

# Arrays

Basic data structures

Shadi Lahham - Programmazione web - Frontend - Javascript

# Arrays

# Array data type

An array is a type of data-type that holds an ordered list of values, of any type:

```
// Let arrayName = [element0, element1, ...];  
const rainbowColors = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Indigo', 'Violet'];  
let raceWinners = [33, 72, 64];  
let myFavoriteThings = ['Broccoli', 60481, 'Love Actually'];
```

The `length` property reports the size of the array:

```
console.log(rainbowColors.length);
```

# Array access

You can access items with "bracket notation". The index starts at 0.

```
let arrayItem = arrayName[indexNum];  
const rainbowColors = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Indigo', 'Violet'];  
let firstColor = rainbowColors[0];  
let lastColor = rainbowColors[6];
```

# Changing arrays

You can also use bracket notation to change the item in an array:

```
let myFavoriteThings = ['Broccoli', 60481, 'Love Actually'];  
myFavoriteThings[0] = 'Celery Root';
```

Or to add to an array:

```
myFavoriteThings[4] = 'Playgrounds';
```

You can also use the push method:

```
myFavoriteThings.push('Dancing');
```

# Creating a new array

```
let points = new Array();           // Bad
let points = [];                     // Good
```

## Why?

1. `new Array()` is slower
2. `new Array()` is not consistent

```
let points = [10];                  // an array with a single element of the value 10
let points = new Array(10);         // an array with 10 empty elements
```

*// these have the same result*

```
let points = [2, 10];
let points = new Array(2, 10);
```

# Loops with arrays and strings

Use a for loop to easily process each item in an array:

```
const rainbowColors = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Indigo', 'Violet'];
for (let i = 0; i < rainbowColors.length; i++) {
  console.log(rainbowColors[i]);
}
```

You can do the same for a string:

```
const rainbowColorsLetters = 'ROYGBIV';
for (let i = 0; i < rainbowColorsLetters.length; i++) {
  console.log(rainbowColorsLetters[i]);
}
```

# Array references

[JavaScript Arrays](#)

[JavaScript Array Reference](#)

[MDN - JavaScript Array Reference](#)

Read carefully. You will need some string methods for the exercises

Check browser compatibility before using since not all Array methods are supported on all browsers

**Note:** You can check on [CanIUse](#) or [MDN](#)



Your turn

# 1. Your top choices

Create an array to hold your top choices (colors, pets, books, whatever).

For each choice, log to the screen a string like: "My #1 choice is blue."

**Bonus:** Change it to add the correct number suffix, e.g. "My **1st** choice, "My **2nd** choice", "My **3rd** choice", "My **4th** choice", etc.

## 2.The Word Guesser

You will be implementing a game similar to hangman:

[Hangman \(game\)](#)

[PlayHangman.com - Play Hangman Game](#)

Instructions on next slides.

Continues on next page >>>

## 2.The Word Guesser

- Create two arrays:
  - one for the letters of the word (e.g. 'C', 'A', 'T')
  - Another for the current guessed letters (start with '\_', '\_', '\_' and add the correct letters to it).
- Write a function called guessLetter that should:
  - Take one argument, a letter.
  - Have a maximum number of guesses (e.g. 6)
  - Check if the letter is in the word array.
  - If the letter matches, add it in the correct position of the guessed array.
  - Show the user the current guessed letters.
  - Tell the user if they guessed a correct letter.
  - Tell the user how many guesses remain.
  - Tell the user if they won or lost the game.

Continues on next page >>>

## 2.The Word Guesser

Call your function to make guesses:

```
guessLetter('G');  
guessLetter('I');  
guessLetter('O');  
guessLetter('A');  
guessLetter('T');
```

- Bonus
  - Add a random reward for correct guesses and subtract a random amount for failed guesses.
  - Show the user the total reward (positive or negative).
  - Draw a hangman image to the console log after each guess.
  - Add a function that generates the letters to guess randomly.
  - Add a function that chooses the initial word to guess from an array of words.

## 3.Cut me up

- In the exercise folder create a .txt or .doc or .md file in which you explain the difference between the following array methods:
  - slice(), splice()
  - Explain the differences in terms of parameters and behavior
  - Provide code examples to prove your point

Bonus

## 4.Abracadabra

- Code 3 *different* solutions to change the 4th letter in the following string "Abracadabra" into an "X"
  - Each solution should be in a separate folder.
    - Name them solution-1, solution-2, etc.
  - Also include a doc file in which you explain what 3 ways you used
- 
- **Bonus:** There are many ways to replace a character in a string. Code other solutions than the above 3



# References

[JavaScript Arrays](#)

[JavaScript Array Reference](#)

[MDN - Array - JavaScript](#)