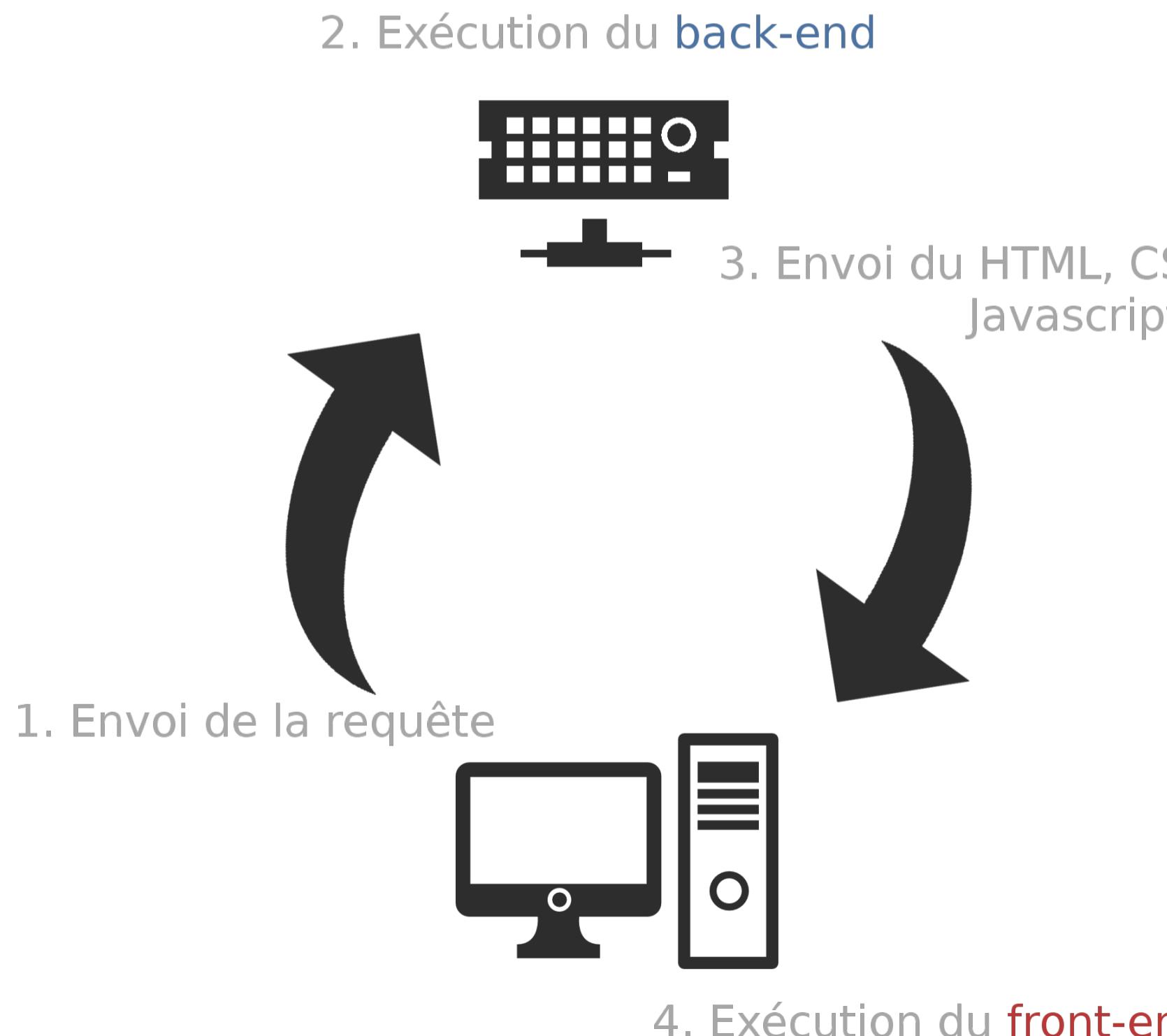


JS

JavaScript

Définition

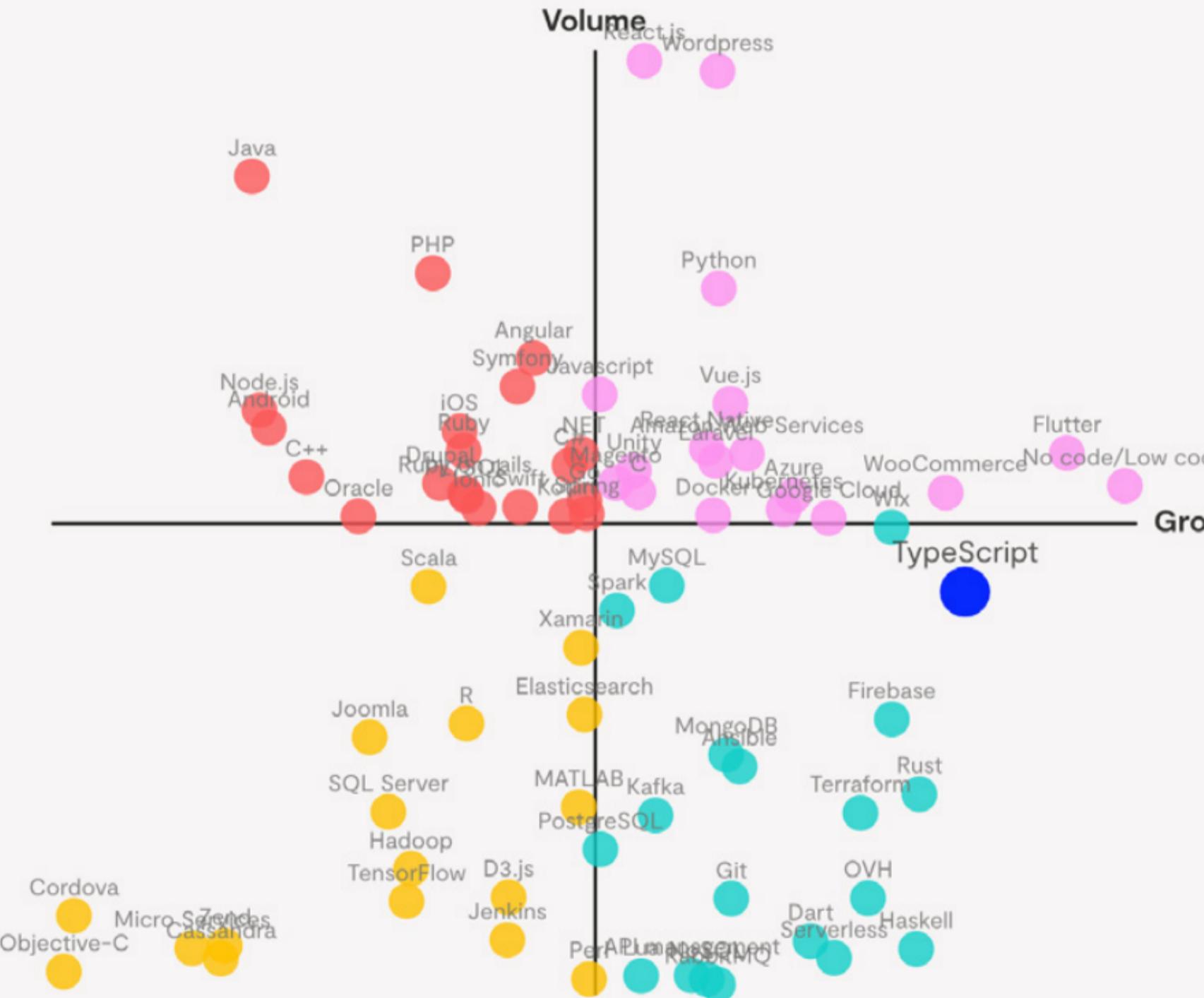
JavaScript est un langage de script, multi-plateforme et orienté objet. C'est un langage léger qui doit faire partie d'un environnement hôte (un navigateur web par exemple) pour qu'il puisse être utilisé sur les objets de cet environnement. (MDN)



Js

Popularité

Malt Quadrant

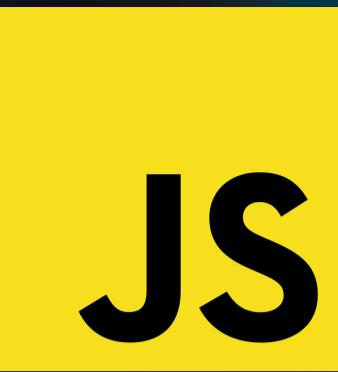


Malte trends 2021



JavaScript ≠ Java

Ecosystème JS



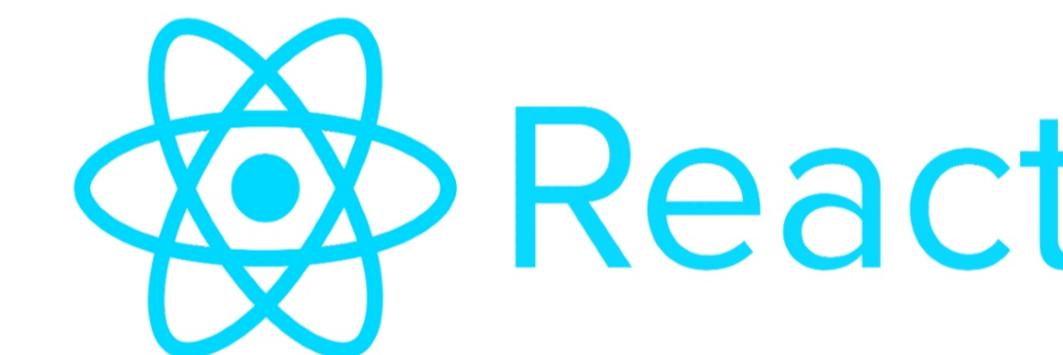
Applications Front



Angular



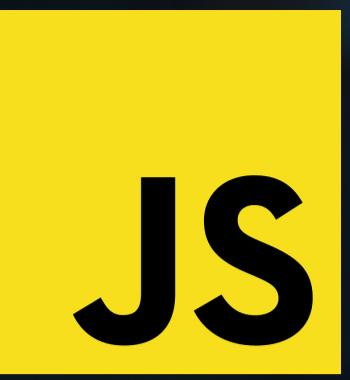
Vue.js



Backend



Qui utilise JS ?



NETFLIX

React + Nodejs



React



Ember Js



React



Angular

L'EQUIPE.FR

Vuejs

Uber

FusionJs + NodeJs

Programme des 6 jours

js

1 - Le DOM

Gestion de tous les événements qui se produisent sur une page web.

2 - La Data

Collecter, gérer, manipuler, afficher tous les types de données...

3 - Les API

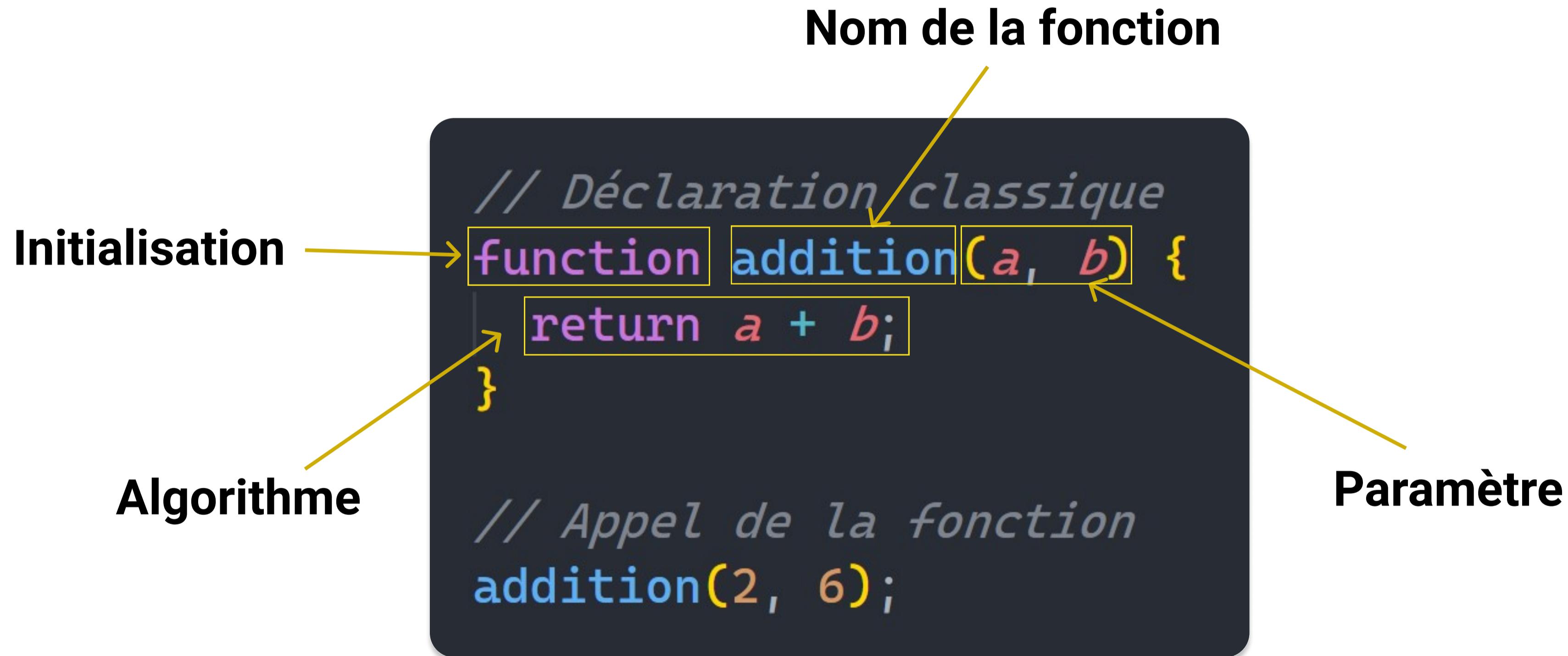
Travailler avec des API, créer des applications...

Les variables

```
// Chaîne de caractère (string)
const fisrtName = "Julien";  
  
// Nombre (number)
let age = 30;  
  
// Booléen (boolean)
let isAdmin = false;  
  
// Tableau (array)
let array = ["Bill", "Gates", 64];
```

L'élément à droite donne sa valeur à
celui de gauche

Les fonctions



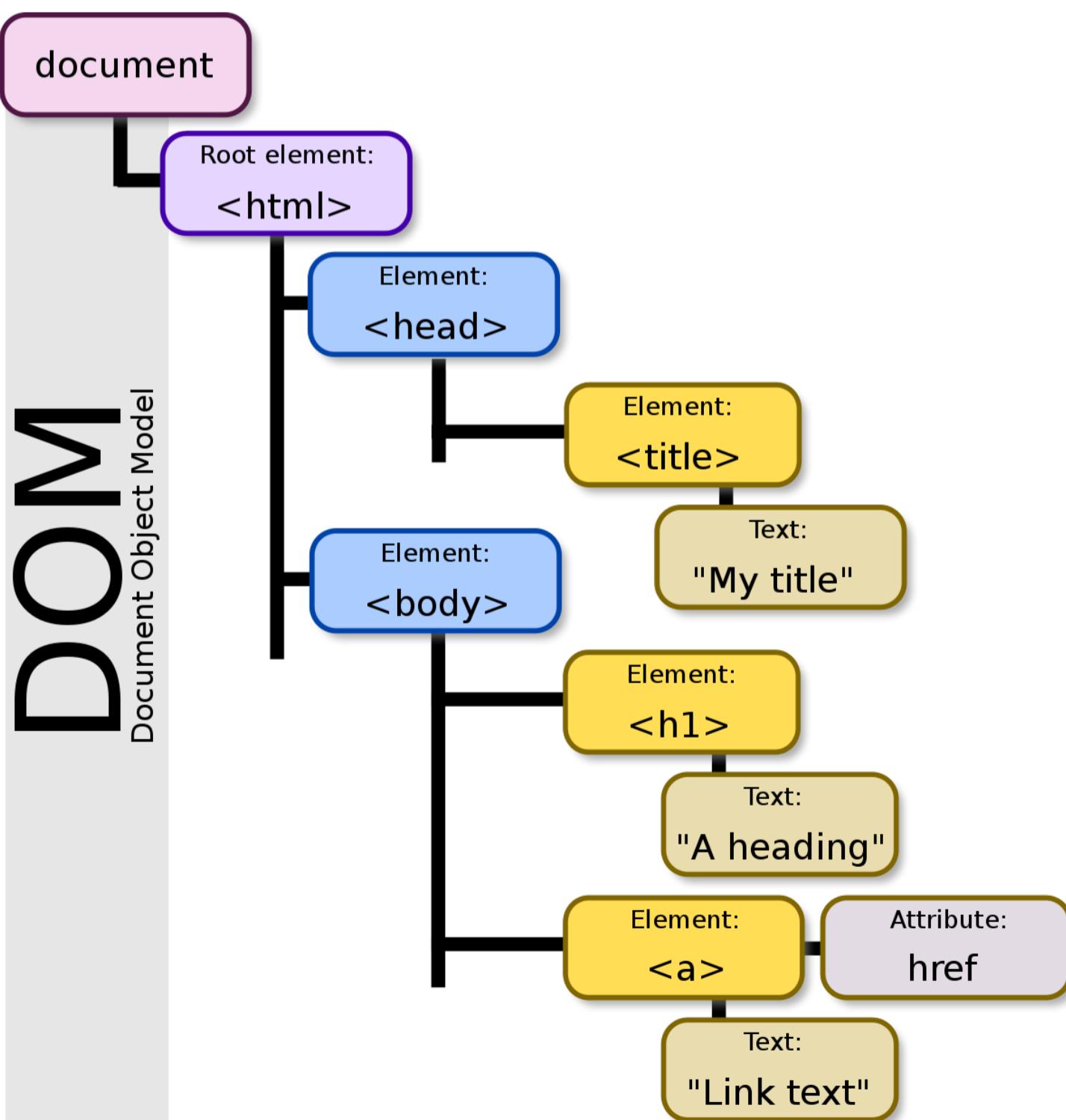
JS

1 - Le DOM

Définition du DOM

JS

Le Document Object Model (DOM) est une interface de programmation qui permet à des scripts d'examiner et de modifier le contenu du navigateur web. Par le DOM, la composition d'un document HTML est représentée sous forme d'un jeu d'objets, reliés selon une structure en arbre.



Pointer un élément

js

```
<h1 class="title">Javascript</h1>
```

```
const title = document.querySelector(".title");
```

La variable “title” symbolise la balise “h1” qui a la classe “title” sur le DOM

querySelector fonctionne comme un sélecteur CSS

Les événements (events) JS

```
const bouton = document.querySelector("#btn-1");
```

```
bouton.addEventListener("click", () => {  
    document.body.style.background = "yellow"  
});
```

Sélecteur

Propriété CSS

Evénement déclencheur

JS

2 - La Data

Les types de données

JS

```
let chaîne = "Ma chaîne de caractères";  
  
let number = 42;  
  
let booleen = true;  
  
let tableau = ["Je", "suis", "un", "tableau"];  
  
let objet = {  
    nom: "Denis",  
    age: 32,  
    admin: false,  
    technos: ["React", "Nodejs"],  
};
```

Afficher des données : .map

JS

le .map permet de “boucler” une série de données

On injecte le résultat du map sur le body

Les données à boucler sont stockées dans un tableau

Nom générique que l'on donne à chaque tour de boucle

```
document.body.innerHTML = monTableau.map((user) =>
  <div class="card">
    <h2>${user.nom}</h2>
    <p>${user.age} ans</p>
  </div>
).join("");
```

JS

3 - Les API

Le “fetch” de base

“fetch” demande à JS d’aller chercher les données présentes sur le lien

```
//Structure d'un fetch
fetch('monlien.com')
  .then((res) => res.json())
  .then((data) => console.log(data))
```

Le “then” est une “promesse”, il attend que la partie précédente soit totalement exécutée avant de se lancer

On met dans le paramètre suivant le “then” la nom que l’on veut. Ce nom représente toute les données qui ont été traitée avant