

TP HTML : de la structure au formulaire

- Téléchargez Visual Studio code via <https://code.visualstudio.com/download>
 - Lancez l'installation du fichier téléchargé
 - Démarrer Visual Studio Code
 - Consulter les raccourcis **Emmet** sur <https://docs.emmet.io/cheatsheet-a5.pdf> pour une génération de code plus rapide
-

Votre document devra être conforme au standard HTML et la balise `html` devra préciser la langue du document :

```
<!DOCTYPE html>
<html lang="fr">
```

À la fin, vous devrez impérativement vérifier votre document avec le validateur officiel du W3C (<https://validator.w3.org/>). L'objectif est d'obtenir un document sans erreur de validation.

Exercice 1 : Textes, titres, citations et formules

Générez la structure minimale d'une page HTML ayant le titre **Introduction au développement Web**.

La page devra contenir plusieurs niveaux de titres :

- un titre principal avec **h1** : Introduction à la programmation Web
- plusieurs titres de niveau 2 avec **h2**
 - ★ HTML : structurer le contenu
 - ★ CSS : mettre en forme le contenu
- au moins un titre **h3** dans chacune de ces sections

Rédigez au moins trois paragraphes présentant le rôle du HTML et du CSS dans le développement Web. Dans vos paragraphes, utilisez les éléments HTML appropriés pour :

- mettre en évidence une information importante
- mettre un terme en emphase
- représenter une information devenue obsolète ou supprimée
- représenter une information nouvellement ajoutée
- ajouter une citation courte intégrée dans un paragraphe
- définir un terme ou une notion du vocabulaire du Web
- indiquer le nom complet correspondant à une abréviation, par exemple VS Code
- indiquer un raccourci clavier utilisé dans Visual Studio Code
- présenter un extrait de code informatique sur plusieurs lignes en conservant son indentation
- représenter le résultat affiché suite à l'exécution de l'extrait de code précédent

Montrez que HTML permet de représenter les expressions mathématiques simples suivantes :

$$E = mc^2$$

$$a^2 + b^2 = c^2$$

ainsi qu'une expression plus complexe, nécessitant plusieurs éléments mathématiques, comme la formule donnant les solutions d'une équation du second degré

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Exercice 2 : Listes

Ajoutez une section intitulée **Les listes en HTML**.

Utilisez une liste ordonnée pour présenter le contenu suivant

Pour apprendre la programmation Web, on commence par :

1. HTML
2. CSS

Et ensuite, on continue avec

3. Bootstrap
4. JavaScript
5. jQuery

Représentez l'organisation suivante avec des listes imbriquées (non ordonnées) :

Frontend

- HTML

- CSS

- JavaScript

Backend

- PHP

- Java

- Python

Base de données

- MySQL

- PostgreSQL

- MongoDB

Pour ne pas afficher les puces, vous pouvez utiliser l'attribut `style="list-style-type:none"`

Exercice 3 : Images et liens

Dans cet exercice, vous allez créer une petite galerie présentant quatre technologies du développement Web :

- HTML
- CSS
- JavaScript
- PHP

L'objectif est de créer une page principale contenant des **miniatures cliquables**. Lorsqu'une miniature est sélectionnée, l'utilisateur doit être redirigé vers une nouvelle page HTML affichant l'image dans une taille plus importante.

Votre projet devra avoir une organisation similaire à :

```
TP-HTML/  
|  
+-- index.html  
|  
+-- web/  
    +-- html.html  
    +-- css.html  
    +-- javascript.html  
    +-- php.html  
|  
+-- images/  
    +-- html.jpg  
    +-- css.jpg  
    +-- javascript.jpg  
    +-- php.jpg
```

Dans `index.html`, ajoutez une section intitulée **Les technologies du développement Web**. Affichez les quatre images sous forme de **miniatures** :

- HTML
- CSS
- JavaScript
- PHP

Les images devront être suffisamment petites pour être présentées comme des miniatures. Chaque image devra être **cliquable**. Lorsque l'utilisateur clique sur :

- l'image HTML, il doit accéder à `html.html`
- l'image CSS, il doit accéder à `css.html`
- l'image JavaScript, il doit accéder à `javascript.html`
- l'image PHP, il doit accéder à `php.html`.

Créez les quatre pages HTML suivantes :

- `html.html`
- `css.html`
- `javascript.html`
- `php.html`

Chaque page devra afficher :

1. un titre indiquant la technologie présentée
2. l'image correspondante en taille plus importante
3. une courte description de la technologie
4. les liens de navigation demandés ci-dessous.

Au-dessus de l'image, ajoutez trois liens :

- **Image précédente**
- **Retour à l'accueil**
- **Image suivante**

Le fonctionnement devra être circulaire. Ainsi :

- depuis la page HTML, “Image suivante” mène vers CSS
- depuis CSS, “Image suivante” mène vers JavaScript
- depuis JavaScript, “Image suivante” mène vers PHP
- depuis PHP, “Image suivante” revient vers HTML

La navigation précédente devra fonctionner dans l'autre sens :

- depuis HTML, “Image précédente” mène vers PHP
- depuis PHP, “Image précédente” mène vers JavaScript
- depuis JavaScript, “Image précédente” mène vers CSS
- depuis CSS, “Image précédente” mène vers HTML

Le lien “Retour à l'accueil” devra permettre de revenir à `index.html`.

Vous devrez respecter les contraintes suivantes :

- utiliser des chemins relatifs correctement construits
- chaque image devra posséder un attribut `alt` pertinent
- le code HTML devra être correctement indenté et structuré

Exercice 4 : Tableaux

On souhaite représenter l'emploi du temps simplifié suivant :

Jour	08h–10h	10h–12h	13h–15h	15h–17h
Lundi	HTML	HTML	CSS	JavaScript
Mardi	Python	Python	Déjeuner	JavaScript
Mercredi	Projet Web	Projet Web	Projet Web	JavaScript
Jeudi	SQL	SQL	CSS	Projet Web
Vendredi	Projet Web	Projet Web	CSS	CSS

On souhaite maintenant obtenir une représentation HTML dans laquelle les cellules identiques sont fusionnées.

Vous devrez notamment représenter les situations suivantes :

- les deux heures de HTML du lundi sont regroupées dans une seule cellule
- les deux heures de Python du mardi sont regroupées dans une seule cellule
- le projet Web du mercredi occupe plusieurs créneaux horaires consécutifs
- le projet Web du jeudi après-midi occupe plusieurs créneaux
- certaines cellules peuvent s'étendre sur plusieurs lignes

Pour réaliser ces fusions, vous utiliserez les attributs HTML :

- **colspan** pour fusionner plusieurs **colonnes**
- **rowspan** pour fusionner plusieurs **lignes**

Le tableau devra comporter :

- une légende avec `<caption>`
- un en-tête avec `<thead>`
- un corps avec `<tbody>`
- des cellules d'en-tête avec `<th>`
- des cellules de données avec `<td>`
- au moins une utilisation de **colspan**
- au moins une utilisation de **rowspan**

Les cellules fusionnées devront être réalisées avec les attributs HTML appropriés et non avec du CSS.

Exercice 5 : Formulaire d'inscription

Créez une section intitulée **Inscription à une formation Web** et qui contient un formulaire permettant à un étudiant de saisir :

Informations personnelles

- nom
- prénom
- adresse e-mail
- date de naissance
- téléphone

Permettez à l'utilisateur de sélectionner son niveau :

- débutant
- intermédiaire
- avancé

Permettez également de sélectionner plusieurs technologies :

- HTML/CSS
- JavaScript
- PHP
- Python

Ajoutez également :

- une liste déroulante permettant de choisir une formation
- une zone de texte pour un commentaire
- une case à cocher permettant d'accepter les conditions d'inscription.

Ajoutez deux boutons :

- envoyer
- réinitialiser

Le formulaire devra respecter les bonnes pratiques suivantes :

- chaque champ devra posséder un **label** correctement associé
- les champs obligatoires devront être identifiés
- les types HTML5 appropriés devront être utilisés
- les champs devront être regroupés logiquement lorsque cela est pertinent