Javascript

console.log(“Hello World !”)

Javascript en quelques mots

- Langage interprété
- Orienté objet, impératif, fonctionnel
- Créé le 4 decembre 1995
- Standardisé sous le nom d'ECMAScript
- N'a **AUCUN** rapport avec Java !!!

Pourquoi utiliser Javascript?

- Simple à utiliser/apprendre
- Plein de concepts sympas (fonctionnel, asynchrone)
- Populaire (1er du classement Stackoverflow Survey 2019)
  => Très grosse communauté
- Parce que c'est moi qui décide

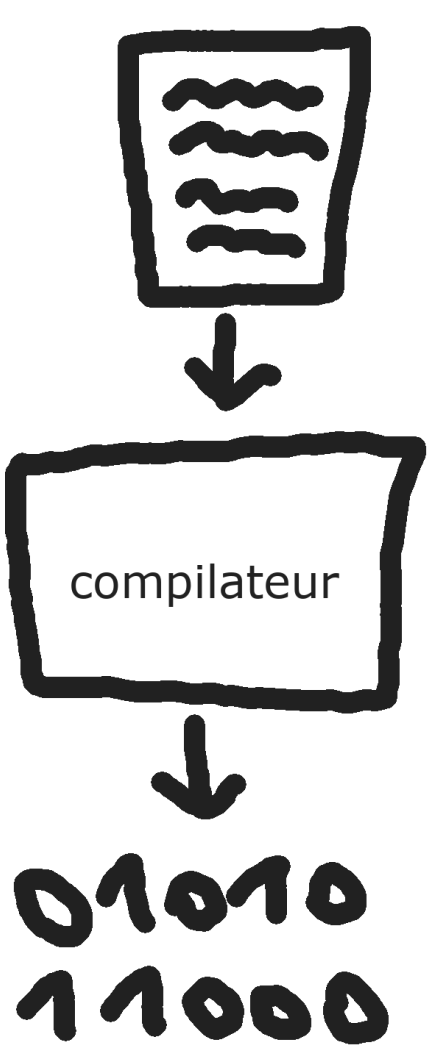
l'ECMAScript

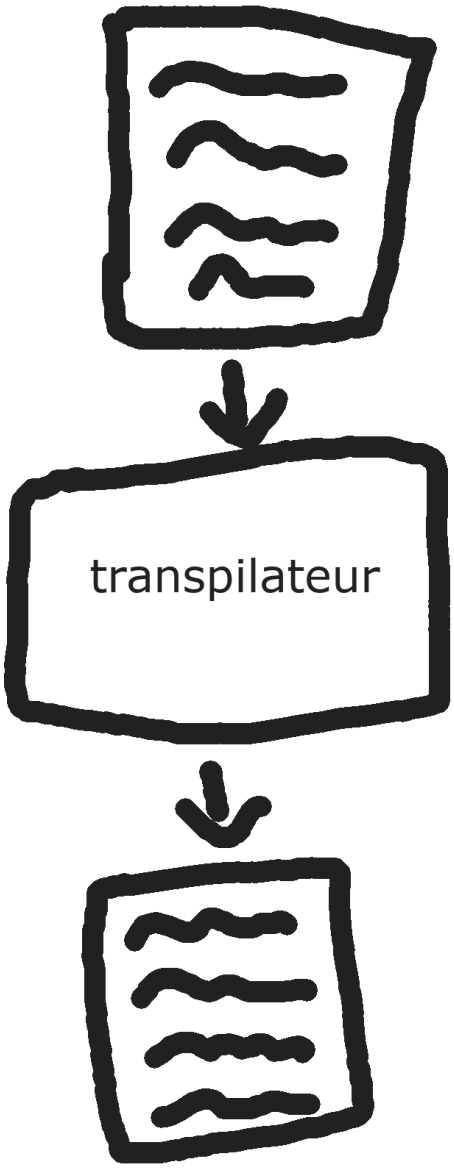
- Avant ES5, une version tous les 3-5ans
- A partir d'ES6 (ES2015), une version tous les ans
- Tous les navigateurs modernes supportent ES5
- Quelques navigateurs supportent ES6
- Aujourd'hui ES2019

Comment on utilise les dernières versions de Javascript ?

les transpileurs

- Le plus utilisé est BabelJS
- Permet de transformer un langage en un autre
- Ne pas confondre avec un compilateur
- ES2019 => ES5, plus de soucis de compatibilité !
- Les polyfills permettent également une meilleure rétrocompatibilité


compilateur
01010
11000


transpilateur

Quelques langages

- TypeScript
- ReasonML
- CoffeeScript
- et pleins d’autres…

Et ça ressemble à quoi ?

Les variables

```
<span class="c1">// Déclarer une variable
</span><span class="c1"></span><span class="kd">let</span> <span class="nx">var1</
<span class="kd">let</span> <span class="nx">var2</span> <span class="o">=</span>
<span class="kd">let</span> <span class="nx">var3</span> <span class="o">=</span>

<span class="nx">var1</span> <span class="o">=</span> <span class="mi">5</span>
<span class="nx">var1</span> <span class="o">=</span> <span class="s2">"tutu"</spa

<span class="c1">// Déclarer une constante
</span><span class="c1"></span><span class="k">const</span> <span class="nx">const
<span class="c1">// const1 = "tata" /!\ IMPOSSIBLE
</span>
```

⚠️ Le mot-clé `var` a été déprécié.

Les conditions

```
<span class="k">const</span> <span class="nx">i</span> <span class="o">=</span> <s

<span class="k">if</span><span class="p">(</span> <span class="nx">i</span> <span
  <span class="nx">console</span><span class="p">.</span><span class="nx">log</spa
<span class="p">}</span> <span class="k">else</span> <span class="p">{</span>
  <span class="nx">console</span><span class="p">.</span><span class="nx">log</spa
<span class="p">}</span>

<span class="nx">i</span> <span class="o">==</span> <span class="s2">"0"</span> <s
</span><span class="c1"></span><span class="nx">i</span> <span class="o">==</span>
</span><span class="c1"></span><span class="nx">i</span> <span class="o">===</spa
</span><span class="c1"></span><span class="nx">i</span> <span class="o">===</spa
</span>
```

Les boucles

```
<span class="kd">let</span> <span class="nx">i</span>
<span class="k">while</span> <span class="p">(</span> <span class="nx">i</span> <s
    <span class="nx">console</span><span class="p">.</span><span class="nx">log</s
    <span class="nx">i</span><span class="o">++</span>
<span class="p">}</span>

<span class="k">for</span> <span class="p">(</span> <span class="kd">let</span> <s
    <span class="nx">console</span><span class="p">.</span><span class="nx">log</s
<span class="p">}</span>
```

Les fonctions

```
<span class="kd">function</span> <span class="nx">add</span> <span class="p">(</sp
    <span class="k">return</span> <span class="nx">val1</span> <span class="o">+</
<span class="p">}</span>
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">add2</span> <span class="o">=</span>
    <span class="k">return</span> <span class="nx">val1</span> <span class="o">+</
<span class="p">}</span>
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">add3</span> <span class="o">=</span>
    <span class="k">return</span> <span class="nx">val1</span> <span class="o">+</
<span class="p">}</span>
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">add4</span> <span class="o">=</span>
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">add5</span> <span class="o">=</span>
```

Les structures de données

```
<span class="k">const</span> <span class="nx">tableau</span> <span class="o">=</sp
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span><span class="nx">console</span><span class="p">.</s
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">dictionnaire</span> <span class="o">
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span><span class="c1"></span><span class="nx">console</span><span class="p">.</s
</span><span class="c1"></span><span class="nx">console</span><span class="p">.</s
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">collection</span> <span class="o">=<
<span class="nx">console</span><span class="p">.</span><span class="nx">log</span>
</span>
```

Quelques méthodes utiles

```
<span class="k">const</span> <span class="nx">users</span> <span class="o">=</spa
<span class="nx">users</span><span class="p">.</span><span class="nx">forEach</spa
<span class="c1">// toto titi tutu
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">uppercaseUsers</span> <span class="
<span class="c1">// ["TOTO", "TITI", "TUTU"]
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">majorUsers</span> <span class="o">=
<span class="c1">// [{name: "toto", age: 18}, {name: "titi", age: 22}]
</span><span class="c1"></span>
<span class="k">const</span> <span class="nx">toto</span> <span class="o">=</span
<span class="c1">// {name: "toto", age: 18}
</span>
```

Mais Javascript n’est pas fait pour être exécuté sur un navigateur ?!

Nodes JS

- Interpréteur Javascript (machine virtuelle)
- Utilise le V8 de Google créé en 2009
- est livré avec `npm` (Node Package Manager)

Comment ça marche ?

```
<span class="c1"># Lancer un script</span>
node fichier.js

<span class="c1"># Initialiser un projet</span>
npm init

<span class="c1"># Installer une dépendance</span>
npm install dependance

<span class="c1"># Lancer un script</span>
npm run script
```

package.json

```
<span class="p">{</span>
  <span class="nt">"name"</span><span class="p">:</span> <span class="s2">"nom"</s
  <span class="nt">"description"</span><span class="p">:</span> <span class="s2">"
  <span class="nt">"version"</span><span class="p">:</span> <span class="s2">"0.1
  <span class="nt">"scripts"</span><span class="p">:</span> <span class="p">{</spa
    <span class="nt">"start"</span><span class="p">:</span> <span class="s2">"echo
  <span class="p">},</span>
  <span class="nt">"dependencies"</span><span class="p">:</span> <span class="p">{
    <span class="nt">"hello-world"</span><span class="p">:</span> <span class="s2"
  <span class="p">},</span>
<span class="p">}</span>
```