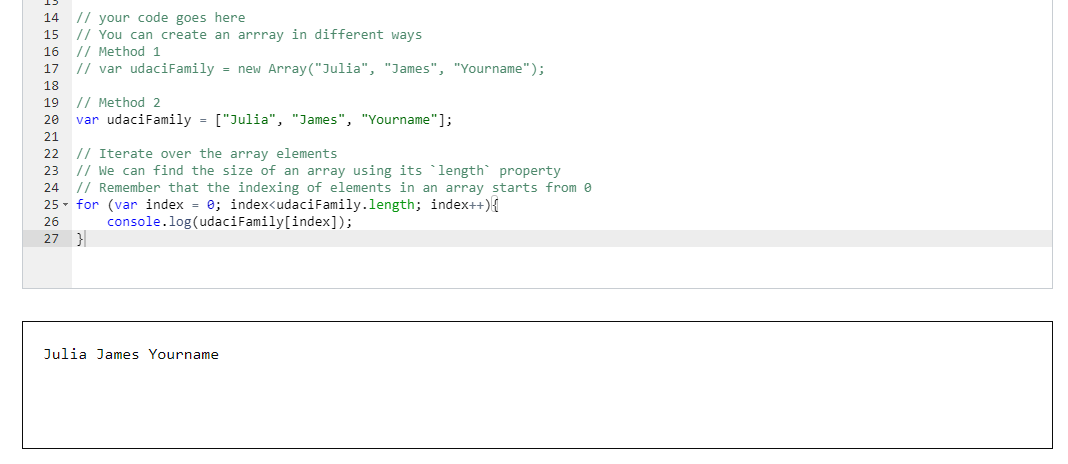
1. Array Index

Ingat bahwa elemen dalam array diindeks mulai dari posisi 0. Untuk mengakses elemen dalam array, gunakan nama array yang langsung diikuti dengan tanda kurung siku yang berisi indeks dari nilai yang ingin Anda akses.





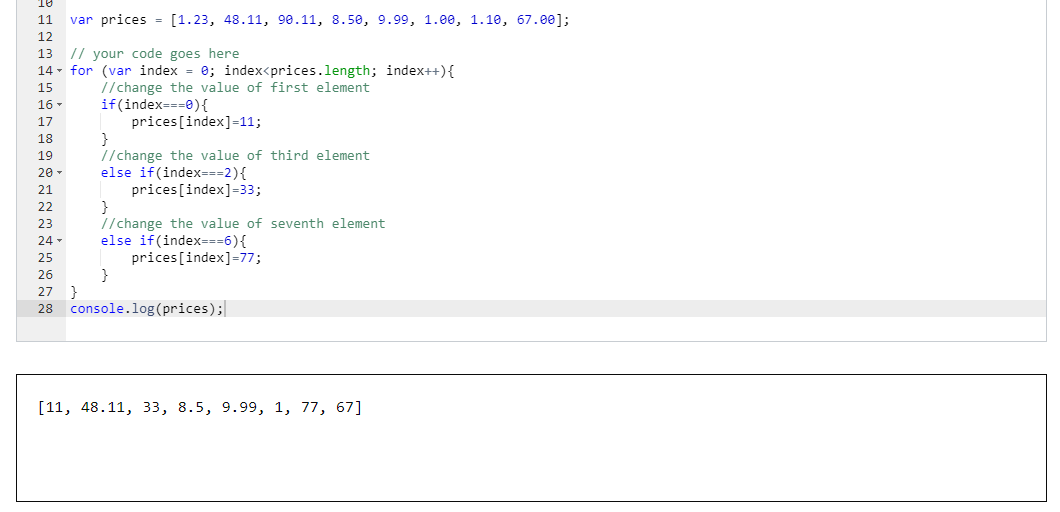
1. Quiz: UdaciFamily (6-1)



1. Quiz: Building the Crew (6-2)



1. Quiz: The Price is Right (6-3)



1. Array Properties and Methods

Array memiliki banyak properti dan metode yang dapat digunakan untuk memanipulasi dan mengelola data dalam array.

1. Length

Array.length adalah properti bawaan dari objek Array di JavaScript yang mengembalikan jumlah elemen dalam array. Properti ini dapat digunakan untuk mengakses dan mengetahui jumlah elemen dalam array, dan dapat digunakan dalam beberapa skenario pemrograman, seperti dalam iterasi melalui array menggunakan loop.



1. Push

push() adalah metode bawaan dari objek Array di JavaScript yang digunakan untuk menambahkan satu atau lebih elemen baru ke akhir array dan mengembalikan panjang array yang baru. Metode ini memodifikasi array yang memanggil metodenya dengan menambahkan elemen baru ke akhirnya.

1. Pop

pop() adalah metode bawaan dari objek Array di JavaScript yang digunakan untuk menghapus elemen terakhir dari array dan mengembalikan elemen yang dihapus. Metode ini memodifikasi array yang memanggil metodenya dengan menghapus elemen terakhir dari array tersebut.



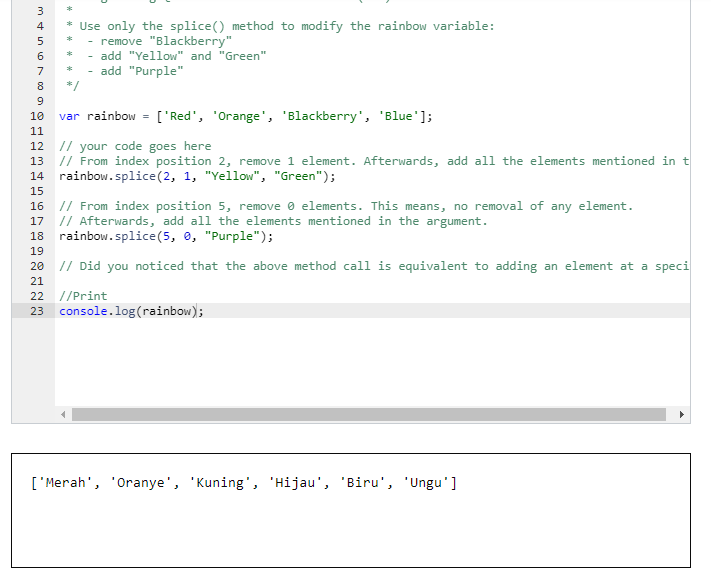
1. Splice

splice()adalah metode praktis lain yang memungkinkan Anda menambah dan menghapus elemen dari mana saja di dalam array.

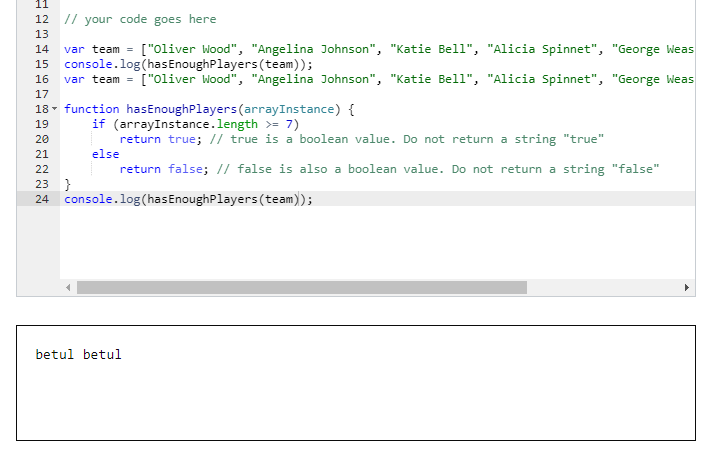
Sementara push()dan pop()membatasi Anda untuk menambah dan menghapus elemen dari akhir array , splice()memungkinkan Anda menentukan lokasi indeks untuk menambahkan elemen baru, serta jumlah elemen yang ingin Anda hapus (jika ada).



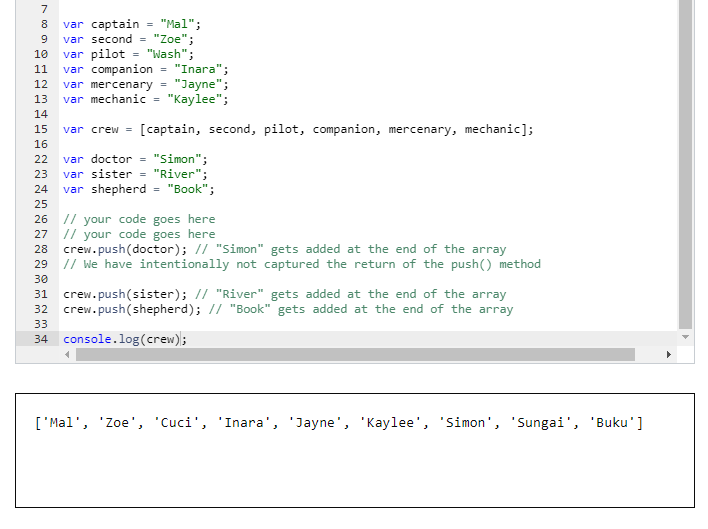
1. Quiz: Rainbow Colors (6-4)



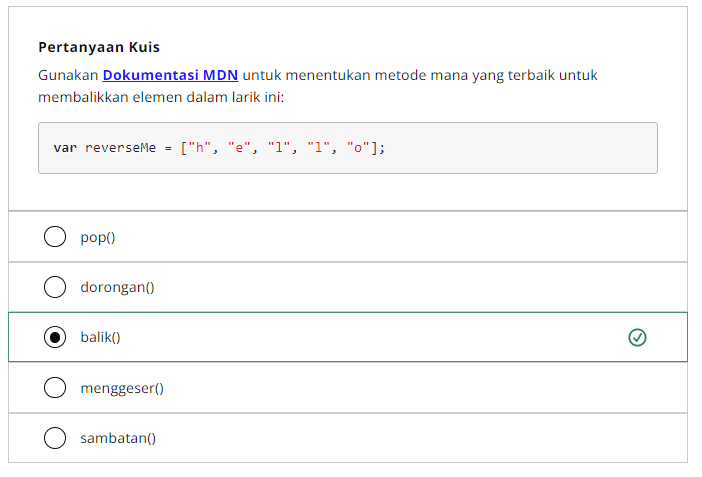
1. Kuis: Piala Quidditch (6-5)

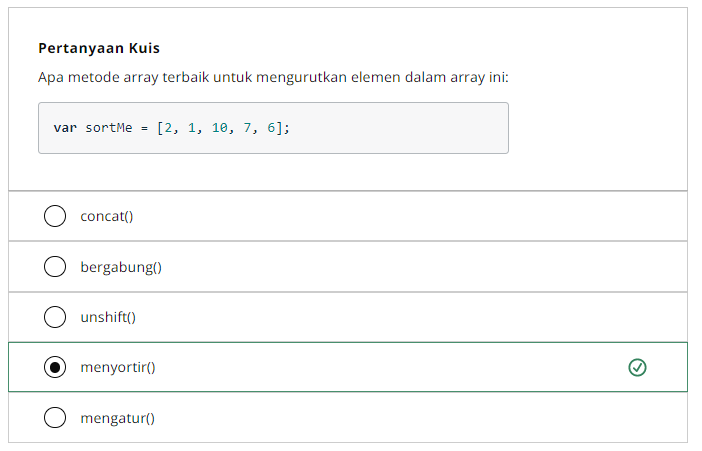


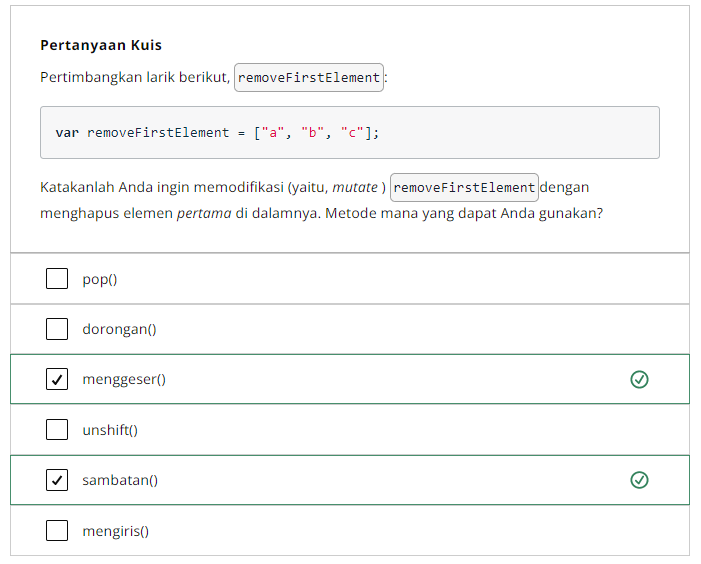
1. Quiz: Join the Crew (6-6)

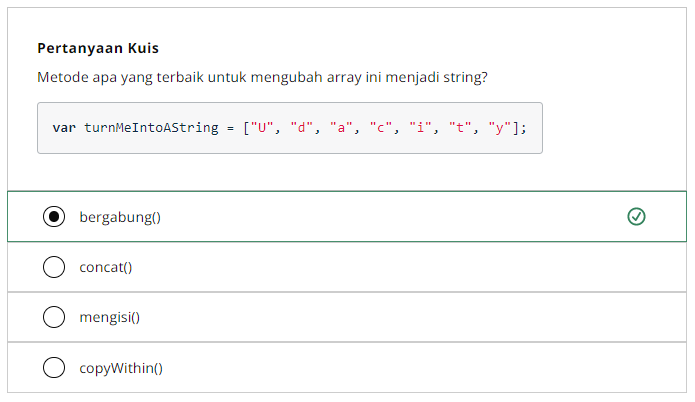


1. Quiz: Checking Documents (6-7)



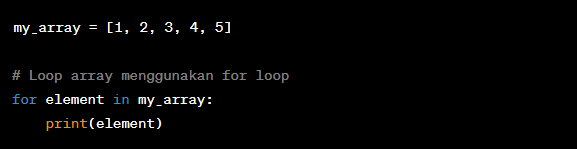






1. Loop Array

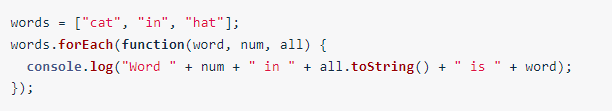
Loop Array adalah sebuah teknik dalam pemrograman komputer yang digunakan untuk mengakses setiap elemen dalam sebuah array secara berulang-ulang. Dalam loop array, perintah atau kode akan dieksekusi untuk setiap elemen dalam array, sehingga memungkinkan pengguna untuk melakukan operasi atau manipulasi data pada setiap elemen secara efisien.

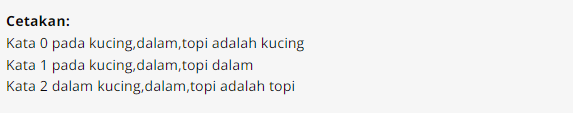


1. forEach Loop

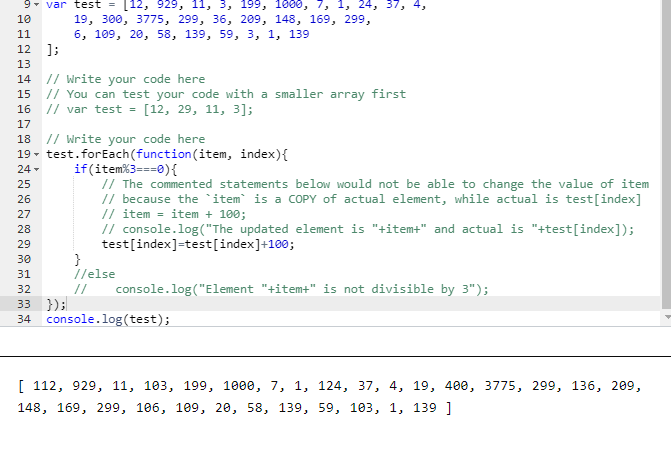
Fungsi yang Anda berikan ke forEach()metode dapat menggunakan hingga tiga parameter. Di dalam video, ini disebut element, index, dan array, tetapi Anda dapat menamainya sesuka Anda.

Metode ini forEach()akan memanggil fungsi ini satu kali untuk setiap elemen dalam array (maka nama forEach.) Setiap kali, ia akan memanggil fungsi dengan argumen yang berbeda. Parameter elementakan mendapatkan nilai dari elemen array. Parameter indexakan mendapatkan indeks elemen (dimulai dengan nol). Parameter arrayakan mendapatkan referensi ke seluruh larik, yang berguna jika Anda ingin memodifikasi elemen.





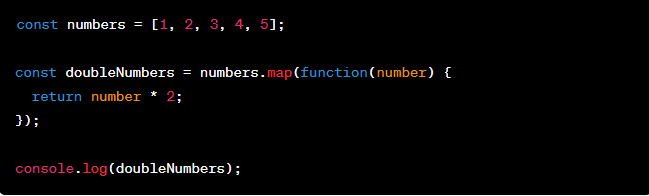
1. Quiz: Other Types of Rounds (6-8)



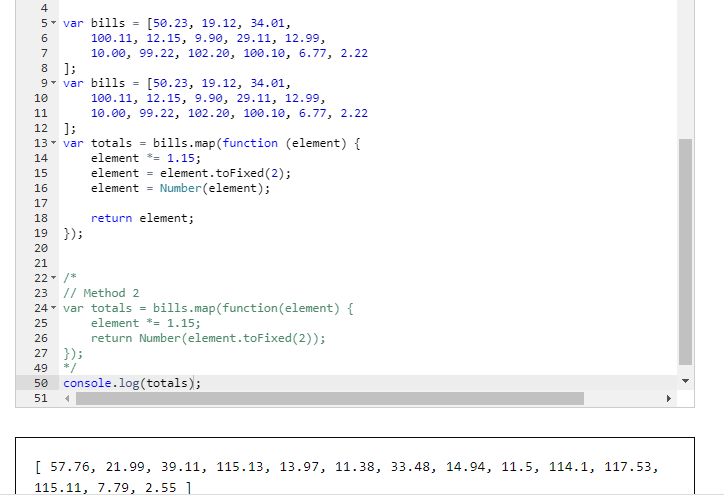
1. Map

map adalah sebuah metode bawaan dari tipe data Array pada bahasa pemrograman JavaScript. Metode ini digunakan untuk mengubah setiap elemen dalam array dengan mengaplikasikan sebuah fungsi pada setiap elemen, dan kemudian mengembalikan array baru dengan elemen-elemen yang telah diubah.

Contoh sederhana dari penggunaan map pada JavaScript adalah sebagai berikut:

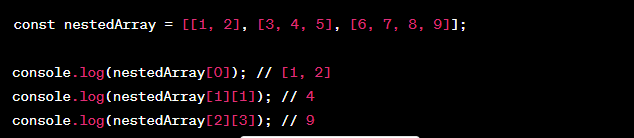


1. Quiz: I Got the Bill (6-9)



1. Array in Array

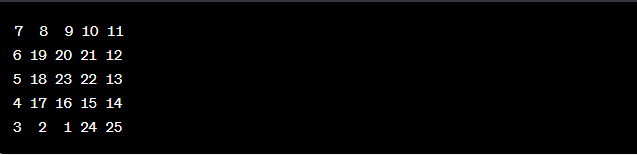
Array in Array atau Array dalam Array, juga dikenal sebagai nested array, adalah struktur data di mana sebuah array dijadikan sebagai elemen dalam array lainnya. Dalam nested array, setiap elemen dalam array induk adalah array yang berdiri sendiri dengan elemen-elemennya sendiri.



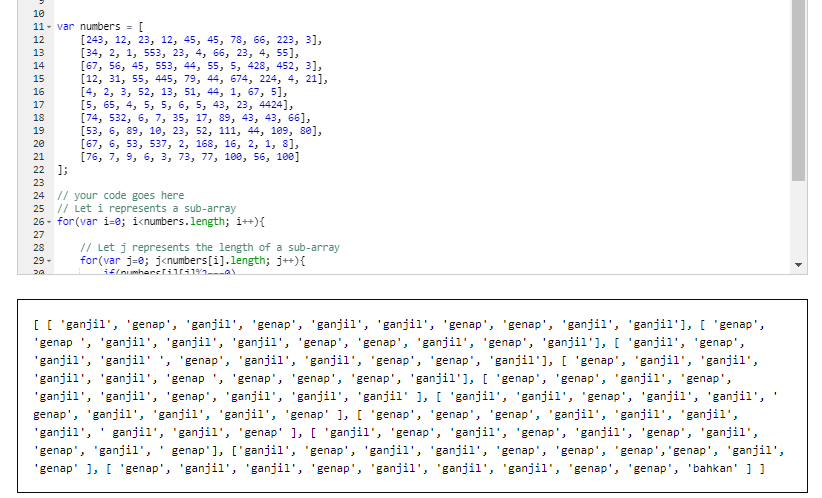
1. 2D Donut Arrangement

2D Donut Arrangement adalah susunan matriks dua dimensi yang memiliki bentuk seperti donat. Susunan ini terdiri dari angka-angka dari 1 sampai n^2 yang dibagi-bagi secara merata dan diposisikan dalam bentuk lingkaran, sehingga menghasilkan bentuk matriks yang berbentuk seperti donat.

Contoh dari 2D Donut Arrangement adalah sebagai berikut:



1. Quiz: Nested Numbers (6-10)



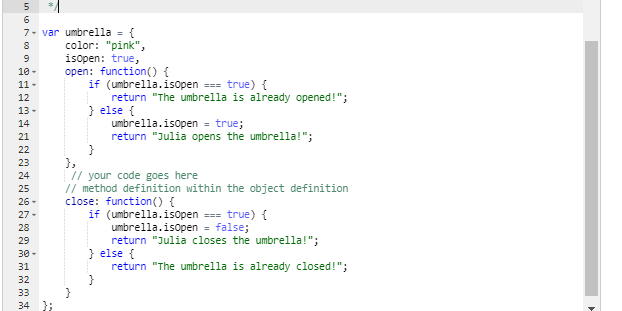
1. Objects in Code

Dalam pemrograman, object adalah sebuah tipe data kompleks yang terdiri dari kumpulan pasangan nama dan nilai, yang disebut sebagai properti (atau dalam bahasa pemrograman lainnya disebut sebagai fields atau attributes). Properti-properti ini dapat berupa nilai primitif seperti string, number, dan boolean, atau bahkan dapat berupa objek lain, termasuk fungsi (yang dalam hal ini sering disebut sebagai method).

Dalam JavaScript, objek dapat didefinisikan dengan menggunakan sintaksis objek literal. Contoh sederhana dari objek literal adalah sebagai berikut:



1. Quiz: Umbrella (7-1)



1. Objek

Objek pada JavaScript adalah kumpulan dari properti (atau key) dan nilai (atau value) yang terkait satu sama lain. Properti adalah nama atau identifier yang diberikan pada nilai dalam objek. Objek pada JavaScript dapat merepresentasikan benda nyata seperti orang, mobil, atau rumah, atau konsep abstrak seperti waktu, tanggal, dan warna.

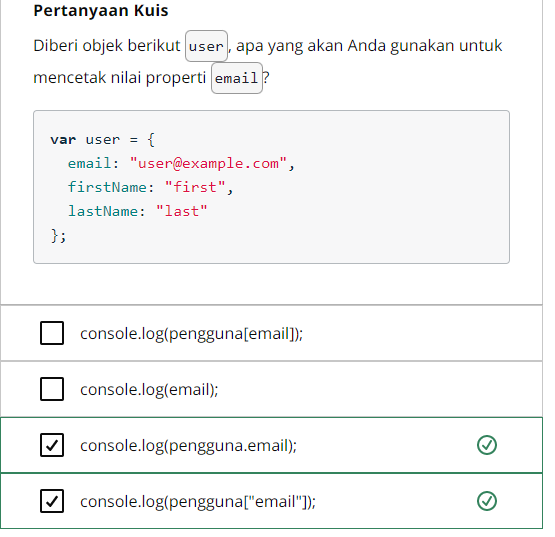


1. Object Literals

Object Literals adalah cara untuk membuat objek pada JavaScript dengan menggunakan sintaksis objek literal {}. Sintaksis ini memungkinkan kita untuk membuat objek secara langsung tanpa harus menggunakan constructor function.



1. Naming Convention

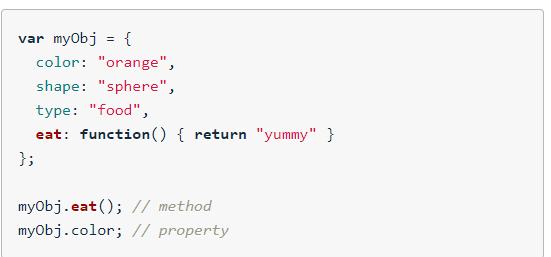




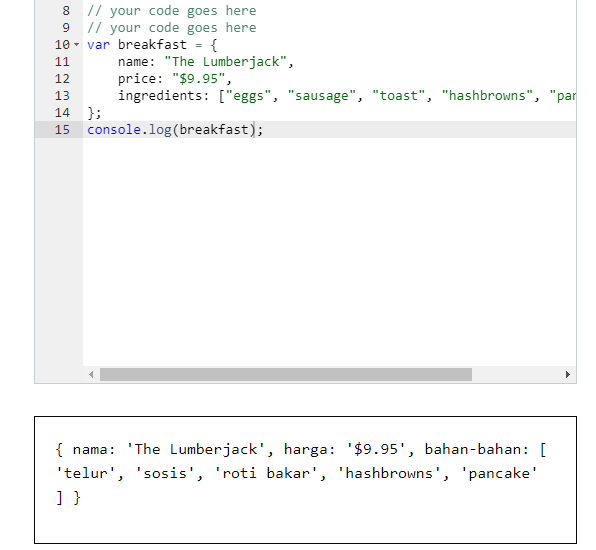
1. Object Summary

Objek adalah salah satu struktur data terpenting dalam JavaScript. Bersiaplah untuk melihat mereka di mana-mana!

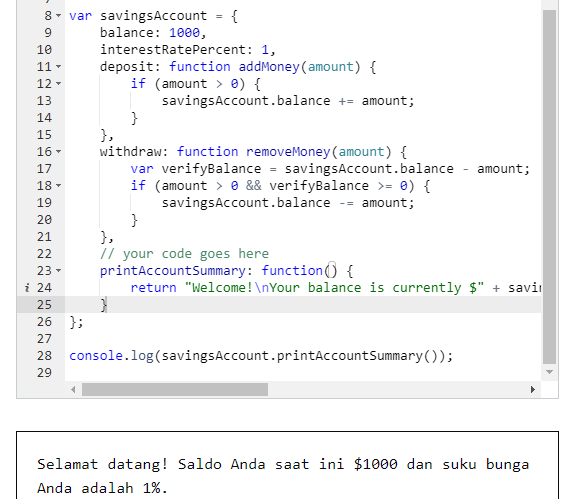
Mereka memiliki properti (informasi tentang objek) dan metode (fungsi atau kemampuan yang dimiliki objek). Objek adalah tipe data yang sangat kuat dan Anda akan melihatnya di semua tempat saat bekerja dengan JavaScript, atau bahasa pemrograman berorientasi objek lainnya.



1. Quiz: Menu Items (7-2)



1. Quiz: Bank Account 1 (7-3)



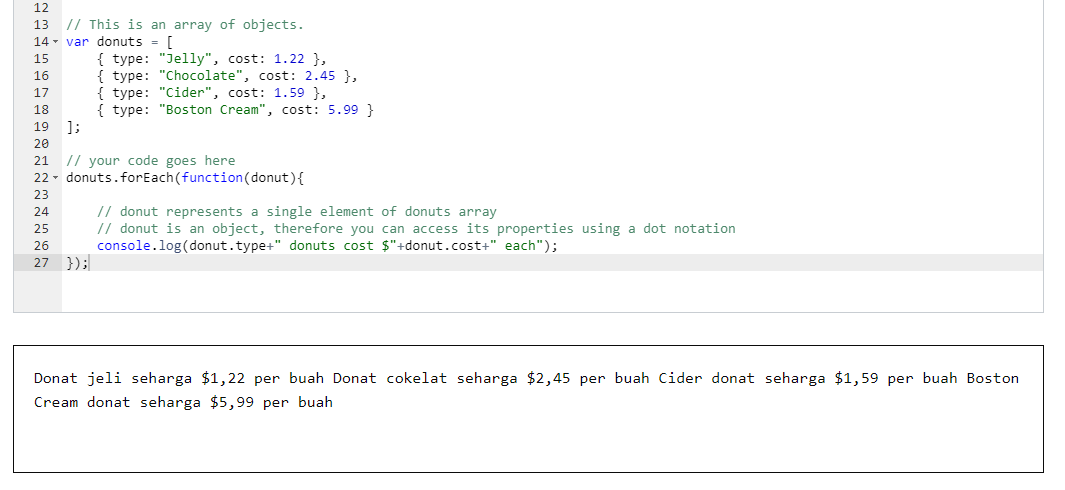
1. Quiz: Bank Account 2 (7-4)

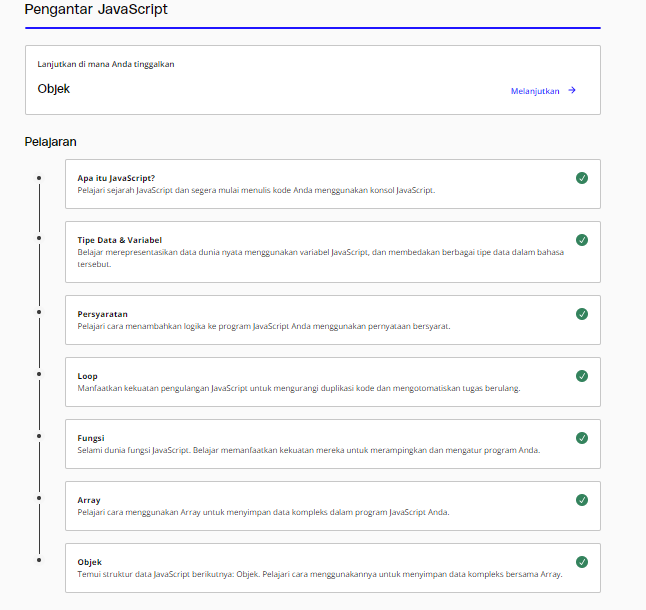


1. Quiz: Facebook Friends (7-5)



1. Quiz: Donut Review (7-6)





**Bukti bahwasannya telah menyelesaikan tugas rangkuman materi JavaScript Fundamentals**