

Développement web S1 (R112) – TP #3

Liens utiles : [CM #1](#) · [TP #2](#)

Avant de commencer, créez un répertoire nommé r112-tp3 contenant des répertoires exo1, exo2, etc. Chacun de ces répertoires contient votre travail sur l'exercice correspondant. Le fichier HTML de chaque exercice doit s'appeler index.html. **Vos exercices ne seront pas corrigés si vous ne respectez pas ces instructions.**

À la fin de la séance, vous devez compresser (en format ZIP ou RAR) le répertoire r112-tp3 avec tous vos exercices et le soumettre sur AMeTICE.

Seuls les exercices marqués avec une étoile (*) seront évalués. La date limite pour rendre vos exercices est visible sur AMeTICE, et aucun travail ne sera accepté après. Le corrigé sera accessible après l'évaluation des exercices.

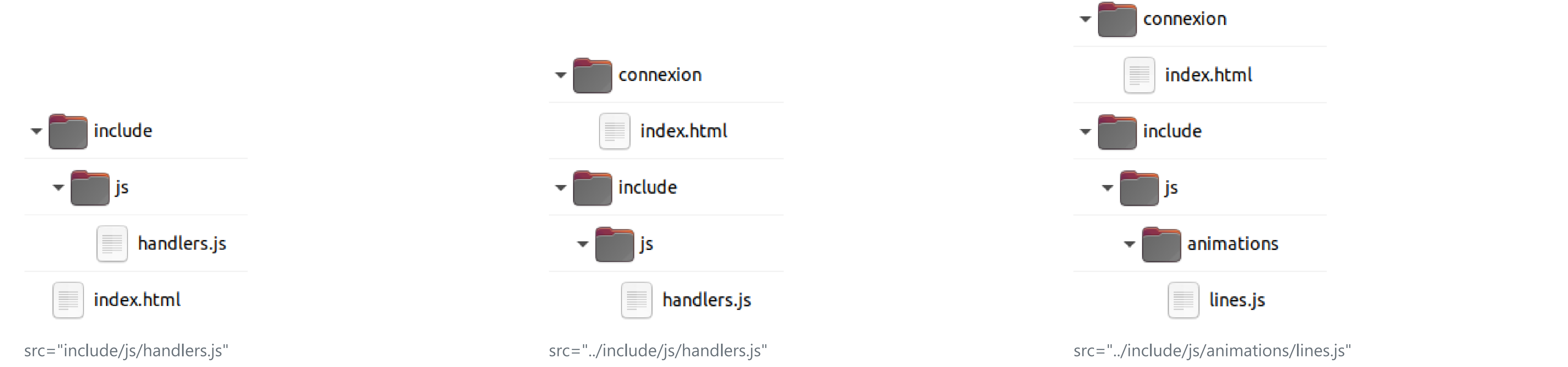
Code JavaScript externe

Un document HTML peut aussi lire du code JavaScript dans un fichier externe comme ceci :

```
<script src="script.js" defer></script>
```

La valeur de l'attribut `src` (comme *source*) est l'adresse relative du fichier qui contient le code JavaScript.

- Si le document JavaScript se trouve dans le même dossier que le document HTML, il suffit d'écrire son nom.
- Si le document JavaScript se trouve dans un dossier qui est dans le même dossier que le document HTML, il faut écrire le nom du dossier suivi du caractère *slash*, suivi du nom du fichier. Par exemple, `"include/script.js"`.
- Autres cas :



Vous devriez écrire votre code JavaScript dans un fichier séparé s'il va être utilisé par plusieurs pages ou si vous n'aimez pas mélanger les langages HTML et JavaScript dans le même fichier.

Exercices

Exercice 1

Écrivez un programme qui demande à l'utilisateur un nombre entre 0 et 100 et qui affiche si ce nombre est entre 0 et 24, entre 25 et 49, entre 50 et 74 ou entre 75 et 100.

Par exemple, si vous avez écrit «18», alors le message sera «18 est entre 0 et 24».

Exercice 2

Écrivez un programme qui affiche dans un seul message tous les jours de la semaine en bon français. Utilisez le tableau suivant :

```
let jours = ["lundi", "mardi", "mercredi", "jeudi", "vendredi", "samedi", "dimanche"]
```

Vous pouvez utiliser une boucle `for` pour avoir un code plus simple et élégant.

Exercice 3

Écrivez un programme qui affiche tous les nombres de 59 à 0 par ordre décroissant.

Cet exercice est très facile à faire si vous utilisez une boucle `for`.

Exercice 4* noté

Écrivez un programme qui demande à l'utilisateur d'écrire 10 mots et qui les affiche ensuite dans un seul message.

Chaque mot doit être rentré dans une fenêtre modale différente. Vous avez le choix entre :

- utiliser un tableau et une boucle `for`
- ou demander 100 nombres au lieu de 10.

Cet exemple montre comment créer un tableau vide et ajouter des valeurs avec l'instruction `push` :

```
let tableau = [] // déclaration d'un tableau vide
tableau.push("lundi") // ajouter la valeur "lundi"
tableau.push("mardi") // ajouter la valeur "mardi"
console.log(tableau) // affiche le tableau avec ses deux valeurs
```

Exercice 5

Répétez l'[exercice 4](#), mais cette fois il faut afficher les mots dans l'ordre inverse, en commençant par le dernier.

Exercice 6

Dans cet exercice vous allez faire un dictionnaire multilingue. Demandez à l'utilisateur d'écrire un jour de la semaine en français et affichez-le en espagnol, italien et allemand dans une phrase bien rédigée.

Utilisez ce tableau :

```
let jours = [
  ["lundi", "lunes", "lunedì", "montag"],
  ["mardi", "martes", "martedì", "dienstag"],
  ["mercredi", "miércoles", "mercoledì", "mittwoch"],
  ["jeudi", "jueves", "giovedì", "donnerstag"],
  ["vendredi", "viernes", "venerdì", "freitag"],
  ["samedi", "sábado", "sabato", "samstag"],
  ["dimanche", "domingo", "domenica", "sonntag"]
]
```

Exercice 7

Refaites l'[exercice 6](#) mais avec ce tableau :

```
let jours = [
  {"fr": "lundi", "es": "lunes", "it": "lunedì", "de": "montag"},
  {"fr": "mardi", "es": "martes", "it": "martedì", "de": "dienstag"},
  {"fr": "mercredi", "es": "miércoles", "it": "mercoledì", "de": "mittwoch"},
  {"fr": "jeudi", "es": "jueves", "it": "giovedì", "de": "donnerstag"},
  {"fr": "vendredi", "es": "viernes", "it": "venerdì", "de": "freitag"},
  {"fr": "samedi", "es": "sábado", "it": "sabato", "de": "samstag"},
  {"fr": "dimanche", "es": "domingo", "it": "domenica", "de": "sonntag"}
]
```

Vous pouvez aussi afficher un message si l'utilisateur écrit un mot qui n'est pas dans le dictionnaire.

Exercice 8

Faites un programme qui demande à l'utilisateur combien de nombres il va saisir, lui demande de les saisir et affiche la moyenne de tous ces nombres à la fin.

Vous pouvez faire cet exercice avec ou sans un tableau.

Exercice 9

Refaites l'[exercice 7](#) mais dans tous les sens possibles.

L'utilisateur peut écrire un jour de la semaine en français, espagnol, italien ou allemand, et le programme affichera sa traduction dans les autres langues.