

Projet du vendredi Flexbox

Niveau 1 : Facile (1 heure)

Objectif : Comprendre les bases de Flexbox en créant une structure de page simple.

Étapes guidées :

1. Création de la structure HTML :

- Créer un nouveau fichier nommé `index.html`.
- Ajouter les balises de base (`<!DOCTYPE html>`, `<html>`, `<head>`, `<body>`).
- Dans le `<head>`, ajouter :
 - `<title>` pour le titre de la page (par exemple, "Introduction à Flexbox").
 - `<meta charset="UTF-8">` pour l'encodage.
 - `<meta name="viewport" content="width=device-width, initial-scale=1.0">` pour le responsive design.
- Dans le `<body>`, créer les éléments suivants :
 - Une balise `<header>` avec le texte "Exercice Flexbox".
 - Une balise `<section>` avec cinq `<div>` à l'intérieur, nommés `.box1`, `.box2`, `.box3`, `.box4`, et `.box5`. Par exemple : `<div class="box1">Boîte 1</div>`.
 - Une balise `<footer>` avec le texte "Pied de page".

2. Application du style de base en CSS :

- Créer un fichier `style.css` et le lier au fichier HTML en ajoutant `<link rel="stylesheet" href="style.css">` dans la balise `<head>`.
- Ajouter les styles suivants dans `style.css` :
 - la section fera au moins 500px en hauteur et en 100 % de la largeur de l'écran.

- Définir une largeur et une hauteur différente pour chaque `.box` avec une couleur de fond différente (par exemple, `.box1 { background-color: red; }`). La hauteur sera comprise entre 20px et 70px et la largeur entre 50px et 100px
- Ajouter une marge de 10px autour de chaque boîte.

3. Découverte de Flexbox :

- Ajouter `display: flex` à la balise `<section>` dans le fichier CSS pour activer Flexbox.
- Modifier la propriété `justify-content` avec différentes valeurs (`flex-start`, `center`, `space-between`, `space-around`) et observer comment les boîtes se placent horizontalement.
- Modifier la propriété `align-items` avec différentes valeurs (`stretch`, `center`, `flex-end`) pour voir l'effet sur l'alignement vertical des boîtes.
- Modifier `flex-direction` à `column` et observer l'impact sur l'organisation des éléments.

Quiz écrit et pratique :

• Questions théoriques :

1. Que fait la propriété `justify-content` dans Flexbox ?
2. Comment la valeur `flex-direction: column` affecte-t-elle la disposition des éléments ?

• Exercices pratiques :

1. Modifier la valeur de `justify-content` dans le fichier CSS et observer l'effet sur la disposition des boîtes. Notez vos observations.
2. Essayez d'aligner les boîtes verticalement au centre en utilisant `align-items`. Quelles valeurs avez-vous testées ?

Partie 2 : Manipulation avancée des Flexbox (2 heures)

Objectifs : Approfondir les connaissances sur Flexbox avec `flex-grow`, `flex-shrink`, et `flex-basis`.

Étapes guidées :

1. Étendre la mise en page :

- Ajouter trois nouveaux `<div>` dans la section avec les classes `.box6`, `.box7`, et `.box8` et un texte comme "Boîte 6".
- Chaque boîte doit avoir un style de base différent en termes de couleur de fond.

2. Utilisation des propriétés avancées :

- Modifier les valeurs de `flex-grow` pour plusieurs boîtes afin de voir comment elles s'agrandissent selon l'espace disponible.
 - Exemple : définir `flex-grow: 2` pour `.box1` et `flex-grow: 1` pour `.box2`.
- Tester `flex-shrink` en attribuant des valeurs différentes aux boîtes.
 - Exemple : `flex-shrink: 0` pour `.box3` et `flex-shrink: 1` pour les autres.
- Configurer `flex-basis` pour définir des tailles spécifiques (ex. `flex-basis: 150px` pour `.box4` et `flex-basis: 100px` pour `.box5`).

Quiz écrit et pratique :

• Questions théoriques :

1. Quelle est la différence entre `flex-grow` et `flex-shrink` ?
2. Comment la propriété `flex-basis` influence-t-elle la taille d'un élément ?

• Exercices pratiques :

1. Appliquez `flex-grow` à plusieurs boîtes et notez comment elles changent de taille en fonction de l'espace disponible.
2. Testez différentes configurations de `flex-shrink` et décrivez les résultats obtenus avec des éléments de tailles différentes.
3. Pour aller plus loin, utilisez flex sur le body, pour que la section prennent toute la place disponible avec le header en haut et le footer en bas.

Partie 3 : Projet pratique - Mise en page d'une page portfolio (1h30 à 2 heures)

1. Instructions détaillées pour la création d'une page complète :

- **Structure de la page :**

- **En-tête** : Contiendra le logo et les liens de navigation. Le document indiquera comment structurer le code avec les balises appropriées (par exemple, `<header>`, `<nav>`, ``, ``, `<a>`).
- **Section de présentation** : Inclura un texte de présentation et éventuellement une image. Les instructions détailleront l'utilisation des balises comme `<section>`, `<h1>`, `<p>`, etc.
- **Galerie d'images** : Utilisera plusieurs éléments `<div>` ou `<figure>` pour afficher les images. Il sera précisé comment structurer les images et leurs légendes.
- **Pied de page** : Contiendra des informations de contact ou des liens supplémentaires. Les balises suggérées seront `<footer>`, ``, et ``.

2. Consignes pour utiliser Flexbox afin de réaliser une disposition responsive :

- **Flexbox pour l'en-tête** : Les instructions montreront comment utiliser `display: flex` pour aligner les éléments de l'en-tête sur une seule ligne, avec `justify-content: space-between` pour les espacer.
- **Disposition de la galerie** : Utilisation de `flex-wrap` pour permettre aux images de se placer sur plusieurs lignes, tout en gardant un alignement harmonieux.
- **Alignement vertical dans la section de présentation** : Utilisation des propriétés `align-items` et `justify-content` pour centrer le contenu.