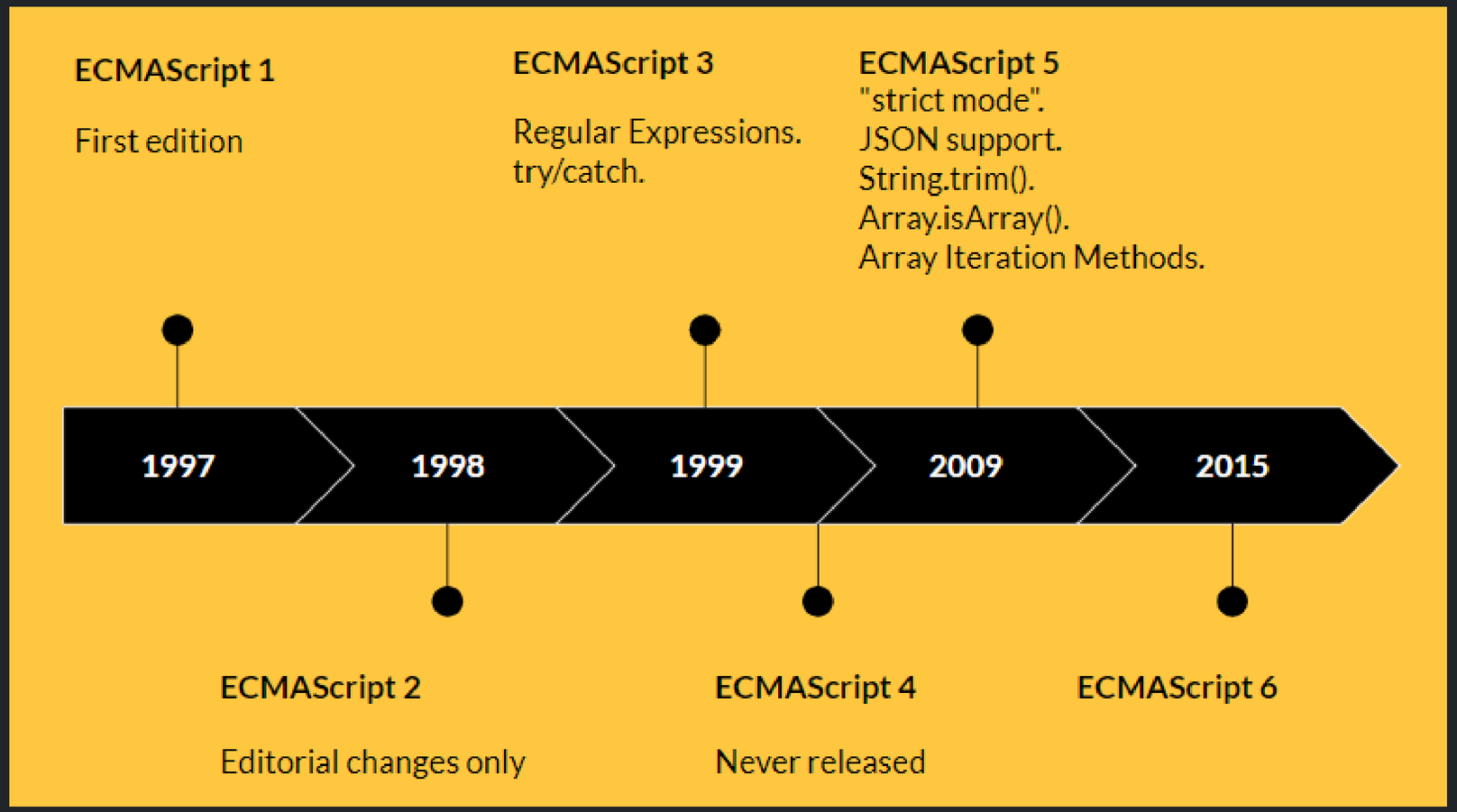
ES .NEXT

MORDERN JAVASCRIPT

TODAY'S AGENDA

- ECMASCRIPT HISTORY
- VARIABLES DECLARATION
- RESET PARAMETERS / SPEARD OPERATOR
- DESTRUCTING (ARRAY / OBJECT)
- ARROW FUNCTION
- API IMPROVEMENT AND NEW OBJECTS
- LAB

HISTORY OF ECMAScript



A new version of Ecma is released every year and so on ....

The TC39 Process

JS

STAGE 0

Ideas (Straw Man)

STAGE 1

Foraml Proposal

STAGE 2

Draft

STAGE 3

Candidiate

STAGE 4

Finished

Variable Declarations

JS

VAR vs LET vs CONST

	var	let	const
Stored in Global Scope	✓	✗	✗
Function Scope	✓	✓	✓
Block Scope	✗	✓	✓
Can Be Reassigned?	✓	✓	✗
Can Be Redeclared?	✓	✗	✗
Can Be Hoisted?	✓	✗	✗

Rest Parameter

- Reset parameter must come at the end of the parameters list

- Reset parameter must be with function signature only

Spread Operator

- Easier in concatenating arrays
- Deep copying arrays and objects
- Can call function with array of params

Destructuring

- Allow us to extract values from array or object
- Can be used to swap values without temp
- Can skip values
- Can be used in the function parameters as well

Default Parameters

- ES6 make it possible to assign default parameters to functions to get a value instead of undefined
- Must come at the end of the params list

Arrow Function

One of JavaScript's most difficult topics, the this keyword, allow us to refer to the object that executes a method.

Its value is determined by where the function is called that uses this

Even after defining 'this' to work a certain way it can still change at any point in your program

Arrow Function

Rules to help determine the value of 'this':

- When you create an object using the new keyword with a constructor function/class this will refer to the new object inside the function.
- Using bind call, or apply will override the value inside a function, and you can hardcode its value for this.
- If a function is called on an object as a method, this will refer to the object that is calling it.

Arrow Function

Rules to help determine the value of 'this':

- If a function is exectued without any of the three pervious criteria being applied, this will refer to the global object, which is window in the browser.
- If you are using strict mode, this will be undefined.
- Arrow function ignore all the above rules, and the value of this is determined by the scope enclosed by the arrow function

Arrow Function

This

When it is inside of an object's method – the function's owner is the object. Thus the 'this' keyword is bound to the object. Yet when it is inside of a function, either stand alone or within another method, it will always refer to the window/global object.

String Methods

String have some new methods that help a lot.

Methods:

- `.startWith()`
- `.endWIth()`
- `.includes()`
- `.repeat()`
- `.trim()` / `.fromCodePonit()`

String Methods

For more methods check the following links:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String

https://exploringjs.com/es6/ch_strings.html

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Template_literals

Array Methods

Array have some new methods that help a lot.

Methods:

- `.some()`
- `.every()`
- `.find()`
- `.map()`
- `.filter()` / `.reduce()`

Number Methods

Use `0o` for octal, `0b` for binary, and `0x` for hexadecimal

Methods:

- `Number.isInteger()`
- `Number.isNaN()`
- `Number.isFinite()`
- `Number.EPSILON()`
- `Number.cbrt()`

Object

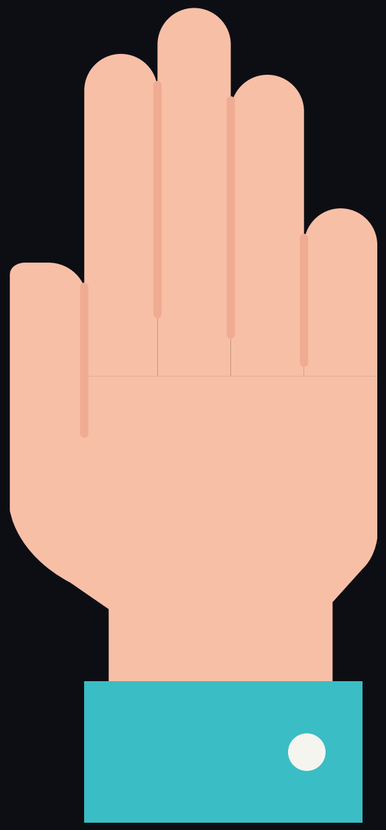
Some new methods can be used to enhance dealing with objects and resolve some problems

Methods:

- `.assign()`
- `.values() / .keys()`
- `.freeze() / .isFrozen`
- `.seal() / .isSealed()`

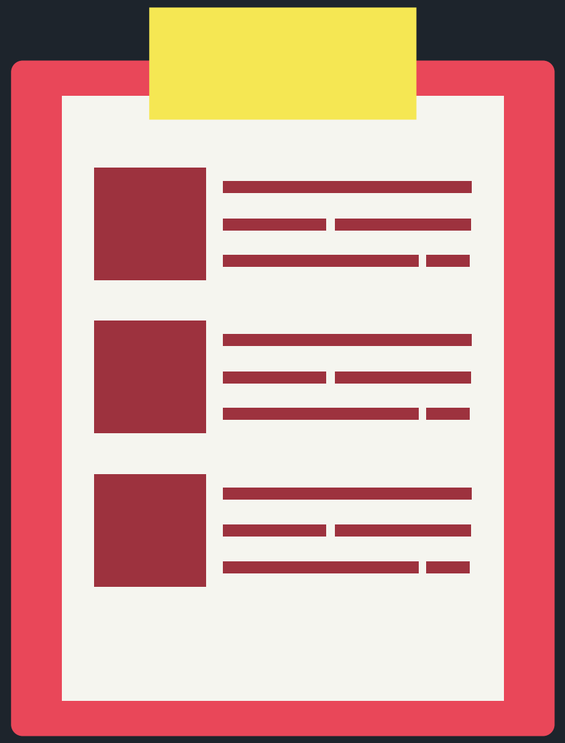
Options Object Parameter

- The options object is a widely used pattern that allows user-defined settings to be passed to a function in the form of properties on an object
- Options objects also make it easy to make parameters optional, when an optional parameter isn't passed in, a default value should be used instead
- Options object for a function with four or more arguments it's usually a good idea



THANK YOU

ANY QUESTIONS ?



LAB

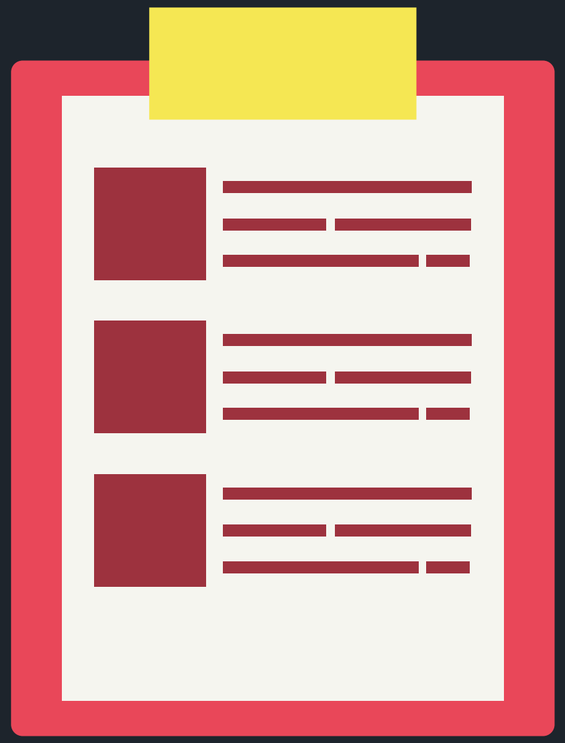
In the JS file, you will find a web designer object.

01

Change 'Your name' placholder to your name

02

Write a getAge() function that takes the years alive array as destructure and return your age, save the value you return in const called 'age'



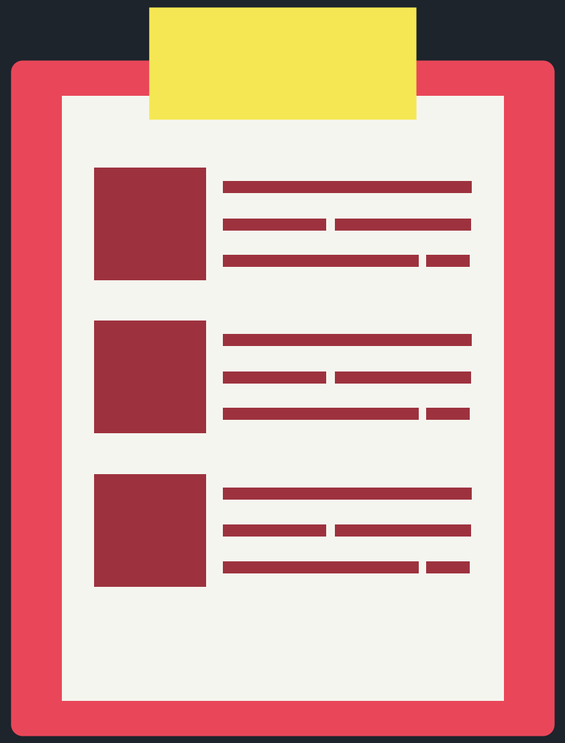
LAB

03

Divide the web designer skills into 2 variables designSkills and developmentSkills

04

Uncomment the newSkills array and merge the developmentSkills array with newSkills array in a new array 'updatedDevSkills'



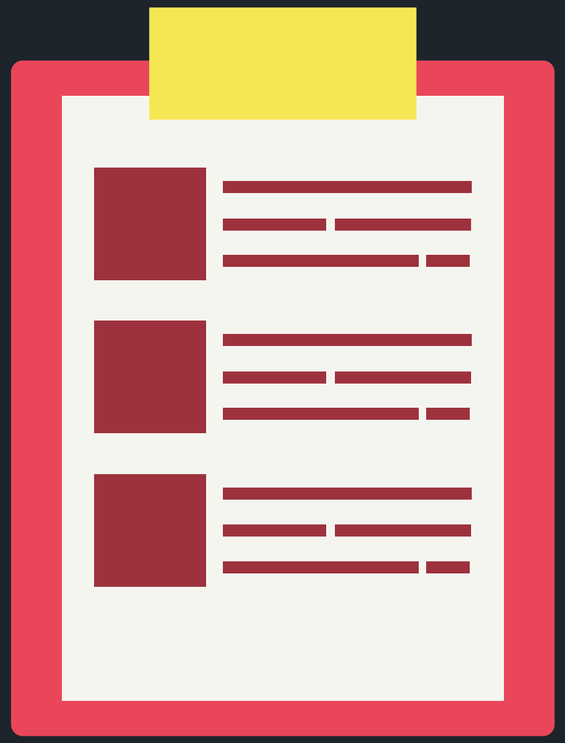
LAB

05

Destructure the diet and wirte a function getDrinks that takes the newly create drinks variables and returns drinks that contain the letter ' T ' call that function and save the returned values in variable 'tDrinks'

06

Uncomment the function buildID(), replace the placeholder data with the actual data



LAB

Using String API methods, Print the following star triangle like the following:

*

**

***

****

BOUNS:

Make it as a dynamic function and pass a default parmater.