Contents

[I. INTRODUCTION 2](#_Toc123419268)

[II. ES6 3](#_Toc123419269)

[1. Arrow Functions 3](#_Toc123419270)

[2. Enhanced Object Literals 3](#_Toc123419271)

[3. Destructuring 3](#_Toc123419272)

[4. Spread 5](#_Toc123419273)

[5. JS Modules 5](#_Toc123419274)

[III. REACT, REACT DOM 6](#_Toc123419275)

[1. document.createElement() 6](#_Toc123419276)

# INTRODUCTION

SPA: Single Page Application

MPA: Multi-Page Application

CSR: Client-Side Rendering

SSR: Server-Side Rendering

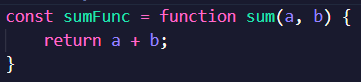
|  |  |
| --- | --- |
| **SPA** | **MPA** |
| * Modern * Trending * Page reloading is not required * Most necessary resources is loaded on the first run * Only need to load new resources when needed * Back-end and front-end independence * Not SEO friendly as MPA * Better user experience on mobile devices * JS dependency * Client-side rendering -> Reduce load for server | * Traditional * Need to reload page * Need to reload the entire page when accessing or routing * More SEO friendly than SPA |

# ES6

## Arrow Functions

-Reduce code length:

Normal:



Arrow:



-To return an object with arrow functions, we must put the object inside **()**:



-If there is **only 1 argument**, we can ignore **()**:



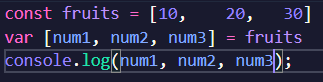
-Arrow function **does not have context** => **Cannot use *this***

## Enhanced Object Literals

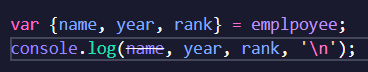
-In **reduce()** function, we must remember to return **accumulator**

## Destructuring

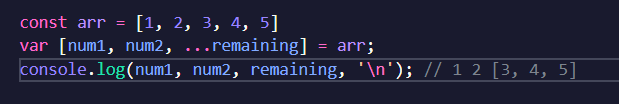
-Destructure an array:

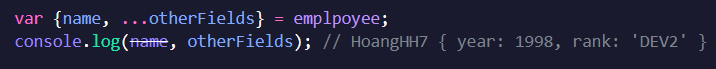


-Destructure an object:

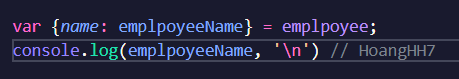


-Rest operator **…var**



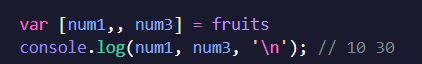


-Change variable’s name when working with objects:



-Omit some elements in the aray.

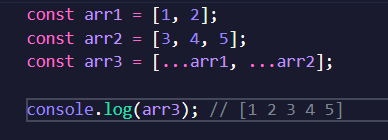
**Ex:** [10, 20, 30]



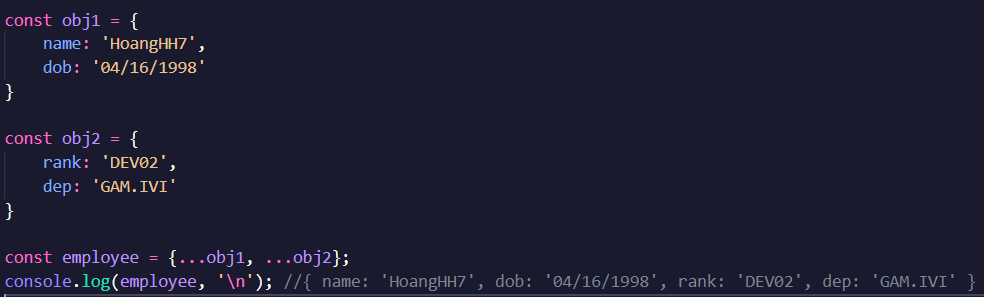
## Spread

-Use the spread operator to “remove” **[]** or **{}**

-Concatenating arrays:



-Concatenating objects:



## JS Modules

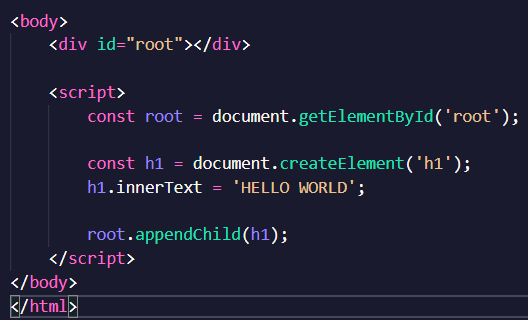
**export default** <function>

**import** <function> **from** <file>

# REACT, REACT DOM

## document.createElement()

-How JS can render HTML elements ?



## Add React to your project

-**NPM** = Node Package Manager

-**UNPKG**

+ Acquire NPM packages as CDN

+ Syntax:

**unpkg.com/:package@:version/:file**

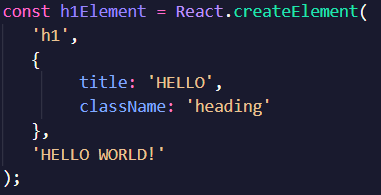
-To add React with UNPKG, use following script:

<script crossorigin src="https://unpkg.com/react@17.0.0/umd/react.production.min.js"></script>

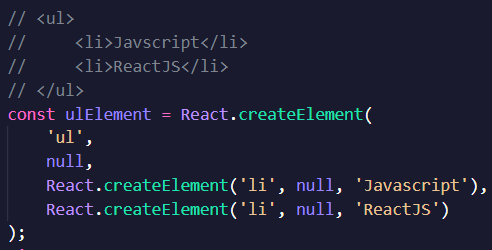
## React.createElement()

-React element is the smallest component of the React library.

-**React.createElement(**<type>, <props>, <childrens…>**)**. Children is also considered as **props**.



-Create element with multiple levels:



## React DOM

-Bridge between React and DOM

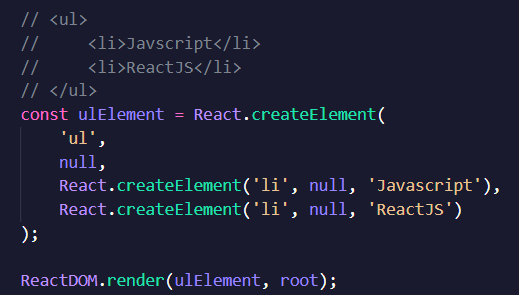
-ReactDOM supports rendering React elements.

-Add ReactDOM:

<script crossorigin src="https://unpkg.com/react-dom@17.0.0/umd/react-dom.production.min.js"></script>

-Syntax:

**ReactDOM.render(**<element>, <container>**)**

****