

TP4 - Introduction au JavaScript

M. Tellene

1 Mise en place et objectif du TP

L'objectif de ce TP est de manipuler le JavaScript afin de créer une page dynamique.

Téléchargez le fichier zip ici. Vous y trouverez un fichier HTML et des images.

Quand vous aurez fini le TP, vous déposerez vos fichiers dans votre dossier personnel.

Vos fichiers doivent **TOUS** être valides selon le validateur W3C¹.

2 Manipulations simples

Commencer par lier le fichier .html avec le fichier .js (il faudra créer le fichier .js). Une fois fait, ouvrez votre page web dans un navigateur puis appuyer sur F12 et sélectionnez l'onglet « Console ». Dans la console, tapez les instructions suivantes :

```
— 3+4  
— "salut" + "ça va?"  
— 3 == 3  
— 3 === 3  
— "3" == 3  
— "3" === 3
```

Vous pouvez remarquer que les derniers appels sont intrigants. Le JavaScript possède un opérateur de comparaison particulier, le `===`. Pour résumer : l'égalité faible (`==`) effectuera une conversion des deux éléments à comparer avant d'effectuer la comparaison. L'égalité stricte (`===`) effectuera la même comparaison mais sans conversion préalable (elle renverra toujours `false` si les types des deux valeurs comparées sont différents).

Avant de passer à l'étape suivante, il est à noter que la console « sera votre meilleure amie ». En effet, en plus de pouvoir faire des manipulations, cette dernière affiche les possibles erreurs de vos programmes mais fait également de l'affichage. Pour faire de l'affichage en JavaScript, nous utilisons l'instruction `console.log(machin_à_afficher)`

3 Le compteur

Nous allons commencer par créer un compteur. Ce compteur devra être placé dans la division (`div`) ayant l'id `exo1`. En utilisant du HTML et éventuellement du CSS, recréer ceci :

1. https://validator.w3.org/#validate_by_upload+with_options



Indication : on a utilisé un tableau (<table>) afin de positionner les éléments ainsi. On a également utilisé 3 boutons (chacun des boutons possède un id) et un paragraphe, possédant lui aussi un id

Le JavaScript

Cette partie est à faire dans votre fichier JS.

- ☐ Créer une variable `cpt`, initialisée à 0

```
let cpt = 0;
```

- ☐ Récupérer, dans une variable, le paragraphe contenant le nombre

```
let paragraphe = document.getElementById("...");  
/*les ... doivent être remplacés par l'id que vous  
avez donné au paragraphe*/
```

- ☐ Récupérer, dans une variable, le bouton « Augmenter »
- ☐ Mettre un écouteur d'événement sur la variable contenant le bouton. Le type de l'événement sera un clic et la fonction lancée sera `augmenter_cpt`. Cela doit être de la forme :

```
variable.addEventListener("evenement", fonction);
```

En remplaçant `variable` par le nom de la variable contenant le bouton, `evenement` par le type de l'événement à écouter et `fonction` par le nom de la fonction à exécuter.

- ☐ Créer une fonction `augmenter_cpt`

```
function ...(){  
    /*corps de la fonction*/  
}
```

En remplaçant les « ... » par le nom de la fonction

- ☐ Dans la fonction `augmenter` la variable `cpt` de 1 et remplacer le contenu du paragraphe par le compteur (variable `cpt`)
- ☐ Créer les écouteurs et les fonctions adéquates pour les deux autres boutons
- ☐ Faire en sorte que si le compteur est à 0 et que l'on clique sur « Diminuer », le compteur reste bloqué à 0

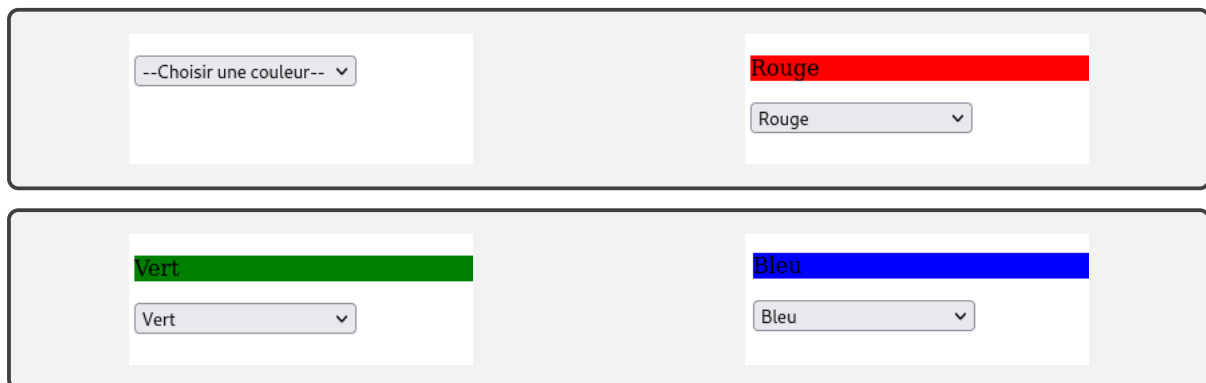
4 Changement de couleur

Créer sous la division `exo1`, une nouvelle division ayant pour id `exo2`. Pour séparer avec la `exo1`, vous pouvez appliquer à `exo2` le style `margin-top: 5%;`.

Dans cette nouvelle division créez un paragraphe vide. Sous ce paragraphe, créez un élément `select` ayant comme id la valeur *couleurs*. Vous trouverez des informations ici sur la balise `select`. Notre élément `select` contiendra 4 options.

- Une option – *Choisir une couleur* –. Cette option devra être présélectionnée lors du chargement de la page **et** devra réinitialiser le contenu et le style du paragraphe
- Une option *Rouge*
- Une option *Vert*
- Une option *Bleu*

Le but de cette partie sera de faire en sorte que, lorsqu'une couleur est sélectionnée, met à jour la paragraphe. Cette mise en forme fera en sorte que la couleur choisie soit écrite dans le paragraphe. De plus, la couleur de paragraphe devra être celle de la couleur choisie. Afin de gérer la mise à jour, vous devrez utiliser la fonction `setInterval()` de JavaScript. Voici le résultat attendu :



Indication : afin d'avoir un affichage presque immédiat, vous pouvez fixer l'intervalle à 50 ms.

5 Événements et images

Créer sous la division *exo2*, une nouvelle division ayant pour id *exo3*. Pour séparer avec la *exo2*, vous pouvez appliquer à *exo3* le style `margin-top: 5%;`. Commencer par recréer, en HTML et CSS la page suivante (*Aide : la taille de l'image est à 30%*) :

Exercice 3



Premier texte sans italique

5.1 Tâches à faire

- ☐ Donner l'id *paragraphe_exo3* au paragraphe (HTML)
- ☐ Récupérer dans une variable *paragraphe_exo3* le paragraphe de l'exercice (JS)
- ☐ Ajouter un événement sur ce paragraphe. L'événement sera un « clic » et devra faire en sorte que le texte soit formaté afin que contenu apparaisse en italique. Pour ce faire vous devrez utiliser la structure suivante (en remplaçant les « ... » et en ajoutant la ou les actions à réaliser) :

```
paragraphe_exo3.addEventListener(..., () => {  
    /* action(s) réalisée(s)  
});
```

- ☐ Lorsque le pointeur de la souris se déplace sur l'image, elle doit être remplacée par l'image *marseille.jpeg*. L'événement à capturer est de type *mouseover*. À vous de faire le nécessaire.
- ☐ Lorsque le pointeur de la souris est en dehors de l'image, l'image initiale doit être restaurée (*provence.jpg*). L'événement à capturer est de type *mouseout*. À vous de faire le nécessaire.

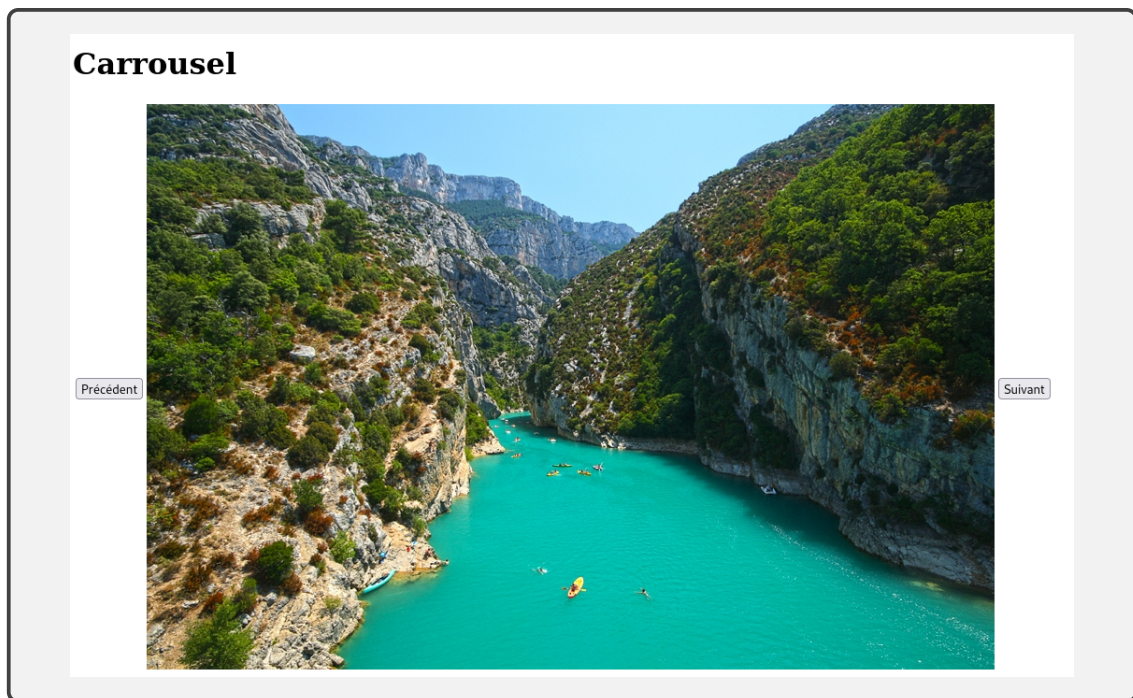
6 Carrousel d'image

Créer sous la division *exo3*, une nouvelle division ayant pour id *exo4*. Pour séparer avec la *exo3*, vous pouvez appliquer à *exo4* le style `margin-top: 5%;`.

6.1 Tâches à faire

- ☐ Créer une page Web (HTML) qui contient deux boutons « précédent » et « suivant ». Spécifier deux identifiants distincts pour ces boutons.
- ☐ Placer l'image entre les deux boutons. L'image qui s'affiche au chargement de la page est *carroussel1.jpg*. Donner un identifiant à l'image

- Modifier le CSS afin d'avoir un rendu proche de celui-ci :



- Créer une variable `tab`. Cette variable contiendra un tableau (Array) où chaque élément sera le chemin des images `carrousel1.jpg`, `carrousel2.jpg`, `carrousel3.jpg` et `carrousel4.jpg`
Aide pour les tableaux : Un tableau est, en programmation, une structure de données permettant de stocker des éléments. Tous les éléments d'un tableau doivent avoir le même type (number, string, ...). Voici quelques instructions utiles sur les tableaux. Plus d'aide [ici](#) :

```
let tab = []; /*création d'un tableau vide*/
let tab = [1,2,3]; /*création d'un tableau avec 3 nombres*/

let tab = [1,2,3]; /*création*/
tab.push(4); /*ajout d'une valeur*/
/*état du tableau : [1,2,3,4]*/

let tab = [1,2,3,4]; /*création*/
let x = tab[0]; /*récupération d'élément*/
/*x contient le nombre 1 (élément à l'indice 0 de tab)*/
```

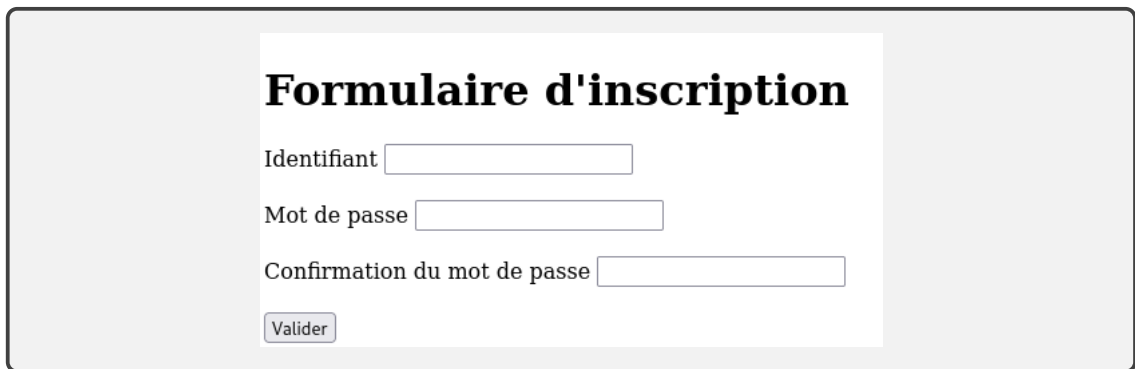
- Effectuer le défilement de l'image au clic sur les boutons « précédents » et « suivants ». À vous de faire le nécessaire.
- Effectuer le défilement automatique de l'image après 2 secondes, grâce aux fonctions `setInterval` et `clearInterval`. Documentez-vous au besoin sur l'utilisation de cette fonction. `setInterval` **devra** appeler une fonction `change_img`. Cette fonction s'occupera du défilement des images. **Attention dès que l'utilisateur clique sur un des deux boutons, le timer de 2 secondes doit être relancé.**

7 Formulaire d'authentification

Créer sous la division *exo4*, une nouvelle division ayant pour id *exo5*. Pour séparer avec la *exo4*, vous pouvez appliquer à *exo5* le style `margin-top: 5%;`.

7.1 Tâches à effectuer

- ☐ En utilisant du HTML seulement, recréer la page suivante :



Formulaire d'inscription

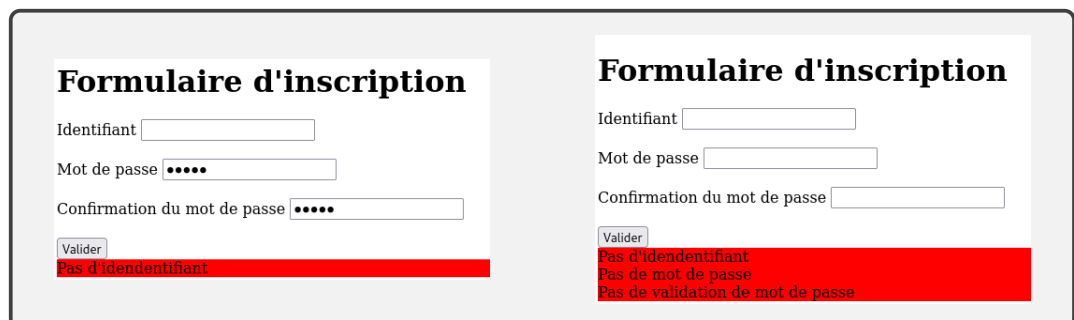
Identifiant

Mot de passe

Confirmation du mot de passe

On précise que les input mot de passe ont le type *password*. Concernant le formulaire vous ne mettez pas d'action, vous utiliserez la méthode post et vous ajouterez, dans la balise, l'attribut `onsubmit` et lui donnant la valeur `"return false"`.

- ☐ Ajouter un écouteur d'événement sur le bouton. Cet événement sera un « clic » et lancera la fonction `verification_formulaire`
- ☐ Créer une fonction `verification_formulaire`. Cette fonction devra faire les choses suivantes :
 - ☐ Récupérer toutes les valeurs entrées dans des variables (une valeur par variable)
 - ☐ Vérifier que l'utilisateur a entré une valeur pour tous les input, auquel cas, il faudra que votre page se comporte comme ceci (il faudra modifier le HTML) :



Formulaire d'inscription

Identifiant

Mot de passe

Confirmation du mot de passe

Pas d'identifiant

Formulaire d'inscription

Identifiant

Mot de passe

Confirmation du mot de passe

Pas d'identifiant
Pas de mot de passe
Pas de validation de mot de passe

Quand le formulaire est correctement rempli, aucun message d'erreur ne doit apparaître.

- ☐ La fonction `verification_formulaire` devra également vérifier que les mots de passe correspondent. Voici le résultat attendu :

Formulaire d'inscription

Identifiant

Mot de passe

Confirmation du mot de passe

Les mots de passe ne correspondent pas

Formulaire d'inscription

Identifiant

Mot de passe

Confirmation du mot de passe

Inscription réussite