Développement front avancé



Danielo JEAN-LOUIS

Import de scripts

- Utilisation de la balise <script>
- Utilisation de modules
 - · Via npm et un bundler (S5)
 - · L'attribut type="module"
- Et...

importmap

- Nouveauté gérée par tous les navigateurs depuis 2023
- Attribut de balise <script>
- Permet de lister les modules à charger et où les télécharger sous forme de dictionnaire

Source(s):

importmap

```
<script type="importmap">
    "imports": {
      "lodash": "https://cdn.skypack.dev/lodash",
      "moment": "https://cdn.skypack.dev/momentjs"
</script>
```

lci, nous définissons deux modules : lodash et moment

Source(s):

importmap

```
1 import lodash from 'lodash';
```

On importe notre module comme l'accoutumée (On n'oublie pas de rajouter "type=module" pour la balise <script>)

importmap - Avantages

- Simplification de la gestion des modules
 - · Raccourcissement des noms
 - Possibilité de gérer facilement plusieurs versions du même module
- Centralisation des versions de modules
- Pas besoin de bundlers ou de npm/yarn

Source(s):

Pratiquons! - importmaps (Partie 1/2)

Pré-requis:

Avoir la ressource ressources/importmap

A télécharger ici :

https://github.com/DanYellow/cours/raw/refs/heads/main/developpement-web-et-dispositif-interactif-s6/travaux-pratiques/numero-3/developpement-web-et-dispositif-interactif-s6_travaux-pratiques_numero-3.ressources.zip

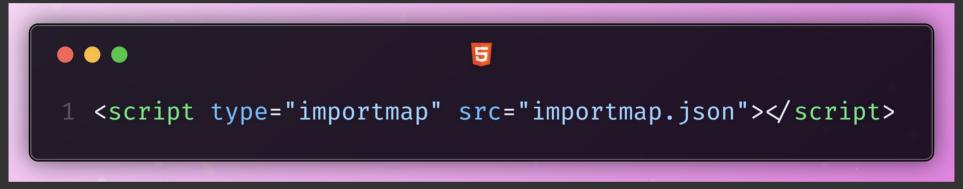
Importmap – Fichier externe

- Fichier au format json
- Doit avoir le MIME type "application/importmap+json"
 - · Pas géré par tous les navigateurs (10/2024)

Source(s) :

• https://github.com/WICG/import-maps?tab=readme-ov-file#installation

importmap – Fichier externe



On charge un importmap externe si son MIME type est "application/importmap+json".

Il est possible, à l'avenir, que l'extension ".importmap" arrive.

Source(s):

- https://github.com/WICG/import-maps?tab=readme-ov-file#installation anglais
- https://www.honeybadger.io/blog/import-maps/ anglais

importmap - Scopes

- Possibilité de charger les modules en fonction de l'emplacement du module qui les charge
- Définition d'une clé "scopes" dans l'importmap

Source(s):

Importmap - Scopes

```
<script type="importmap">
  "imports": {
    "lodash": "https://cdn.jsdelivr.net/npm/lodash@4.17.21/+esm"
 },
  "scopes": {
  "/test/": {
      "utils": "./test/utils.js"
</script>
```

Le module "utils" ne sera disponible que si le module est dans un dossier "test"

Importmap - Scopes

 Si un module matche plusieurs modules identiques, le navigateur utilisera celui avec le chemin le plus précis

Source(s):

Importmap - Scopes

```
<script type="importmap">
  "imports": {
    "lodash": "https://cdn.jsdelivr.net/npm/lodash@4.17.21/+esm"
  },
  "scopes": {
  "/test/": {
      "lodash": "https://cdn.jsdelivr.net/npm/lodash@4.6.0/+esm"
</script>
```

Si un module est contenu dans le dossier "test", il chargera la version 4.6.0 de lodash, les autres la version 4.17.21

Pratiquons! - importmaps (Partie 3)

Pré-requis:

Avoir la ressource ressources/importmap

A télécharger ici :

https://github.com/DanYellow/cours/raw/refs/heads/main/developpement-web-et-dispositif-interactif-s6/travaux-pratiques/numero-3/developpement-web-et-dispositif-interactif-s6_travaux-pratiques_numero-3.ressources.zip

Importmap - Limites

- Pas adapté pour les gros projets
 - · A préférer pour le prototypage
- Impossibilité de :
 - · Faire du tree shaking ou du hot reloading
 - · Charger plusieurs importmap par page

Source(s):

Questions?