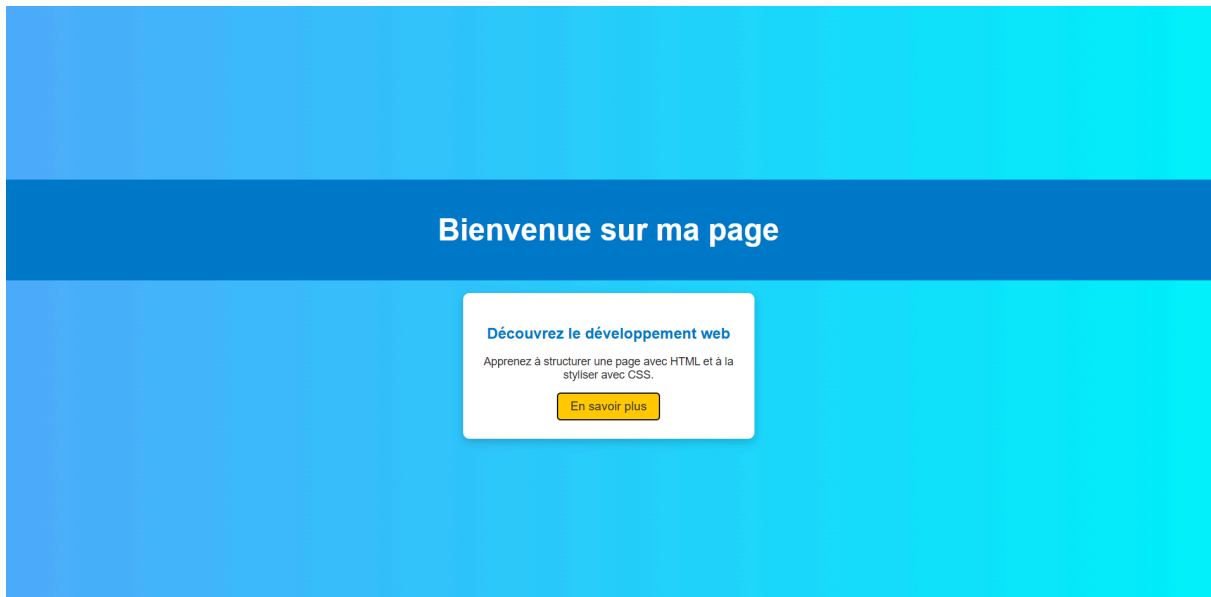


Activité HTML Penney

I. Introduction : Reproduire une Mise en Page Web

Objectif

A partir du code HTML et CSS donné produisez cette page :



Le code HTML :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <title>Ma Première Page Web</title>
  <link rel="stylesheet" href="src/style.css">
</head>
<body>
  <header>
    <h1>Bienvenue sur ma page</h1>
  </header>
  <section>
    <p>Ceci est un texte d'exemple pour s'entraîner.</p>
    <button>Découvrir</button>
```

```
    </section>
</body>
</html>
```

Le code CSS :

```
body {
  font-family: Arial, sans-serif;
  text-align: center;
  background-color: #f4f4f4;
}

header {
  background: #007acc;
  color: white;
  padding: 20px;
}

section {
  margin: 20px;
  padding: 20px;
  background: white;
  box-shadow: 2px 2px 10px rgba(0, 0, 0, 0.1);
}

button {
  padding: 10px;
  background: #ffcc00;
  border: none;
  font-size: 1.2em;
  cursor: pointer;
}

button:hover {
  background: #ff9900;
}
```

FICHE MEMO HTML/CSS

Principales balises HTML expliquées

📌 Structure de base

- `<html>` : Définit le début du document HTML.
- `<head>` : Contient les informations invisibles (titre, styles, métadonnées).
- `<body>` : Contient tout ce qui est visible sur la page.

📌 Titres et texte

- `<h1>` à `<h6>` : Titres (du plus grand `<h1>` au plus petit `<h6>`).
- `<p>` : Paragraphe de texte.
- `` : Permet de styliser une partie d'un texte sans sauter de ligne.

📌 Liens et images

- `` : Lien hypertexte vers une autre page.
- `` : Affiche une image.

📌 Listes

- `` : Liste non ordonnée (avec des puces).
- `` : Liste ordonnée (avec numérotation).
- `` : Élément d'une liste.

📌 Mise en page et structuration

- `<div>` : Bloc générique (utile pour structurer la page).
- `<section>` : Regroupe un contenu thématique.
- `<header>`, `<footer>` : Partie haute et basse d'une page.
- `<nav>` : Contient les liens de navigation.

📌 Formulaires et interactions

- `<input>` : Champ de saisie utilisateur.
- `<button>` : Bouton interactif.

Modifier le style avec CSS

♦ Changer la couleur de fond

```
body {  
  background-color: lightblue;  
}
```

🎨 Exemple : `background-color: #ffcc00;` (jaune) / `background: linear-gradient(to right, #4facfe, #00f2fe);` (dégradé)

♦ Changer la police du texte

```
body {  
  font-family: 'Arial', sans-serif;  
}
```

📌 Essayez : `font-family: 'Courier New', monospace;`

♦ Modifier la taille du texte

```
h1 {  
  font-size: 2em;  
}
```

📌 Essayez : `font-size: 1.5em;` (réduit) ou `font-size: 3em;` (agrandi)

♦ Styliser un bouton

```
button {  
  background: #ffcc00;  
  padding: 10px 20px;  
  border: none;  
  font-size: 1.2em;  
  cursor: pointer;  
  border-radius: 5px;  
  transition: background 0.3s ease-in-out;  
}  
button:hover {  
  background: #ff9900;  
  color: white;  
}
```

📌 Explication :

- `background` : Définit la couleur du bouton.
 - `padding` : Ajoute de l'espace autour du texte.
 - `border-radius` : Arrondit les coins du bouton.
 - `cursor: pointer` : Change le curseur en main lorsqu'on survole le bouton.
 - `transition` : Crée un effet de changement de couleur fluide.
-

♦ Centrer un élément

```
.container {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  height: 100vh;  
}
```

📌 Parfait pour centrer du texte ou une image !

♦ Ajouter une ombre

```
.box {  
  box-shadow: 0px 5px 15px rgba(0, 0, 0, 0.2);  
}
```

📌 Effet d'élévation sur un bloc ou un bouton.

♦ Modifier les marges et les espacements

```
p {  
  margin: 20px;  
  padding: 10px;  
}
```

📌 *margin* : espace autour / *padding* : espace à l'intérieur !

CA BLOQUE ?

Si vous rencontrez un problème n'hésitez pas à utiliser ChatGPT pour vous aider à trouver la solution.

Exemple :

Je n'arrive pas à réaliser les coins du bouton !

1 - J'envoie à ChatGPT ma page HTML et CSS avec les informations sur l'organisation de mes fichiers :

Je vais t'envoyer ma page web d'abord le fichier HTML puis CSS. Mon fichier CSS se trouve dans un dossier nommé src.

Envoyer HTML

Envoyer CSS

2 - Je lui explique ce que je veux réaliser simplement et clairement :

Peux-tu m'expliquer étape par étape comment rendre les coins de mon bouton arrondis ?

IV. Application : Théorème de Penney

A) Installation des Prérequis

1. Création du Projet

- Créez un dossier pour votre projet.
- Créez un fichier `index.html` et un dossier `src`.

2. Organisation des Fichiers

- Placez dans le dossier `src` :
 - `style.css`
 - `script.js` (fichier JavaScript à ne pas modifier)

3. Exemple de Code pour `index.html`

```
<!DOCTYPE html>

<html lang="fr">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Simulation du Paradoxe de Penney</title>

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

</head>

<body>

  <header>

    <h1>Simulation du Paradoxe de Penney</h1>

  </header>

  <main>

    <!-- Section pour entrer les séquences et démarrer la simulation -->

    <section id="inputs">

      <p>Entrez deux séquences de 3 lettres (P ou F) :</p>

      <input type="text" id="seq1" placeholder="Ex : PFF">

      <input type="text" id="seq2" placeholder="Ex : FFP">

      <button id="start">Lancer la simulation</button>

    </section>

  </main>

</body>

</html>
```

```

</section>

<!-- Section pour afficher l'animation et le résultat -->

<section id="display">

    <div id="coin-container"></div>

    <div id="result"></div>

</section>

<!-- Section bonus pour simulations multiples et historique -->

<section id="bonus">

    <button id="simulateMultiple">Simuler plusieurs fois</button>

    <button id="resetHistory">Réinitialiser l'historique</button>

    <h2>Historique des simulations</h2>

    <ul id="history"></ul>

</section>

</main>

<script src="src/script.js"></script>

</body>

</html>

```

B) Personnalisation de la Simulation

Objectifs

- Comprendre la structure HTML existante.
- Modifier et enrichir le HTML pour rendre la page claire et agréable.
- Personnaliser le style avec CSS pour obtenir un rendu esthétique et moderne.
- Intégrer des images (pour l'animation et les résultats) et ajouter des effets visuels.

Matériel Fourni

- **index.html** : Structure de base de la page.
- **style.css** : Feuille de style à modifier.
- **script.js** : Code JavaScript (à ne pas modifier).
- **Images** (à trouver sur internet et à placer dans votre projet) :
 - **coin_flip.gif** : Animation de lancer de pièce.
 - **heads.png** et **tails.png** : Images représentant respectivement « Pile » et « Face ».

Tâches à Réaliser

1. **Analyser et Commenter le HTML**
 - Identifier les balises importantes : `<header>`, `<section>`, `<main>`, etc.
 - Repérer les éléments clés avec leurs identifiants (ex. "seq1", "start", "coin-container").
2. **Modifier et Enrichir le HTML**
 - Ajouter un message d'accueil ou une description de l'exercice.
 - Intégrer une image de logo ou un autre élément visuel via la balise ``.
3. **Personnaliser le CSS**
 - **Fond de Page et Couleurs**
Remplacez le fond uni par un dégradé ou une image de fond.

Exemple :

```
body {  
  
    background: linear-gradient(to right, #4facfe, #00f2fe);  
  
    color: #333;  
  
    font-family: 'Arial', sans-serif;  
  
    margin: 0;  
  
    padding: 0;  
  
}
```

- **En-tête et Titres**

Appliquez un style distinct au `<header>`.

Exemple :

```
header {  
  
    background-color: #007acc;
```



```

color: white;

padding: 20px;

width: 100%;

text-align: center;
}

```

- **Formulaire d'Entrée**

Stylisez les `<input>` et `<button>` pour une meilleure lisibilité.

Exemple :

```

button {

    background: #ffcc00;

    border: none;

    padding: 10px 20px;

    font-size: 1.2em;

    border-radius: 5px;

    cursor: pointer;

    transition: background 0.3s ease;

}

button:hover {

    background: #ff9900;

    color: white;

}

```

- **Animation et Affichage des Résultats**

Ajoutez des effets pour le conteneur d'animation et la zone de résultat.

Indice : Vous pouvez utiliser `box-shadow` pour créer un effet d'élévation, par exemple :

```
#coin-container, #result {

    margin: 20px;

    padding: 20px;

    background: rgba(255, 255, 255, 0.8);

    border-radius: 10px;

    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);

}
```

- **Section Bonus (Historique)**

Stylisez la liste et la zone d'affichage des statistiques.

```
#history {

    list-style: none;

    padding: 0;

}

#history li {

    background: #fff;

    margin: 5px;

    padding: 10px;

    border-radius: 5px;

    box-shadow: 0 2px 5px rgba(0,0,0,0.1);

}

#average {

    margin-top: 10px;

    font-size: 1.2em;

    font-weight: bold;

}
```

V. Organisation & Conseils

1. Analysez le Code HTML

Commencez par comprendre la structure et identifiez les éléments à personnaliser.

2. Travail Progressif

- Commencez par la mise en page de base.
- Ajoutez progressivement des effets visuels (dégradés, ombres, transitions).

3. Testez Régulièrement

Utilisez l'outil « Inspecter l'élément » ou touche F12 de votre navigateur pour visualiser en temps réel les changements et voir les erreurs dans la console.

4. Utilisez la Fiche Mémo

Revenez sur la fiche mémo HTML/CSS pour réviser les concepts essentiels.