

# Développement Web: Introduction à HTML/CSS

Abdelraouf Hecham - IUT de Montpellier - AS

hecham@lirmm.fr ([www.lirmm.fr/~hecham](http://www.lirmm.fr/~hecham))

8 Janvier 2018

# Outline

- 1 Introduction
- 2 Introduction à HTML
- 3 CSS
- 4 Couleur
- 5 Mise en forme de textes
  - Les propriétés de type font
  - Les propriétés CSS de type text-
- 6 Le Modèle des Boites en CSS
- 7 CSS héritage et Sélecteurs
- 8 Les pseudo classes de CSS
- 9 Positionnement Responsive avec les Flexbox

# Introduction

- **HTML** permet de décrire le contenu d'une page Web.
- **CSS** permet de décrire l'affichage d'une page Web.

## HTML

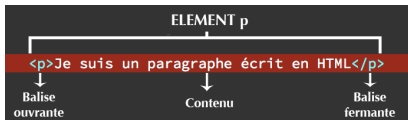


## CSS

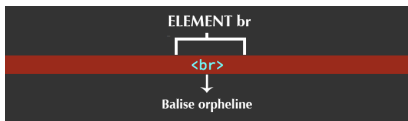


# Introduction à HTML

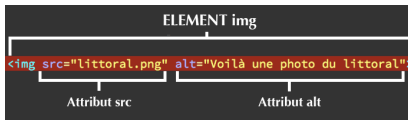
## ■ Balise (élément) HTML:



## ■ Balise orpheline :



## ■ Attribut d'une balise:



# Introduction à HTML

## ■ Structure d'un document HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Première page HTML</title>
    <meta charset= "utf-8">
  </head>
  <body>
  </body>
</html>
```

## ■ Imbrication de balises HTML:

```
<balise ouvrante élément A>
  <balise ouvrante élément B>
  </balise fermante élément B>

  <balise ouvrante élément C>
  </balise fermante élément C>

  <balise orpheline élément D>

</balise fermante élément A>
```

## ■ Commentaires en HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Première page HTML</title>
    <meta charset= "utf-8">
  </head>
  <body>
    <!-- Je suis un commentaire, je ne serai pas affiché-->
    <h1>Je viens d'écrire un titre en HTML !</h1>

    <!--Ci-dessous, vous allez trouver
    mon premier paragraphe-->
    <p>Et voilà mon premier paragraphe :)</p>
  </body>
</html>
```

# Introduction à HTML

## ■ Titres et Paragraphes:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Première page HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <!--Deux titres h1 et h2-->
    <h1>Mon titre principal</h1>
    <h2>Je suis un titre important</h2>

    <!--Deux paragraphes différents-->
    <p>Voici mon premier paragraphe.</p>
    <p>Et en voilà un second !</p>
  </body>
</html>
```

## ■ Résultat:



# Introduction à HTML

## ■ Retour à la ligne:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Première page HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <!--Les espaces ne seront pas affichés-->
    <h1>Mon titre principal</h1>
    <h2>Je suis un titre important</h2>

    <!--br nous sert à créer des retours à la ligne-->
    <p>Voici <br>mon <br>premier <br>paragraphe.</p>
    <p>Et en voilà un second !</p>
  </body>
</html>
```

## ■ Résultat:



## Mon titre principal

### Je suis un titre important

Voici  
mon  
premier  
paragraphe.

Et en voilà un second !

# Introduction à HTML

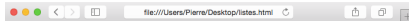
## ■ Listes ordonnées et non-ordonnées:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les listes en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Les listes</h1>

    <!--Listes imbriquées-->
    <ol>
      <li>Introduction</li>
      <li>Partie I
        <!--On imbrique une liste non-ordonnée dans une liste ordonnée-->
        <ul>
          <li>Définitions</li>
          <li>Auteurs</li>
          <li>Exemples</li>
        </ul>
      </li>
      <li>Partie II</li>
      <li>Conclusion</li>
    </ol>
  </body>
</html>
```

## ■ Résultat:



### Les listes

1. Introduction
2. Partie I
  - Définitions
  - Auteurs
  - Exemples
3. Partie II
4. Conclusion



# Introduction à HTML

## ■ Liens externes:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les liens en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Les liens</h1>

    <p>Cliquez sur <a href="http://wikipedia.org">ce lien </a>pour
      aller sur Wikipédia.</p>
  </body>
</html>
```

## ■ Résultat:



## Les liens

Cliquez sur [ce lien](http://wikipedia.org) pour aller sur Wikipédia.

# Introduction à HTML

## ■ Liens internes:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les liens en HTML</title>
    <meta charset= "utf-8">
  </head>

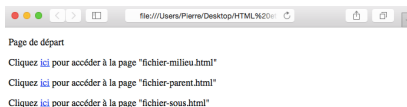
  <body>
    <p>Page de départ</p>

    <!--On crée un lien vers une page dans un même dossier-->
    <p>Cliquez <a href="fichier-milieu.html">ici</a> pour accéder à
    la page "fichier-milieu.html"</p>

    <!--On crée un lien vers une page dans un dossier parent-->
    <p>Cliquez <a href=" ../fichier-parent.html">ici</a> pour accéder à
    la page "fichier-parent.html"</p>

    <!--On crée un lien vers une page dans un sous dossier-->
    <p>Cliquez <a href="sous/fichier-sous.html">ici</a> pour accéder à
    la page "fichier-sous.html"</p>
  </body>
</html>
```

## ■ Résultat:



# Introduction à HTML

- Liens pour se déplacer dans la même page:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les liens en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Les ancres</h1>
    <p>Accès direct à <a href="#ancrer1">mon ancre n°1</a></p>
    <h2 id="ancrer1">Une première ancre</h2>
  </body>
</html>
```

- Résultat:



# Introduction à HTML

- Liens pour se déplacer dans la même page:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les liens en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Les ancrs</h1>
    <p>Accès direct à <a href="#ancrer1">mon ancre n°1</a></p>
    <h2 id="ancrer1">Une première ancre</h2>
  </body>
</html>
```

- Liens pour télécharger un fichier:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les liens en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Télécharger un fichier</h1>
    <p>Téléchargez <a href="cours.pdf">le cours</a></p>
  </body>
</html>
```

# Introduction à HTML

## ■ Les images:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Insérer des images en HTML</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Un joli coucher de soleil</h1>
    
  </body>
</html>
```

## ■ Résultat:



## Un joli coucher de soleil



# Introduction à CSS

## ■ Les sélecteurs:

```
p{  
  color: blue;  
  font-size: 14px;  
}
```

- **p** est un sélecteur pour les balise **<p>**
- Les propriétés sont définies en tuples **propriété: valeur;** a l'intérieur d'accolades

# Où écrire le code CSS?

- A l'intérieur du code HTML avec la balise `<style></style>` :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Où écrire le CSS ?</title>
    <meta charset= "utf-8">

    <style>
      body{
        background-color: orange;
      }
      p{
        color: blue;
        font-size: 16px;
      }
    </style>
  </head>

  <body>
    <h1>Un titre de niveau 1</h1>
    <p>Un paragraphe</p>
    <p>Un deuxième paragraphe</p>
  </body>
</html>
```

- Résultat:



# Où écrire le code CSS?

- Dans la balise ouvrante des éléments HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Où écrire le CSS ?</title>
    <meta charset= "utf-8">
  </head>

  <body style="background-color:orange;">
    <h1>Un titre de niveau 1</h1>
    <p style="color: blue;font-size: 16px;">Un paragraphe</p>
    <p>Un deuxième paragraphe</p>
  </body>
</html>
```

- Résultat:





# Introduction à CSS

## ■ Commentaires CSS:

```
/*Je suis un commentaire*/
```

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

```
/*Un deuxième  
 *commentaire,  
 *multi-lignes*/
```

```
p{  
    /*color: blue;*/  
    font-size: 14px;  
}
```

# Introduction à CSS

## ■ Les sélecteurs de type élément:

```
/*Notre titre h1 va s'afficher en rouge*/  
h1{  
    color: red;  
}  
  
/*Nos paragraphes seront bleus*/  
p{  
    color: blue;  
}  
  
/*Le texte de nos liens sera vert et gras*/  
a{  
    color: green;  
    font-weight: bold;  
}
```

## ■ Résultat:



## Un titre de niveau 1

Un paragraphe contenant un [lien](#)

Un deuxième paragraphe

# Introduction à CSS

- Les attributs **id** et **class** et les sélecteurs associés:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Où écrire le CSS ?</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Un titre de niveau 1</h1>
    <p id="p1">Un paragraphe contenant</p>
    <p class="p1">Un deuxième paragraphe</p>
    <p>Un troisième paragraphe</p>
  </body>
</html>

/*L'élément portant l'id "p1" sera en bleu*/
#p1{
  color: blue;
}

/*L'élément portant la class "p1" sera en rouge*/
.p1{
  color: red;
}
```

- Résultat:



## Un titre de niveau 1

Un paragraphe contenant

Un deuxième paragraphe

# Héritage en CSS

- La notion d' **héritage** signifie que tout élément HTML enfant va hériter, « en cascades », des styles de ses parents:

```
<!DOCTYPE html>
<html>
  <head>
    <title>L'héritage en CSS</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Un titre de niveau 1</h1>
    <p>Un paragraphe avec du <strong>texte important</strong></p>
    <p>Un autre paragraphe</p>
  </body>
</html>

body{
  color: purple;
}
```

- Résultat:



## Un titre de niveau 1

Un premier paragraphe

Un deuxième paragraphe

Un troisième paragraphe

# Priorité en CSS

- Si un élément reçoit plusieurs fois une même propriété avec des valeurs différentes, le style qui va être prioritaire va être le style le plus proche de l'élément :

```
body{  
    color: purple;  
}  
  
p{  
    color: blue;  
}  
  
#para3{  
    color: red;  
}  
  
strong{  
    color: green;  
}
```

- Résultat:



## Un titre de niveau 1

Un paragraphe avec du texte important

Un autre paragraphe

Un troisième paragraphe

# Les types d'éléments HTML: Block vs. Inline

- Un élément de type **block** va toujours commencer sur une nouvelle ligne et prendre toute la largeur disponible dans la page:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Eléments block vs inline</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Eléments block et inline</h1>
    <p>L'élément p est un élément de type block</p>
    <p>Un <strong>autre</strong> paragraphe</p>
  </body>
</html>

p{
  border: 4px solid #888B11;
}
```

- Résultat:



## Eléments block et inline



- Exemples: les balise **<p>**, **<h1>**, **<div>**, etc.

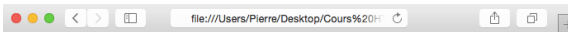
# Les types d'éléments HTML: Block vs. Inline

- Un élément de type **inline** ne va pas commencer sur une nouvelle ligne mais s'insérer dans la ligne actuelle:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Eléments block vs inline</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <h1>Eléments block et inline</h1>
    <p>L'élément p est un élément de type block</p>
    <p>Un <strong>autre</strong> paragraphe</p>
  </body>
</html>
```

```
strong{
  border: 4px solid #888811;
}
```

- Résultat:



## Eléments block et inline

L'élément p est un élément de type block

Un **autre** paragraphe

- Exemples: les balise `<a>`, `<img>`, `<span>`.

# Les éléments HTML <div> et <span>

- L'élément HTML **div** est un élément de type block. Cet élément va souvent être utilisé comme conteneur pour plusieurs autres éléments HTML:

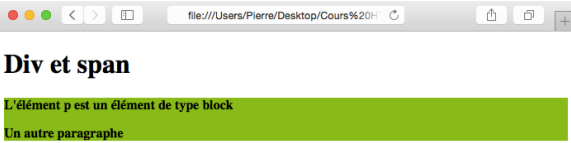
```
<!DOCTYPE html>
<html>
  <head>
    <title>Eléments div et span</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Div et span</h1>

    <div class="div-para">
      <p>L'élément p est un élément de type block</p>
      <p>Un <strong>autre</strong> paragraphe</p>
    </div>
  </body>
</html>

/*Nous utilisons des notations hexadécimales pour la couleur ici.
 *Nous verrons ces notations plus tard dans le cours*/
div-para{
  background-color: #888B11;
  font-weight: bold;
}
```

- Résultat:





# Les éléments HTML <div> et <span>

- L'élément HTML `span` est un élément de type inline. Cet élément va souvent servir de conteneur pour du contenu textuel:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Éléments div et span</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <h1>Div et span</h1>

    <div class="green">
      <p>L'élément p est <span class="blue">un élément de type block</span></p>
      <p>Un <strong>autre</strong> paragraphe</p>
    </div>
  </body>
</html>

/*Nous utilisons des notations hexadécimales pour la couleur ici.
 *Nous verrons ces notations plus tard dans le cours*/
.green{
  background-color: #88BB11;
  font-weight: bold;
}

.blue{
  background-color: #1188BB;
}
```

- Résultat:



## Div et span

L'élément p est un élément de type block

Un autre paragraphe

# La couleur en CSS

- La propriété **color** (ou **background-color**) accepte différents types de valeurs :

- 1 Un nom de couleur (red, blue, etc.) ;
- 2 Une valeur hexadécimale (#AA8811, etc.) ;
- 3 Une valeur RGB ou RGBA.

```
/*On définit une couleur de fond pour bien voir la
*transparence de nos textes.
*Nous étudierons le fond plus tard dans ce cours*/
body{
    background-color: #008080;
}

h1{
    color: orange;
    opacity: 0.5;
}

/*Quasiment transparent*/
.p1{
    color: #FFF;
    opacity: 0.2;
}

/*Quasiment opaque*/
.p2{
    color: RGBA(255,000,000,0.9);
}
```

- Résultat:



## Couleur - Exercice

Faites l'exercice "ex1" du dossier "Exercice HTML/CSS"

# La police de Texte

- La propriété CSS **font-family** va nous permettre de définir la police de notre texte, généralement on définit plusieurs police dans le cas où le navigateur n'a pas la police souhaitée

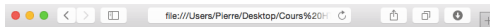
```
<!DOCTYPE html>
<html>
  <head>
    <title>Propriétés de type font</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Les propriétés de type font</h1>

    <p class="p1">Un premier paragraphe</p>
    <p class="p2">Un autre paragraphe</p>
  </body>
</html>
```

```
body{
  font-family: "Source code pro", Verdana, sans-serif;
}
```

- Résultat:



## Les propriétés de type font

Un premier paragraphe

Un autre paragraphe

# La police de Texte: Google font

- Google a développé sa propre liste de polices d'écriture appelées les Google Fonts :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Propriétés de type font</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
    <link rel="stylesheet" href="http://fonts.googleapis.com/css?family=Open+Sans">
  </head>

  <body>
    <h1>Les propriétés de type font</h1>

    <p class="p1">Un premier paragraphe</p>
    <p class="p2">Un autre paragraphe</p>
  </body>
</html>
```

```
body{
  font-family: "Open sans", Georgia, serif;
}
```

- Résultat:



## Les propriétés de type font

Un premier paragraphe

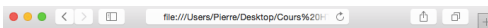
Un autre paragraphe

# Les propriétés de type font

- Les propriétés CSS de type **font-** vont nous permettre de modifier l'apparence de notre police d'écriture, et donc de nos textes:
- La propriété CSS **font-size** va nous permettre de modifier la taille de notre police:

```
/*Notre titre h1 aura une taille de 40px*/  
h1{  
    font-size: 40px;  
}  
  
/*Notre paragraphe p1 aura une taille de 150%*/  
.p1{  
    font-size: 150%;  
}  
  
/*Attention à bien utiliser un point et non pas une virgule*/  
.p2{  
    font-size: 1.2em;  
}
```

- Résultat:



## Les propriétés de type font

Un premier paragraphe

Un autre paragraphe

# Les propriétés de type font

- La propriété CSS **font-style** va nous permettre de modifier le style de notre police:

```
/*Le texte de notre paragraphe p1 s'affiche en italique*/  
.p1{  
    font-style: italic;  
}  
  
/*Le texte de notre paragraphe p1 s'affiche penché*/  
.p2{  
    font-style: oblique;  
}
```

- Résultat:



## Les propriétés de type font

*Un premier paragraphe*

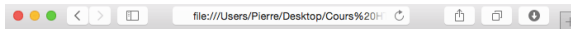
*Un autre paragraphe*

# Les propriétés de type font

- La propriété CSS **font-weight** va nous permettre de définir le poids d'une police, c'est-à-dire son épaisseur: **bold**, **normal** ou un nombre.

```
/*Le texte de notre paragraphe p1 est épais (gras)*/  
.p1{  
    font-weight: bold;  
}  
  
/*Le texte de notre paragraphe p2 est normal*/  
.p2{  
    font-weight: 400;  
}
```

- Résultat:



## Les propriétés de type font

Un premier paragraphe

Un autre paragraphe



# Les propriétés CSS de type « text- »

- Les propriétés CSS de type **text-** vont nous permettre de changer la mise en forme de nos textes et leur apparence:
- La propriété CSS **text-align** va nous permettre de définir l'alignement de notre texte: **left**, **right**, **center**, ou **justify**.

```
h1{  
    text-align: center;  
}  
  
.p1{  
    text-align: left;  
}  
  
.p2{  
    text-align: right;  
}
```

- Résultat:



## Propriétés CSS "text-"

Un premier paragraphe

Un autre paragraphe

# Les propriétés CSS de type « text- »

- La propriété CSS `text-transform` va nous permettre de transformer un texte en majuscules ou en minuscules:

```
h1{
  text-transform: uppercase;
}

.p1{
  text-transform: lowercase;
}

.p2{
  text-transform: capitalize;
}
```

- Résultat:



# Les propriétés CSS de type « text- »

- La propriété CSS `text-decoration` va nous permettre d'ajouter ou d'enlever des décorations à nos textes:

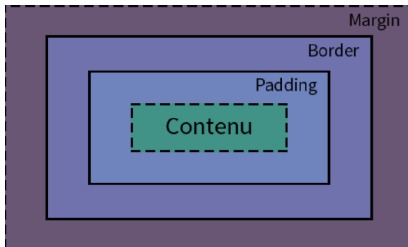
```
h1{  
  text-decoration: underline;  
}  
  
.p1{  
  text-decoration: line-through;  
}  
  
.p2{  
  text-decoration: overline;  
}  
  
a{  
  text-decoration: none;  
}
```

- Résultat:



# Margin, Padding et Border

- Chaque balise a un **padding**, **margin** et une **border** :



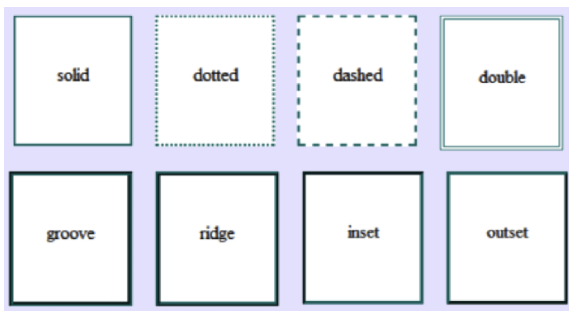
```
div{
  /*Couleur de fond (bleu-vert)*/
  background-color: #088;
  /*Largeur de l'élément en soi*/
  width: 500px;
  /*Marge intérieure*/
  padding: 20px;
  /*Bordure (vert)*/
  border: 20px solid #0C0;
  /*Marge extérieure*/
  margin: 50px;
}
```

# Margin, Padding

- Une balise a quatre cotes (top - right- bottom - left).
- `margin-top` , `padding-top` permettent de définir l'espace en **haut** à l'extérieur et à l'intérieur d'une balise. Exemple `margin-top = 10px;`
- `margin-right` , `padding-right` permettent de définir l'espace à **droite** à l'extérieur et à l'intérieur d'une balise. Exemple `margin-right = 10px;`
- `margin-bottom` , `padding-bottom` permettent de définir l'espace en **bas** à l'extérieur et à l'intérieur d'une balise. Exemple `margin-bottom = 10px;`
- `margin-left` , `padding-left` permettent de définir l'espace à **gauche** à l'extérieur et à l'intérieur d'une balise. Exemple `margin-left = 10px;`
- `margin: 20px;` , `padding: 20px` permettent de définir un espace de 20px à l'extérieur et à l'intérieur d'une balise (de tous les cotés).
- `margin: 10px 20px 30px 40px;` , `padding: 10px 20px 30px 40px;` (haut, droite, bas, gauche) permettent de définir un espace de 10px en haut, 20px à droite, 30px en bas et 40px à gauche.

# Bordure

- Nous pouvons définir les bordures d'un élément de différentes manières en CSS : soit en utilisant les trois propriétés `border-width`, `border-color`, et `border-style`, soit en utilisant directement la notation courte `border`.
- `border-width` : permet de définir la taille de la bordure en pixel.
- `border-color` : permet de définir la couleur de la bordure en pixel.
- `border-style` : permet de définir le style (forme) de la bordure.



# Bordure

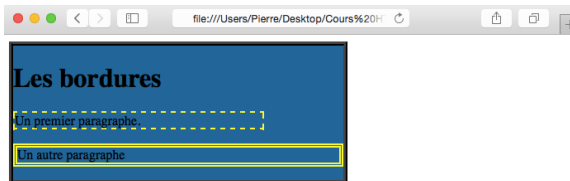
- **border** : permet de définir directement la taille, style et couleur:

```
div{
  background-color: #269;
  width: 400px;
  border: 5px ridge #444;
}

.para1{
  /*Le contenu du paragraphe va prendre 75% de la largeur de
  *son élément parent, c'est-à-dire du div*/
  width: 75%;
  border-width: 2px;
  border-style: dashed;
  border-color: yellow;
}

.para2{
  border: 5px double yellow;
}
```

- Résultat:



# Centrer des Éléments Horizontalement

- **margin** nous permet de centrer horizontalement en affectant une valeur pour les marges haute et basse (au choix) et la valeur **auto** pour les marges gauche et droite:

```
h1{
  background-color: #AA2;
  width: 400px;
  margin: 0px auto;
  text-align: center;
}

div{
  background-color: #89B;
  border: 5px solid #777;
  width: 300px;
  margin: 30px auto;
}
```

- Résultat:





## Display: Transformer une balise Block en Inline

- **Attention!** On ne peut pas donner une marge (margin) à une balise inline (contrairement aux balises block). Pour transformer une balise inline en balise block en utilise: `display: block` :
- Pour afficher une balise block comme une balise inline tout en gardant la possibilité de lui affecter une marge on utilise `display: inline-block` :

## Police et Tailles - Exercice

Faites l'exercice "ex2" du dossier "Exercice HTML/CSS"

# Le sélecteur CSS universel « \* »

- Le sélecteur **\*** (étoile) permet de sélectionner tous les éléments HTML d'une page d'un coup :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Sélecteurs CSS avancés</h1>

    <p>Un paragraphe</p>
    <p>Un <strong>paragraphe important</strong></p>
  </body>
</html>

*{
  border: 1px solid black;
  padding: 5px;
  margin: 5px;
}
```

- Résultat:



# Grouper les sélecteurs

- Appliquer un style a plusieurs éléments:

```
h1, p{  
  color: blue;  
}
```

- Résultat:



## Un titre de niveau 1

Un paragraphe

Un autre paragraphe

Un troisième paragraphe

# Sélectionner un élément par rapport à un autre

- le sélecteur **A B** (remplacez « A » et « B » par n'importe quel élément) va nous permettre de sélectionner un élément B contenu dans un élément A:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Sélecteurs CSS avancés</h1>

    <div class="test">
      <p>Un premier paragraphe dans un div</p>
      <p>Un <strong>paragraphe important</strong></p>
    </div>

    <div>
      <p>Un paragraphe dans un deuxième div</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>

.test p{
  font-size: 20px;
  color: blue;
}
```

## Sélecteurs CSS avancés

[Un premier paragraphe dans un div](#)

[Un paragraphe important](#)

[Un paragraphe dans un deuxième div](#)

# Sélectionner un élément par rapport à un autre

- Le sélecteur **A + B** nous permet de sélectionner tous les éléments B de même niveau et suivant immédiatement les éléments A:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <h1>Sélecteurs CSS avancés</h1>
    <div class="test">
      <p>Un premier paragraphe dans un div</p>
      <p>Un <strong>paragraphe important</strong></p>
    </div>
    <div>
      <p>Un paragraphe dans un deuxième div</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>

div + p{
  background-color: orange;
}
```



## Sélecteurs CSS avancés

Un premier paragraphe dans un div

Un **paragraphe important**

Un paragraphe dans un deuxième div

**Un paragraphe hors div**

# Sélectionner un élément par rapport à un autre

- Le sélecteur **A > B** va sélectionner tous les éléments B qui sont des éléments enfants directs de A:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <div class="page-container">
      <h1>Sélecteurs CSS avancés</h1>

      <div class="test">
        <p>Un premier paragraphe dans un div</p>
        <p>Un <strong>paragraphe important</strong></p>
      </div>

      <div>
        <p>Un paragraphe dans un deuxième div</p>
      </div>
      <p>Un paragraphe hors div</p>
    </div>
  </body>
</html>
```

```
body > div{
  border: 1px solid black;
}
```

Sélecteurs CSS avancés

Un premier paragraphe dans un div

Un **paragraphe important**

Un paragraphe dans un deuxième div

# Les pseudo classes de CSS

- **:hover** nous permet de changer l'aspect d'un élément lorsque les utilisateurs poseront leur curseur dessus.
- **:link** nous permet de changer l'aspect d'un lien non visité.
- **:visited** nous permet de changer l'aspect d'un lien une fois celui-ci visité.
- **:active** nous permet de modifier l'aspect d'un lien lors du clic.

```
h1:hover{
    color: orange;
    font-family: Verdana, sans-serif;
}

a:link{
    color: blue;
    text-decoration: underline;
}

/*Font-weight et text-decoration ne fonctionneront pas car les
*navigateurs modernes ont désactivé la plupart des propriétés
*CSS liées à cet état pour réduire le risque de faille de sécurité*/
a:visited{
    color: green;
    font-weight: bold;
    text-decoration: none;
}
```



# Les pseudo classes de CSS

- `:first-child` et `:last-child` nous permet de changer l'aspect le du premier (dernier) élément de son élément parent:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pseudo classes</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Pseudo classes CSS</h1>

    <div class="test">
      <p>Un premier paragraphe</p>
      <p>Un deuxième paragraphe</p>
      <p>Un troisième paragraphe</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>
```

```
p:first-child{
  color: orange;
}

p:last-child{
  color: green;
}
```

file:///Users/Pierre/Desktop/Cours%20H

## Pseudo classes CSS

Un premier paragraphe

Un deuxième paragraphe

Un troisième paragraphe

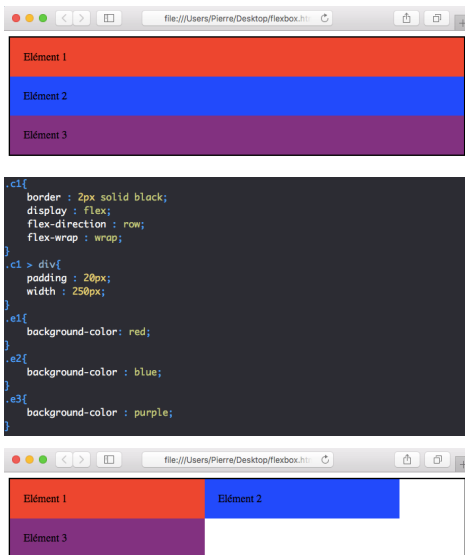
## Sélecteurs et pseudo class CSS - Exercice

Faites l'exercice "ex3" du dossier "Exercices HTML-CSS"

# Positionnement Responsive avec les Flexbox

- Le **flexbox** est aujourd'hui l'outil le plus puissant et simple pour créer des structures **responsives** et flexibles, c'est-à-dire pour positionner et aligner des éléments HTML sans même connaître leur dimension à priori.
- Le **flexbox** permet d'ordonner les enfants d'un élément HTML horizontalement ou verticalement, et d'indiquer pour chaque enfant sa dimension (width ou height) selon l'espace disponible.
- Pour utiliser le flexbox il faut appliquer **display: flex;** à la balise parent, puis indiquer le sens (**flex-direction: row;** pour une organiser les enfants horizontalement et (**flex-direction: column;** pour les organiser verticalement).

# Positionnement Responsive avec les Flexbox



## Flexbox - Exercice

Faites l'exercice "ex4" du dossier "Exercices HTML-CSS"