

MR ROLIN

[WEB] Html & CSS: Decouverte

2025

Introduction

Ce document propose une série d'exercices progressifs pour découvrir HTML et CSS. L'objectif est de partir d'une simple page de texte pour arriver à une page structurée avec un fichier CSS externe.

Étape 1 – Première page HTML (structure + texte)

Objectifs

- Comprendre la structure minimale d'une page HTML.
- Utiliser les balises de base pour le texte.
- Enregistrer et afficher un fichier .html dans le navigateur.

Consignes

1. Crée un dossier nommé projet_html.
2. Dans ce dossier, crée un fichier index.html.
3. Tape le code suivant, puis adapte le contenu à ta guise :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Ma première page</title>
  </head>
  <body>
    <h1>Ma première page HTML</h1>

    <p>Ceci est mon tout premier paragraphe en HTML.</p>

    <h2>Une deuxième section</h2>
    <p>Encore un paragraphe pour tester l'affichage.</p>

    <hr>

    <p>Ligne du bas<br>Avec un retour à la ligne.</p>
  </body>
</html>
```

4. Enregistre le fichier, puis ouvre-le dans le navigateur.
5. Modifie le contenu :
 - change le texte du <title>
 - change le texte des titres
 - ajoute un ou deux nouveaux paragraphes.

Mémo

- Structure
 - <!DOCTYPE html> : indique le type de document (HTML5).
 - <html> : élément racine de la page.
 - lang="fr" : langue principale (français).
 - <head> : informations techniques (titre, encodage, liens CSS...).
 - <meta charset="UTF-8"> : encodage des caractères (pour les accents).
 - <title> : titre affiché dans l'onglet du navigateur.
 - <body> : tout le contenu visible.
- Texte

- `<h1>` : titre principal (en général une seule fois).
- `<h2>`, `<h3>`... : sous-titres.
- `<p>` : paragraphe.
- `
` : retour à la ligne.
- `<hr>` : ligne horizontale.
- `` : texte important (souvent gras).
- `` : texte mis en valeur (souvent italique).

Étape 2 – Liens et images

Objectifs

- Ajouter des liens (`<a>`).
- Insérer une image (``).
- Comprendre les attributs `href`, `src`, `alt`.

Consignes

Dans le même fichier `index.html` :

1. Sous le contenu existant, ajoute une section « Liens utiles » :

```
<h2>Liens utiles</h2>

<p>
  Visite mon site préféré :
  <a href="https://www.example.com">Mon site préféré</a>
</p>

<p>
  Va voir le site de ma commune :
  <a href="https://www.exemple-commune.be">Site de la commune</a>
</p>
```

2. Crée un sous-dossier `images` dans `projet_html`.
3. Place une image dans ce dossier (par ex. `photo.jpg`).
4. Ajoute l'image dans la page :

```
<h2>Une image</h2>


```

5. Teste :
 - Clique sur les liens.
 - Renomme l'image ou modifie son chemin pour vérifier le rôle du texte `alt`.

Memo

Lien hypertexte :

- `<a>` : balise de lien.
- `href="..."` : adresse de destination.
- `target="_blank"` (optionnel) : ouvre dans un nouvel onglet.

Image:

- `` : insère une image.
- `src` : chemin vers le fichier image.
- `alt` : texte alternatif (affiché ou lu si l'image ne charge pas).

Bonnes pratiques :

- Utiliser un dossier `images/`.
- Donner des noms de fichiers clairs (`tour-eiffel.jpg` plutôt que `img001.jpg`).

Étape 3 – Présentation avec listes

Objectifs

- Utiliser les listes :
 - liste à puces ()
 - liste numérotée ()
- Structurer une petite fiche de présentation.

Consignes

Dans index.html, sous la partie « Liens », ajoute :

```
<h2>À propos de moi</h2>

<h3>Mes hobbies</h3>
<ul>
  <li>Jeux vidéo</li>
  <li>Lecture</li>
  <li>Sport</li>
</ul>

<h3>Ma journée type</h3>
<ol>
  <li>Je me lève</li>
  <li>Je vais à l'école</li>
  <li>Je fais mes devoirs</li>
  <li>Je me détends</li>
</ol>

<h3>Contact</h3>
<p>Tu peux me contacter via l'école.</p>
```

Adapte les contenus avec tes propres goûts et ta propre journée.

Memo

- : liste à puces
- : liste numérotée
- : élément de liste

Étape 4 – Structure de page (header/nav/main/footer) + tableau

Objectifs

- Utiliser des balises sémantiques modernes : <header>, <nav>, <main>, <section>, <footer>.
- Créer un tableau simple (<table>, <tr>, <th>, <td>).

Consignes

1. Réorganise le <body> de la page comme ceci :

```
<body>
  <header>
    <h1>Mon site personnel</h1>
  </header>

  <nav>
    <p>
      <a href="#accueil">Accueil</a> |
```

```

        <a href="#hobbies">Hobbies</a> |
        <a href="#contact">Contact</a> |
        <a href="#planning">Planning</a>
    </p>
</nav>

<main>
    <!-- Sections ici -->
</main>

<footer>
    <p>&copy; 2025 - Mon nom</p>
</footer>
</body>

```

2. À l'intérieur de <main>, crée des sections avec des identifiants :

```

<main>
    <section id="accueil">
        <h2>Accueil</h2>
        <p>Texte d'introduction...</p>
    </section>

    <section id="hobbies">
        <h2>Mes hobbies</h2>
        <!-- Liste de hobbies -->
    </section>

    <section id="contact">
        <h2>Contact</h2>
        <p>Tu peux me contacter via l'école.</p>
    </section>

    <section id="planning">
        <h2>Mon planning de la semaine</h2>
        <!-- Tableau ici -->
    </section>
</main>

```

3. Dans la section planning, ajoute un tableau :

```

<table>
    <caption>Emploi du temps simplifié</caption>
    <tr>
        <th>Jour</th>
        <th>Matin</th>
        <th>Après-midi</th>
    </tr>
    <tr>
        <td>Lundi</td>
        <td>Cours</td>
        <td>Cours</td>
    </tr>
    <tr>
        <td>Mardi</td>

```

```
<td>Cours</td>
<td>Sport</td>
</tr>
</table>
```

4. Vérifie que les liens du menu amènent bien aux bonnes sections.

Memo

Balises sémantiques :

- <header> : en-tête de page (logo, titre, menu).
- <nav> : barre de navigation (liens importants).
- <main> : contenu principal (une seule fois par page).
- <section> : section de contenu regroupé.
- <article> : contenu autonome (article de blog...).
- <aside> : contenu secondaire (sidebar, infos complémentaires).
- <footer> : pied de page (contact, mentions...).

Tableau :

- <table> : conteneur du tableau.
- <tr> : ligne.
- <th> : cellule d'en-tête (titre de colonne/ligne).
- <td> : cellule de données.
- <caption> : titre du tableau.

Étape 5 – Découverte du CSS (style interne)

Objectifs

- Comprendre la notion de règle CSS.
- Utiliser un style interne via la balise <style>.
- Modifier couleur, police, alignement de base.

Consignes

1. Dans le <head> de index.html, ajoute un bloc <style> :

```
<head>
<meta charset="UTF-8">
<title>Mon site personnel</title>

<style>
  body {
    font-family: Arial, sans-serif;
    line-height: 1.5;
  }

  h1, h2 {
    color: darkblue;
    text-align: center;
  }

  p {
    color: #333333;
  }
</style>
</head>
```

2. Recharge la page et observe les changements.

3. Modifie quelques valeurs (couleurs, polices, tailles) et regarde l'effet.

Memo

Structure d'une règle CSS :

```
sélecteur {  
  propriété: valeur;  
  autre-propriété: autre-valeur;  
}
```

Sélecteurs courants :

- Élément : p, h1, section...
- Classe : .important → <p class="important">.
- ID : #contact → <section id="contact">.

Propriétés de texte fréquentes :

- color : couleur du texte.
- font-family : police.
- font-size : taille du texte.
- font-weight : normal, bold.
- font-style : normal, italic.
- text-align : left, center, right, justify.
- text-decoration : none, underline...

Étape 6 – Le modèle de boîte (box model)

Objectifs

- Comprendre qu'un élément est une « boîte ».
- Utiliser width, margin, padding, border, background-color.

Consignes

1. Ajoute la classe bloc à tes sections principales :

```
<section id="accueil" class="bloc">...</section>  
<section id="hobbies" class="bloc">...</section>  
<section id="contact" class="bloc">...</section>  
<section id="planning" class="bloc">...</section>
```

2. Dans le bloc <style> (ou plus tard dans style.css), ajoute :

```
.bloc {  
  width: 80%;  
  margin: 20px auto;  
  padding: 15px;  
  border: 1px solid #cccccc;  
  border-radius: 8px;  
  background-color: #f9f9f9;  
}
```

3. Observe le résultat (sections centrées, encadrées, plus lisibles).

Memo

Chaque élément est composé de :

- contenu (texte, image)
- padding : espace intérieur
- border : bordure
- margin : espace extérieur

Propriétés importantes :

- width, height : taille du contenu.
- padding : espace intérieur.
- margin : espace extérieur.
- border : bordure.
- border-radius : coins arrondis.
- background-color : couleur de fond.

Étape 7 – Fichier CSS externe

Objectifs

- Séparer contenu (HTML) et présentation (CSS).
- Utiliser un fichier CSS externe lié à la page.

Consignes

1. Crée un dossier styles dans projet_html.
2. Crée le fichier styles/style.css.
3. Coupe tout le contenu du bloc `<style> ... </style>` et colle-le dans style.css (sans les balises `<style>`).
4. Dans le `<head>`, remplace le bloc `<style>` par :

```
<head>
  <meta charset="UTF-8">
  <title>Mon site personnel</title>
  <link href="styles/style.css" rel="stylesheet">
</head>
```

5. Vérifie que le style est toujours appliqué.
6. Dans style.css, ajoute par exemple :

```
nav {
  text-align: center;
  margin-bottom: 20px;
}

nav a {
  margin: 0 10px;
  text-decoration: none;
  color: darkblue;
}

nav a:hover {
  text-decoration: underline;
}
```

Memo

Lien vers la feuille de style :

```
<link href="styles/style.css" rel="stylesheet">
```

Avantages :

- HTML plus lisible.
- Même fichier CSS réutilisable sur plusieurs pages.
- Mise à jour du design en un seul endroit.

Étape 8 – Formulaire de contact et fin du mini-site

Objectifs

- Créer un formulaire simple.
- Finaliser le mini-site avec une mise en page cohérente.
- Améliorer l’affichage des listes, images, liens.

Consignes

1. Dans la section contact, ajoute un formulaire :

```
<section id="contact" class="bloc">
  <h2>Contact</h2>

  <form>
    <p>
      <label for="nom">Nom :</label><br>
      <input type="text" id="nom" name="nom">
    </p>

    <p>
      <label for="email">E-mail :</label><br>
      <input type="email" id="email" name="email">
    </p>

    <p>
      <label for="message">Message :</label><br>
      <textarea id="message" name="message" rows="4"
cols="40"></textarea>
    </p>

    <p>
      <button type="submit">Envoyer</button>
    </p>
  </form>
</section>
```

2. Dans styles/style.css, ajoute :

```
form {
  max-width: 400px;
  margin: 0 auto;
}

input, textarea {
  width: 100%;
  box-sizing: border-box;
}
```



```
button {
  padding: 8px 16px;
  background-color: darkblue;
  color: white;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

button:hover {
  background-color: #003366;
}
```

3. Améliore quelques éléments :

```
ul, ol {
  margin-left: 20px;
}

li {
  margin-bottom: 4px;
}

img {
  max-width: 100%;
  height: auto;
}
```

4. Vérifie que :

- la page est lisible et cohérente
- la navigation fonctionne
- le formulaire s'affiche correctement

Memo

Formulaire :

- `<form>` : conteneur.
- `<label>` : texte associé à un champ.
- `<input>` : champ (texte, email...).
- `<textarea>` : zone de texte multilignes.
- `<button>` : bouton d'action.

Propriétés CSS utiles en plus :

- `list-style-type: none;` → enlever les puces d'une liste.
- `a:hover { ... }` → effet au survol des liens.
- `img { max-width: 100%; height: auto; }` → image responsive.