

TP JavaScript : manipulation du DOM

- Utilisez **Visual Studio Code** pour réaliser les exercices.
 - Chaque exercice doit être réalisé dans un fichier JavaScript distinct, associé à une page HTML permettant son exécution.
 - Les fichiers doivent être nommés `exercice1.html` et `exercice1.js`, puis `exercice2.html` et `exercice2.js`, etc.
 - Chaque fichier HTML doit charger correctement le fichier JavaScript correspondant à l'aide de la balise `<script>`.
 - Contrairement au précédent TP JavaScript consacré à l'algorithmique, ce TP porte sur la manipulation du **DOM** et des éléments HTML.
 - L'interaction avec l'utilisateur doit principalement être réalisée à l'aide d'éléments HTML et d'événements JavaScript.
-

Avant de commencer, vérifiez que chaque exercice est réalisé dans son propre couple de fichiers :

```
tp-dom/  
|  
+-- exercice1.html  
+-- exercice1.js  
|  
+-- ...  
|  
+-- exercice6.html  
+-- exercice6.js
```

Exercice 1 : Diaporama d'images Nintendo

Téléchargez l'archive contenant les images Nintendo disponible à l'adresse suivante : <https://www.el-mouelhi.fr/images.zip>. Les images seront utilisées pour réaliser un petit diaporama entièrement contrôlé avec JavaScript.

Question 1 : changer une image

Créer une page HTML contenant :

- une première `<div>` contenant un bouton **Changer l'image**
- une seconde `<div>` contenant une image

L'image affichée initialement doit être celle de **Mario**.

Lorsque l'utilisateur clique sur le bouton, l'image de Mario doit être remplacée par celle de **Luigi**.

Le changement d'image doit être réalisé à l'aide de JavaScript.

Question 2 : faire défiler les images

Modifier le programme précédent afin que chaque clic sur le bouton affiche l'image suivante.

L'ordre de défilement doit être :

Mario → Luigi → Bowser → Peach → Toad → Toadette → Yoshi

Le défilement doit donc être **cyclique** : lorsque la dernière image est atteinte, le clic suivant doit revenir à la première image.

Question 3 : défilement automatique

Ajouter deux nouveaux boutons **Démarrer** et **Arrêter**.

- **Démarrer** doit lancer automatiquement le changement d'image toutes les **2 secondes**
- **Arrêter** doit interrompre le défilement automatique.

Indice : utiliser les fonctions JavaScript `setInterval()` et `clearInterval()`.

Exercice 2 : Manipulation des classes CSS

Créer une page contenant trois paragraphes et associer à chacun des paragraphes une classe CSS différente. Chacune de ces classes applique la couleur qu'elle porte sur le texte associé.

- **rouge** : texte rouge
- **bleu** : texte bleu
- **vert** : texte vert

Dans une `div`, ajouter deux boutons **Supprimer les couleurs** et **Changer les couleurs**.

Question 1 : supprimer les couleurs

Lorsque l'utilisateur clique sur le bouton **Supprimer les couleurs**, supprimer les classes de couleur associées aux trois paragraphes. Les paragraphes doivent alors retrouver leur apparence normale.

Question 2 : faire tourner les couleurs

Lorsque l'utilisateur clique sur le bouton **Changer les couleurs**, effectuer la rotation suivante :

Avant	Après
rouge	bleu
bleu	vert
vert	rouge

Chaque clic sur le bouton doit effectuer une nouvelle rotation.

Exercice 3 : Mise en forme d'un texte

Créer une page contenant un paragraphe présentant brièvement JavaScript.

Par exemple :

JavaScript est un langage de programmation principalement utilisé pour rendre les pages web interactives. Il permet notamment de modifier le contenu et la structure d'une page HTML grâce au DOM.

Ajouter quatre boutons :

- **Gras**;
- *Italique*;
- Souligné;
- Initialiser.

Question 1 : gras

Lorsque l'utilisateur clique sur le bouton **Gras**, le texte doit être affiché en gras.

Question 2 : italique

Lorsque l'utilisateur clique sur le bouton *Italique*, le texte doit être affiché en italique.

Question 3 : souligné

Lorsque l'utilisateur clique sur le bouton Souligné, le texte doit être souligné.

Les différentes mises en forme doivent pouvoir être combinées.

Question 4 : initialiser

Lorsque l'utilisateur clique sur le bouton **Initialiser**, toutes les mises en forme doivent être supprimées. Le texte doit alors retrouver son apparence initiale.

Exercice 4 : Mini-calculatrice

Créer une mini-calculatrice permettant d'effectuer des opérations sur deux nombres. La page doit contenir :

- un premier `<input type="number">`
- un deuxième `<input type="number">`
- un bouton `+`
- un bouton `-`
- un bouton `*`
- un bouton `/`.

L'utilisateur saisit deux nombres puis clique sur l'opération souhaitée.

Le résultat doit être affiché dans une boîte de dialogue `alert()`.

Exemple

- premier nombre : 12 ;
- deuxième nombre : 4 ;
- opération : multiplication.

Le programme doit afficher : $12 * 4 = 48$

Question bonus : Les boutons correspondant aux opérations doivent être désactivés tant que les deux champs ne sont pas correctement renseignés. Dès que les deux valeurs sont saisies, les boutons doivent être automatiquement réactivés.

Exercice 5 : Gestion de listes

Créer une interface permettant de gérer une liste de sports (avec `<select>` et `<option>`). La page doit contenir deux listes déroulantes :

- **Sports disponibles**
- **Sports préférés**

La première liste doit contenir les sports suivants :

- Football
- Handball
- Hockey
- Rugby
- Tennis
- CrossFit

Entre les deux listes, placer quatre boutons : `>`, `>>`, `<`, `<<`.

Question 1 : déplacer un sport vers la droite

Lorsque l'utilisateur sélectionne un sport dans la liste **Sports disponibles** et clique sur le bouton `>`, le sport sélectionné doit être déplacé dans la liste **Sports préférés**. Le sport doit disparaître de la première liste.

Question 2 : déplacer tous les sports vers la droite

Le bouton `>>` doit déplacer **tous les sports** de la liste **Sports disponibles** vers la liste **Sports préférés**.

Question 3 : déplacer un sport vers la gauche

Lorsque l'utilisateur sélectionne un sport dans la liste **Sports préférés** et clique sur le bouton `<`, le sport sélectionné doit être déplacé vers la liste **Sports disponibles**.

Question 4 : déplacer tous les sports vers la gauche

Le bouton `<<` doit déplacer **tous les sports** de la liste **Sports préférés** vers la liste **Sports disponibles**.

Exercice 6 : Création dynamique de paragraphes

Créer une interface permettant à l'utilisateur de saisir un texte et de l'ajouter dynamiquement à la page. L'interface doit contenir :

- un `<textarea>`
- trois boutons radio pour choisir la couleur
- trois cases à cocher pour choisir les mises en forme
- un bouton **Ajouter**

Les trois boutons radio doivent correspondre aux couleurs :

- Rouge
- Vert
- Bleu

Les trois cases à cocher doivent correspondre aux mises en forme :

- Gras
- Italique
- Souligné

Prévoir également une `<div>` destinée à recevoir les paragraphes créés dynamiquement.

Question 1 : ajouter un paragraphe

L'utilisateur :

1. saisit un texte dans le `textarea`
2. choisit éventuellement une couleur
3. sélectionne éventuellement une ou plusieurs mises en forme
4. clique sur le bouton **Ajouter**

Un nouveau paragraphe doit alors être créé dynamiquement et ajouté dans la `<div>` prévue à cet effet.

Chaque nouveau paragraphe doit être ajouté **sous le précédent**.

Par exemple :

Premier paragraphe

Deuxième paragraphe

Troisième paragraphe

Les propriétés choisies par l'utilisateur doivent être appliquées uniquement au paragraphe qui vient d'être créé.

Question 2 : réinitialiser le formulaire

Après l'ajout d'un paragraphe, le formulaire doit être automatiquement réinitialisé.

Il faut notamment :

- vider le `textarea`
- décocher **Gras**
- décocher **Italique**
- décocher **Souligné**
- réinitialiser le choix de couleur

Le formulaire doit donc être prêt pour la saisie d'un nouveau paragraphe.

Question 3 : supprimer un paragraphe

Chaque paragraphe créé doit être accompagné de son propre bouton **Supprimer**.

Lorsque l'utilisateur clique sur ce bouton :

- seul le paragraphe correspondant doit être supprimé
- les autres paragraphes doivent rester présents

Attention : il ne faut pas supprimer tous les paragraphes. Seul le paragraphe associé au bouton cliqué doit disparaître.