

L'outillage javascript.

Ahmadou Waly Ndiaye (Sir Kane)
TheCodingMachine, Lead developer



Google
Developer
Groups

Sommaire

(1)

Minification

(2)

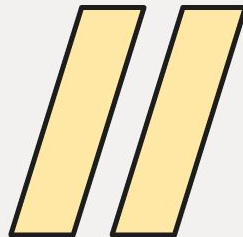
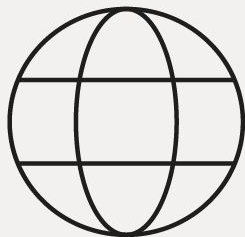
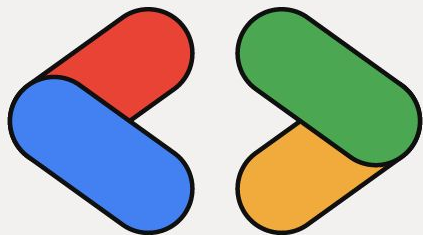
Bundling

(3)

Tree Shaking

(4)

Code splitting



(1)

Minification



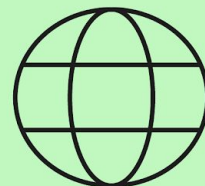
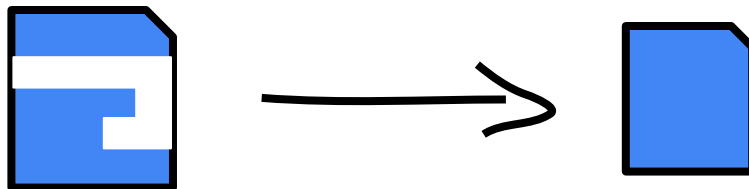
Google
Developer
Groups

Web @DevFest



Google
Developer
Groups

**Objectif: Réduire le poids des fichiers
à charger par le navigateur**



Web @DevFest

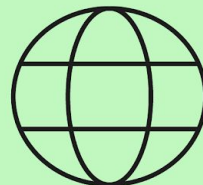


Google
Developer
Groups

math.js

```
export function add(firstNumber, secondNumber) {  
  return firstNumber + secondNumber;  
}  
  
// Recursivity  
export function factorial(number) {  
  if (number === 0) {  
    return 1;  
  }  
  
  return number * factorial(number - 1);  
}
```

228



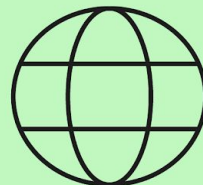
Web @DevFest



Google
Developer
Groups

math.js

```
export function add(firstNumber, secondNumber) {  
  return firstNumber + secondNumber;  
}  
  
// Recursivity  
export function factorial(number) {  
  if (number === 0) {  
    return 1;  
  }  
  
  return number * factorial(number - 1);  
}
```



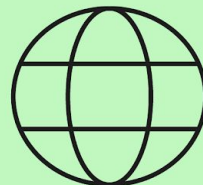
Web @DevFest



Google
Developer
Groups

math.min.js

```
export function add(firstNumber, secondNumber) {  
  return firstNumber + secondNumber  
}  
  
export function factorial(number) {  
  if (number === 0)  
    return 1  
  return number * factorial(number - 1)  
}
```



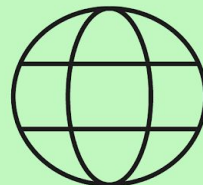
Web @DevFest



Google
Developer
Groups

math.min.js

```
export function add(firstNumber, secondNumber) {  
  return firstNumber + secondNumber  
}  
  
export function factorial(number) {  
  return number === 0  
    ? 1  
    : number * factorial(number - 1)  
}
```



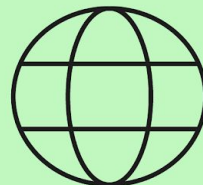
Web @DevFest



Google
Developer
Groups

math.min.js

```
export function add (f,s){  
  return f+s  
}  
  
export function factorial (n)  
{  
  return === 0  
    ? 1  
    : n * factorial(n - 1)  
}
```



Web @DevFest

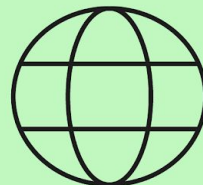


Google
Developer
Groups

math.min.js

```
export function add(f,s){return f+s}export function factorial(n){return n===0?  
1:n*factorial(n - 1)}
```

98



Minification: Outils



2018



2019



2020

Benchmark: <https://esbuild.github.io/>

Web @DevFest



Google
Developer
Groups

LCP

Largest Contentful Paint



Une bonne valeur LCP est de 2,5 secondes ou moins.

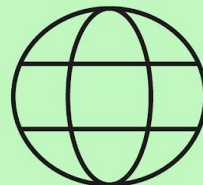
FCP

First Contentful Paint



Les bonnes valeurs FCP sont inférieures ou égales à 1,8 seconde. Les valeurs médiocres sont supérieures à 3 secondes.

<https://web.dev/articles/lcp?hl=fr#what-is-a-good-lcp-score>
<https://web.dev/articles/fcp?hl=fr#what-is-a-good-lcp-score>



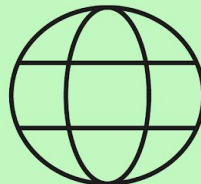
Web @DevFest

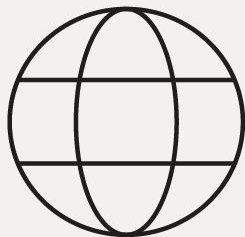
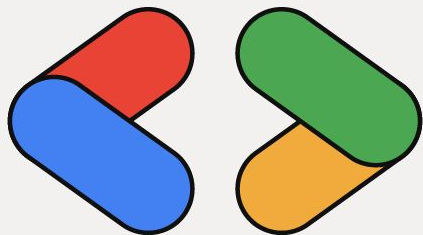


Google
Developer
Groups

"Don't minify what you don't need. Minification without tree-shaking is like compressing a trash can without taking out the garbage."

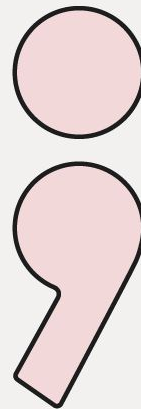
Philip Walton





(2)

Bundling



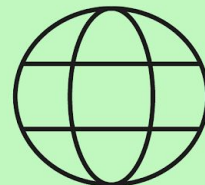
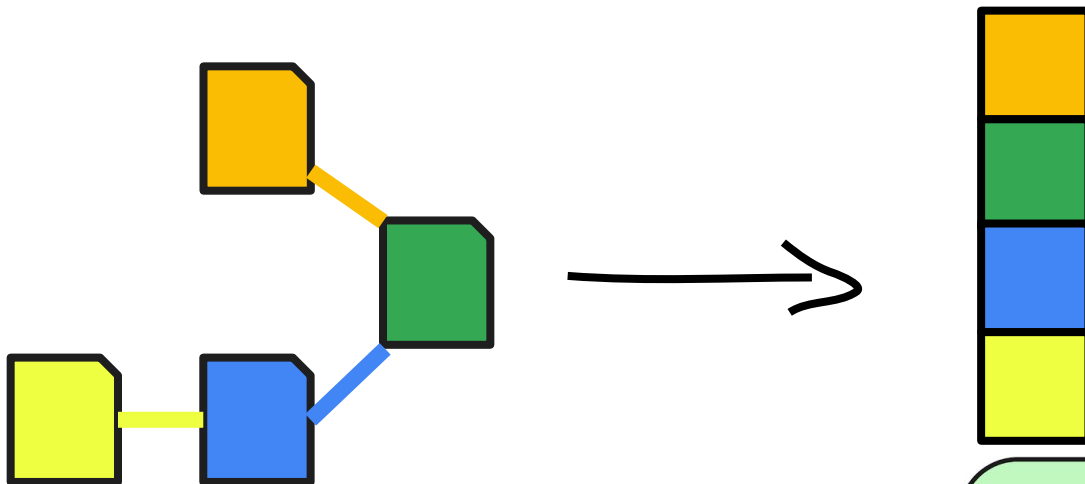
Google
Developer
Groups

Web @DevFest



Google
Developer
Groups

Objectif: Concatener les fichiers en 1



Web @DevFest



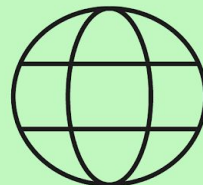
Google
Developer
Groups

```
// math.js  
export const add (a, b) {  
  return a + b  
}
```

```
// index.js  
import { add } from './math'  
  
console.log(add(2, 3))
```



```
// bundle.js  
const add (a, b) {  
  return a + b  
}  
  
console.log(add(2, 3))
```

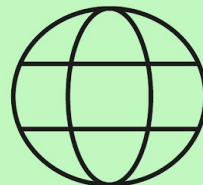


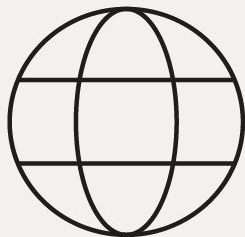
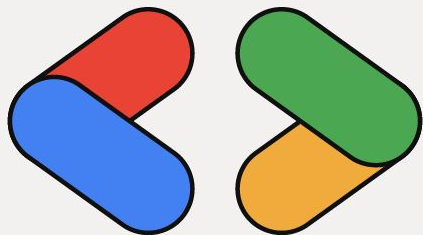
Web @DevFest



Google
Developer
Groups

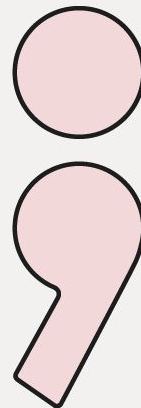
DEMO 





(3)

Tree Shaking



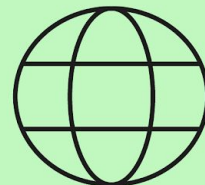
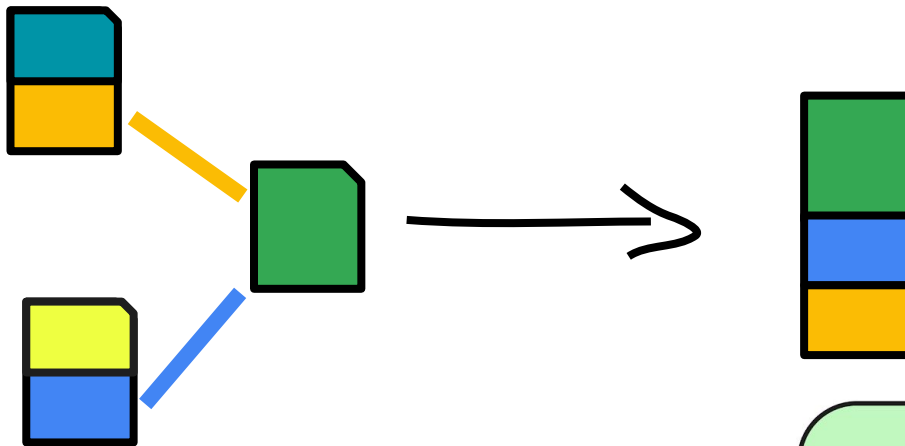
Google
Developer
Groups

Web @DevFest



Google
Developer
Groups

Objectif: **Supprimer le code non utilisé**



Web @DevFest



Google
Developer
Groups

```
// math.js
export const add (a, b) {
  return a + b
}
```

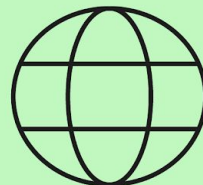
```
// index.js
import { add } from './math'

console.log(add(2, 3))
```



```
// bundle.js
const add (a, b) {
  return a + b
}

console.log(add(2, 3))
```



Web @DevFest



Google
Developer
Groups

Code non tree shakable

```
// math.js
const math = {
  add: (a, b) => a + b,
  subtract: (a, b) => a - b
}
```

```
export default math
```

```
// index.js
import math from './math'

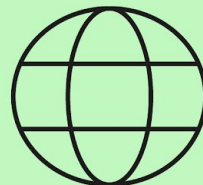
console.log(math.add(2, 3))
```

Code tree shakable

```
// math.js
export const add = (a, b) => a + b
export const subtract = (a, b) => a - b
```

```
// index.js
import { add } from './math'
console.log(add(2, 3))
```

VS



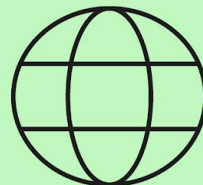
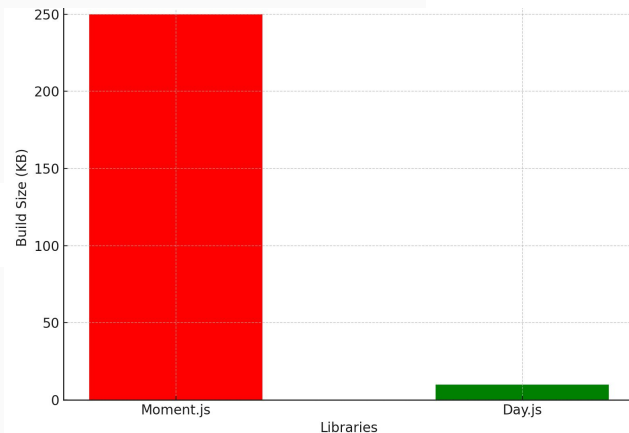
Web @DevFest



Google
Developer
Groups

```
// momentjs  
import moment from "moment";  
const today = moment().format("YYYY-MM-DD");
```

```
// Dayjs  
import dayjs from "dayjs";  
const today = dayjs().format("YYYY-MM-DD");
```

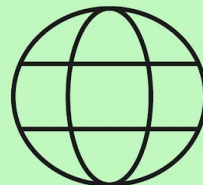


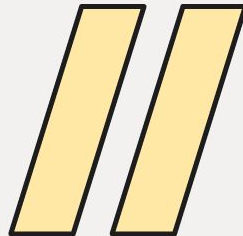
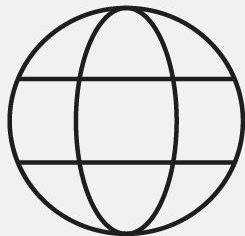
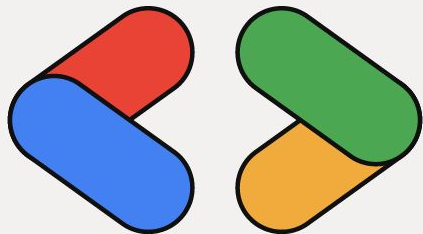
Web @DevFest



Google
Developer
Groups

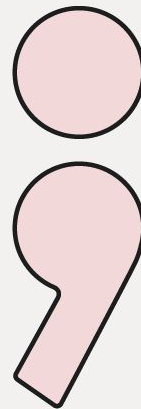
DEMO 





(4)

Code splitting



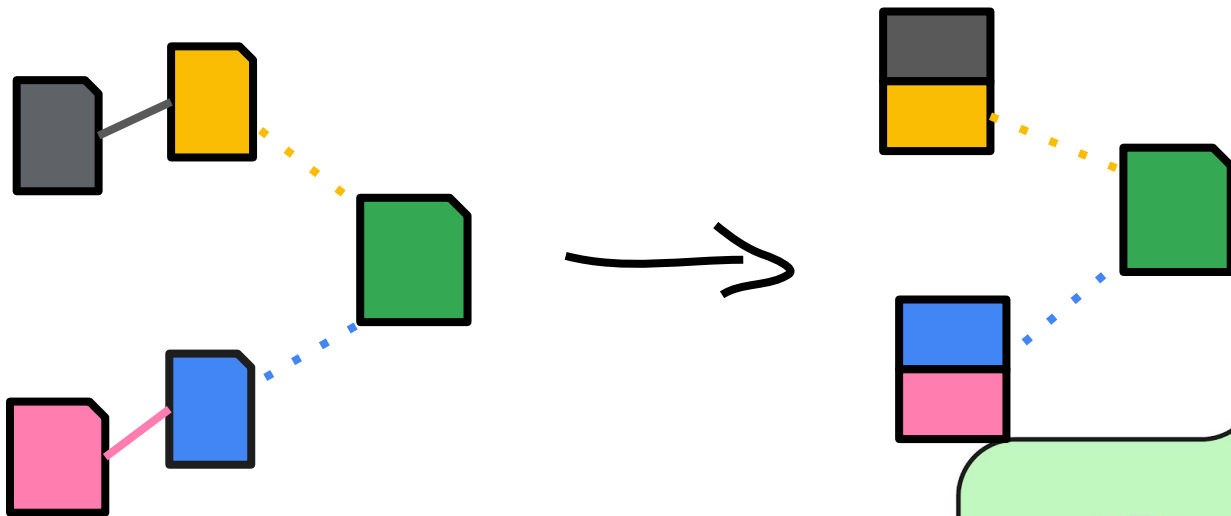
Google
Developer
Groups

Web @DevFest



Google
Developer
Groups

Objectif: Charger le fichier à la demande



Web @DevFest



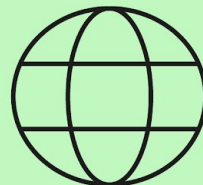
Google
Developer
Groups

Refs:

- L'import dynamique

```
document.querySelector('button').addEventListener("click", async () => {  
  const { hello } = await import("./utils/hello.js");  
  console.log(hello("Sir Kane"));  
});
```

- Le composant “defineAsyncComponent” de VueJS
<https://vuejs.org/guide/components/async>
- Les directives client sur Astro
<https://docs.astro.build/fr/reference/directives-reference/#directives-client>

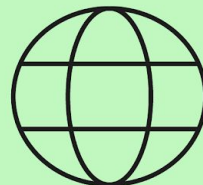


Web @DevFest



Google
Developer
Groups

DEMO 

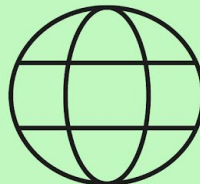


Les ressources / refs

- Les demos: <https://github.com/sir-kain/js-tools>
- Le code splitting en action avec Astro:
<https://docs.astro.build/fr/reference/directives-reference/#directives-client>
- Les metrics lighthouse
<https://web.dev/articles/lcp?hl=fr#what-is-a-good-lcp-score>
<https://web.dev/articles/fcp?hl=fr#what-is-a-good-lcp-score>
- Conf de Hubert Sablonnière sur la compression
<https://www.youtube.com/watch?v=zxijXQpRa4E>
- Bundlephobia pour consulter la taille d'un package npm
<https://bundlephobia.com/package/moment@2.30.1>



Google
Developer
Groups

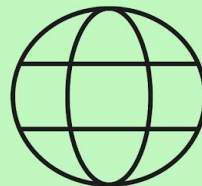


Web
@DevFest

MERCI ❤️



Google
Developer
Groups



Web
@DevFest