

<b>Institut Galilée</b> <b>Paris 13</b>	<b>LANGAGES WEB – JAVASCRIPT</b> Exercice Articles	Réf. : WEBC [0.9] Page : I - 1
--	---	-----------------------------------

## Etape 1

Un début de structure HTML est présent dans index.html, et l'objectif est de remplir la balise <section> par autant d'articles présent dans le tableau de données. Pour chaque article, il faudra inclure une balise <article> qui contiendra le titre de l'article en h2 et un paragraphe pour le résumé de l'article.

Pour structurer votre travail, il faut prévoir au moins deux fonctions :

- creerElemAvecTexte(tagName, content) qui prend comme paramètre le nom d'un tag (balise), et du contenu (du texte). En retournant cet élément du DOM créé, elle vous servira à créer les éléments de chaque article. On devra utiliser la méthode du DOM `document.createElement()`.
- ajoutArticle(article) qui prend un article en paramètre, et retourne l'élément DOM à intégrer dans le conteneur section. C'est dans cette fonction que vous accrocherez les balises (article, h2 et p) les unes aux autres avec la fonction du DOM `append()`.

Il vous reste à parcourir (avec un "forEach" et une "Arrow Function") le tableau et à afficher tous les articles dans votre page. Testez également la syntaxe "for of". Il faudra donc "accrocher" vos articles à la balise section que vous récupérerez avec la méthode du DOM `document.querySelector()` par exemple.

Profitez-en pour tester également la méthode `document.getElementsByTagName()` qui retourne une HTMLCollection et la méthode `document.querySelectorAll()` qui retourne elle, une NodeList statique. Profitez-en pour lire/comprendre la documentation : [https://www.w3schools.com/jsref/dom\\_obj\\_htmlcollection.asp](https://www.w3schools.com/jsref/dom_obj_htmlcollection.asp) et éventuellement [https://www.w3schools.com/jsref/dom\\_obj\\_html\\_nodelist.asp](https://www.w3schools.com/jsref/dom_obj_html_nodelist.asp)

Pour finir cette étape, on ajoute un peu de style.

Insérer le lien CSS (vers le fichier style.css) dans index.html

Effectuez les ajouts/modifications dans articles.js pour visualiser le survol des articles. Dans la feuille de style, une classe CSS hover est présente. L'objectif est donc d'ajouter/retirer dynamiquement cette classe aux éléments article quand la souris survole l'élément (événements mouseout/mouseover). On devra utiliser la méthode du DOM `addEventListener()`. Là aussi, essayez d'utiliser une "Arrow Function".

<b>Institut Galilée</b> <b>Paris 13</b>	<b>LANGAGES WEB – JAVASCRIPT</b> Exercice Articles	Réf. : WEBC [0.9] Page : 1 - 2
--	---	-----------------------------------

## Etape 2

On veut ajouter une date à chacun des articles comme suit :

### Les composants d'un ordinateur - 2024/07/29

Un ordinateur est composé de divers composants essentiels, tels que le processeur, la mémoire vive, et le disque dur.

Mais pour l'instant, nos articles n'ont pas de date. Nous allons donc dater nos articles de manière aléatoire, en disant que les articles ont été écrit à **une date comprise entre maintenant et les 31 jours précédents**. Codez d'abord une fonction `dateFormatee(uneDate)` qui prend un objet `Date` en paramètre et retournera la date au format "aaaa/mm/jj". Testez votre fonction avec la date d'aujourd'hui.

Ensuite vous écrirez une fonction `dateAleatoire(nbJourMax = 31)` qui, en utilisant `dateFormatee()`, sera appelé pour modifier chaque article. N'oubliez pas de modifier également votre code pour afficher au bout de chaque titre, la date de l'article, comme l'exemple ci-dessus.

Pour les fonctions `dateFormatee()` et `dateAleatoire()` il faudra un peu se documenter sur la classe `Date`:

- Comment on crée la date correspondant à l'instant de la création.
- Comment on récupère juste l'année, juste le jour, juste le mois...
- Comment on crée un nombre aléatoire compris entre 0 et 30 inclus pour pouvoir retirer ce nombre de jours à la date d'aujourd'hui.
- Comment on change la date du jour d'une date donnée.
- Petite aide : Si on est le 22 du mois et que l'on tire le nombre 30, le mois est automatiquement décrémenté (`dateDu22.getDate() - 30`) passera donc au mois précédent et le jour sera ajusté en conséquence.
- Pour compléter avec un caractère 0 le jour ou le mois vous pouvez utiliser la méthode `padStart()`.
- Si besoin, pour convertir un nombre en chaîne on peut utiliser la fonction `String(nombre)`.

## Etape 3

À l'aide d'un bouton, nous souhaitons faire disparaître les anciens articles et ne laisser présent que les trois plus récents. Ce bouton servira aussi à les faire réapparaître. Plutôt que d'avoir à régénérer le DOM à chaque pression du bouton, il est plus pratique d'utiliser l'attribut `hidden` qui fait disparaître un élément.

Pour afficher et faire disparaître les éléments, une méthode consiste à créer un tableau contenant les indices des articles les plus récents. Ensuite, dans une fonction, il ne reste qu'à parcourir votre `blogcontainer`, en vérifiant pour chaque indice s'il est inclus dans votre tableau d'indices à conserver, et le rendre `hidden` le cas échéant. Et une autre

<b>Institut Galilée</b> <b>Paris 13</b>	<b>LANGAGES WEB – JAVASCRIPT</b> Exercice Articles	Réf. : WEBC [0.9] Page : 1 - 3
--	---	-----------------------------------

fonction, qui supprime cet attribut de tous les articles pour tout réafficher. N'oubliez pas d'ajouter un `addEventListener()` au bouton que vous ajouterez.

Fonctions à utiliser :

- `sort()` qui permet de trier un tableau en accord avec une fonction fournie (privilégiez la notation Arrow Function)
- `slice()` sur un tableau renvoie une portion de tableau
- `map()` qui permet de créer un nouveau tableau en accord avec une fonction fournie (privilégiez la notation Arrow Function)
- `includes()` vérifie la présence d'une valeur dans un tableau
- pour rappel la propriété "children" retourne la collection des éléments enfants d'un élément englobant (la section ici).

#### Etape 4

Modifier le texte du bouton en fonction de l'action à entreprendre (3 articles ou tout afficher). Créer un style du bouton supplémentaire (une classe "buttonALL" par exemple) pour prendre en charge le survol du bouton, des couleurs différentes en fonction de son état... Profitez-en pour tester la fonction `toggle()`.

#### Etape 5

Faire en sorte que le tableau initial d'articles soit stocké dans un fichier au format json appelé "data.json". A vous de construire ce fichier de données. Récupérer ce fichier en effectuant une requête HTTP utilisant la fonction `fetch()` (voir ci-après). Il vous faudra désormais "travailler en réseau" c'est-à-dire en utilisant un serveur web par exemple uwamp. Attention ! Il faudra aussi sans doute réorganiser un tout petit peu votre code JavaScript à cause de l'utilisation de la fonction **asynchrone** suivante !

```
async function getJSON(url) {
  const data = await fetch(url);
  return await data.json() ?? [];
}
```

#### Etape 6

Dernier test. Un script php "data.php" vous est fourni. Il fera office de mini-back end écrit en langage php. Il renvoi au format json la liste des articles. Sous l'environnement uwamp utilisez ce script pour initialiser votre tableau d'articles.