



# CSS

Première partie



# CSS

- Cascading Style Sheets (Feuilles de style en cascade)
- CSS 3
- Un fichier CSS contient des règles css
- Permet la mise en forme du contenu du HTML
- Qu'est-ce que la mise en forme ?
  - Couleurs des balises
  - Taille des balises
  - Placement des balises sur la page et par rapport aux autres
  - Ombrages, bordures
  - Polices de caractères
  - ...
- Peut-être intégré directement au HTML mais les bonnes pratiques veulent qu'on place le CSS dans des fichiers différents des fichiers HTML, avec l'extension .css



# CSS : Exemple de règle

```
h1 {  
  color: yellow;  
}
```

- en rouge: le sélecteur
- en bleu: la propriété
- en vert: la valeur
- Ensemble propriété-valeur: déclaration
- Une règle peut contenir plusieurs déclarations
- Les accolades, le ":" pour séparer la propriété de la valeur et le ";" pour terminer la déclaration sont indispensables



# CSS : Commentaires

```
h1 {  
    /* ceci est un commentaire CCS. Il n'est pas interprété */  
    color: yellow;  
}
```

- Ouverts par "/\*"
- Fermés par "\*/"



# CSS : À propos des conventions de code

- Les conventions de code sont des conventions sur le style du code
- Au sein d'un même projet, il y a dans la majorité des cas plusieurs développeurs. Il est important que ces développeurs utilisent le même style de code. Cela permet une meilleure compréhension et une meilleure maintenabilité du code.

Les deux exemples de CSS suivants sont tous les deux corrects syntaxiquement et font la même chose, cependant le second sera préféré car généralement considéré comme plus lisible:

```
p { color: yellow; }
```

```
p {  
    color: yellow;  
}
```



# CSS : L'attribut "id"

- L'attribut "id" est un attribut qui peut être attribué à n'importe quelle balise HTML
- Son nom signifie "identifiant", ce qui implique que sa valeur doit être unique au sein d'une page HTML
- Une balise HTML ne peut avoir au maximum qu'un seul id

```
<h1 id="ceciDoitEtreUnique" > TITRE </h1>
```



# CSS : L'attribut "class"

- L'attribut "class" est un attribut qui peut être attribué à n'importe quelle balise HTML
- Contrairement à l'id, plusieurs balises HTML peuvent avoir la même classe
- Contrairement à l'id, une même balise HTML peut avoir plusieurs classes:

```
<h1 class="myClass fontBig backgroundRed" > TITRE 2 </h1>
```



# CSS : Sélecteurs

```
h1, h2, p {  
    color: red;  
}
```

- Sélecteurs multiples: le contenu des balises <h1>,< h2> et <p> sera rouge

```
#loginText {  
    font-size: small;  
}
```

- Sélecteur d'identifiant: le contenu de la balise possédant l'attribut "id" avec la valeur "loginText" aura une taille de police *small*
- Rappel important: la valeur de l'attribut id doit être unique au sein d'un même document HTML

```
.bigText {  
    font-size: large;  
}
```

- Sélecteur de classe: le contenu des balises possédant l'attribut "class" avec la valeur "bigText" aura une taille de police *Large*

```
input[value] {  
    color: blue;  
}
```

- Sélecteur d'attribut: le contenu des balises <input> possédant l'attribut "value" sera bleu





# CSS : Sélecteurs

```
a:visited {  
    color: brown;  
}
```

```
div p {  
    color: yellow;  
}
```

```
h1, .bigText, div p {  
    font-size: large;  
}
```

- Sélecteur de pseudo-classes: affecte les balises qui sont dans l'état spécifié. Ici, les liens une fois qu'ils ont été visités
  - [Liste des pseudo-classes](#)
  - Un sélecteur peut-être imbriqué: pour celui-ci, les balises <p> au sein des balises <div> auront une couleur jaune
  - Les sélecteurs peuvent être combinés entre eux à volonté
- 
- La liste présentée ici est non-exhaustive
  - Pour voir d'autres exemples de sélecteurs:  
[https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)



# CSS : Héritage

Considérons la règle suivante:

```
div {  
    color: red;  
}
```

Et les balises HTML suivantes:

```
<div>  
    <p>Mon paragraphe</p>  
</div>
```

- Certaines propriétés sont héritées du parent (color, font-family...)
- Ici, le paragraphe sera en rouge car il hérite de la propriété "color" de son parent <div>
- Toutes les propriétés ne sont pas héritées du parent, sinon ce ne serait vraiment pas pratique...



# CSS : Pourquoi "en cascade" ?

Considérons les règles suivantes:

```
h1 {  
    color: red;  
    background-color: yellow;  
}  
#title_blue {  
    color: blue;  
}
```

Et les balises HTML suivantes:

```
<h1> TITRE 1 </h1>  
<h1 id="title_blue" > TITRE 2 </h1>
```

- En théorie, les deux règles s'appliquent au deuxième titre
- Cependant, la règle la plus spécifique a la priorité, le deuxième titre sera donc en bleu sur fond jaune
- Cascade: les règles sont choisies en descendant de la plus générique à la plus spécifique



# CSS : Pourquoi "en cascade" ?

Considérons les règles suivantes:

```
h1 {  
    color: red;  
}  
  
h1 {  
    color: blue;  
}
```

Et la balise HTML suivante:

```
<h1> TITRE 1 </h1>
```

- Le titre sera bleu car, à même niveau de spécificité, c'est la dernière règle déclarée qui prévaut



# CSS : Les unités de longueur et taille

- Il existe de nombreuses unités pour exprimer les longueurs et les tailles en css, pour les tailles de police, margin, padding, border, balises...
- les unités absolues:
  - px: pixel
  - pt: point
  - ...
- les unités relatives:
  - em: 1em = taille de la police de la balise parente
  - rem: 1rem = taille de la police de la balise racine (html, 16px)
  - vw: 1vw = 1% de la largeur de la fenêtre du navigateur
  - vh: 1vh = 1% de la hauteur de la fenêtre du navigateur
  - ...
- De manière générale, pour les attributs type hauteur, longueur, margin et padding, il vaut mieux privilégier les unités relatives, cela s'adapte mieux aux différentes résolutions d'écrans
- [Documentation sur les unités](#)

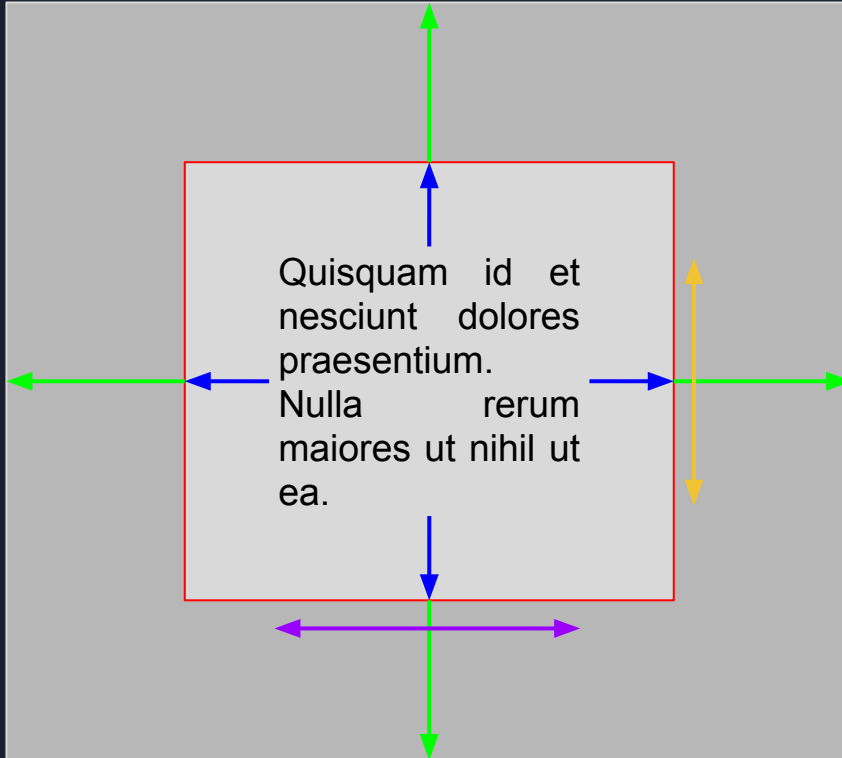


# CSS : Quelques propriétés CSS classiques

	Propriété	Description	Ex. de valeurs
Couleurs	color	couleur du texte	red, #ffbb00, rgb(10, 5, 255)
	background-color	couleur du fond	blue, #ffbb00, rgba(255, 1, 24, 0.3)
Polices	font-family	police du texte	Arial, Helvetica
	font-size	taille du texte	2em, 15px

[À propos des couleurs en hexadécimal](#)

# CSS : Le modèle des boîtes



- En vert, le margin: c'est l'espacement entre la balise et les balises autour
- En bleu, le padding: c'est l'espacement entre les bords de la balise et son contenu
- En rouge, le border: c'est la bordure de la balise, entre la zone de margin et la zone de padding
- En jaune, la hauteur (height) de la balise
- En violet, la largeur (width) de la balise



# CSS : Le modèle des boîtes

- Pour les balises type **block** (<div>, <p>, <table>...):
  - Les balises provoquent un retour à la ligne
  - Les balises prennent autant de largeur que possible (largeur du parent)
- Pour les balises type **inline** (<a>, <span>, <label>, <input>...):
  - les balises prennent juste autant d'espace que nécessaire pour leur contenu
  - les balises ne provoquent pas de retour à la ligne
- [Liste des balises HTML par type inline ou block](#)
- Le type d'une balise peut être changé avec la propriété css "display" avec la valeur "inline" ou "block"
- Le mieux pour bien comprendre tout ça, c'est d'essayer, de faire des tests





# CSS : Le modèle des boîtes

	Propriété	Description	Ex. de valeurs
Margin	margin	taille du margin	2em, 15px, -10px
	margin-   left   right   top   bottom	taille du margin   gauche   droit   haut   bas	2em, 15px, -5em
Padding	padding	taille du padding	2em, 15px
	padding-left   right   top   bottom	taille du margin gauche   droit   haut   bas	2em, 15px
Largeur	width	largeur du composant	100px, 50vw, 10em, 25%
Hauteur	height	hauteur du composant	100px, 50vh, 10em, 50%

# CSS : Le modèle des boîtes

	Propriété	Description	Ex. de valeurs
Border	border-style border-left-style, border-right-style, border-top-style, border-bottom-style	type de bordure	solid, dashed, dotted, inset, outset
	border-width border-left-width, ...	épaisseur de la bordure	1px, 0.1em
	border-color border-left-color, ...	couleur de la bordure	blue, #ffbb00
	border-radius	arrondi du coin	10px, <b>15px 50px 30px 5px</b>
	border	raccourci pour toutes les propriétés du border	<b>1px solid #32a1ce</b>
	border-   left   right   top   bottom	raccourci pour toutes les propriétés du border   gauche   droit   haut   bas	<b>0.5em dashed blue</b>



## ● CSS : Bordure de tableaux

1. Créez un nouveau répertoire de travail "CSS"
2. Dans ce répertoire, créez deux nouveaux fichiers "table.html" (avec le HTML minimum) et "table.css"
3. Modifiez votre fichier "table.html":
  - 3.1. Dans la balise <head>, créez une balise auto-fermante <link> avec en attributs:
    - 3.1.1. "rel" avec la valeur "stylesheet"
    - 3.1.2. "href" avec la valeur "table.css"
4. Votre page HTML "table.html" va maintenant se voir appliquer les styles définis dans "table.css"
5. Créez le tableau ci-dessous et faites en sorte qu'il ait des bordures visibles

Sites dédiés au développement		
Nom	Site	Theme
Open Classrooms	<a href="https://openclassrooms.com">https://openclassrooms.com</a>	Tutoriels de développement
Stack overflow	<a href="https://stackoverflow.com">https://stackoverflow.com</a>	Résolution de problèmes
w3schools	<a href="https://www.w3schools.com">https://www.w3schools.com</a>	Développement Web



# ● CSS : Visualiser les boîtes (1)

1. Dans votre répertoire de travail, créez deux nouveaux fichiers "box.html" (avec le HTML minimum) et "box.css"
2. Modifiez votre fichier "box.html" pour qu'il se voit appliquer les styles de "box.css"
3. Dans "box.html":
  - 3.1. Créez une balise <div> avec l'id "container"
  - 3.2. Dans cette <div>, créez une balise <p> avec l'id "paragraphe" et du contenu textuel de votre choix
4. Dans "box.css":
  - 4.1. Pour l'id "container":
    - 4.1.1. margin et padding: 1em
    - 4.1.2. couleur de fond: yellow
    - 4.1.3. bordure: dotted, red, 1px
  - 4.2. Pour l'id "paragraphe":
    - 4.2.1. margin et padding: 1em
    - 4.2.2. couleur de fond: blue
    - 4.2.3. bordure: solid, red, 2px
5. Ouvrez "box.html" dans votre navigateur

# ● CSS : Premiers styles (1)

1. Dans votre répertoire de travail, créez deux nouveaux fichiers "index.html" (avec le HTML minimum) et "index.css"
2. Modifiez votre fichier "index.html" pour qu'il se voit appliquer les styles de "index.css"
3. Créez un formulaire de connexion dans une balise <div>, qui contient:
  - 3.1. Un titre <h1> avec le contenu "Connexion"
  - 3.2. Un input text avec le label "Nom d'utilisateur:", un input password avec le label "Mot de passe:", un input checkbox avec le label "Se souvenir"
  - 3.3. Chaque couple <label><input> doit être dans une balise <div>
  - 3.4. Dans une balise <div>, un lien <a> sans attribut href pour le moment, avec le contenu "Connexion"

# Connexion

Nom d'utilisateur:

Mot de passe:

☐ Se souvenir

[Connexion](#)



## ● CSS : Premiers styles (2)

1. Un peu de couleur:
  - 1.1. Appliquez la couleur "red" aux titres <h1>
  - 1.2. Appliquez la couleur "#00008B" aux <label>
  - 1.3. Appliquez la couleur de fond "#e6fff2" aux <input>
  - 1.4. Appliquez la couleur de fond "#ffffcc" à la page <html>
2. Un peu de police:
  - 2.1. Appliquez la police "Arial" à l'ensemble de votre document (sélecteur \*)
  - 2.2. Appliquez la taille de police "2em" à vos titres <h1>
3. Un peu d'espace:
  - 3.1. Affectez la classe "input-spacing" aux balises <div> parentes de chaque input
  - 3.2. Appliquez la propriété "margin-top" avec la valeur "1em" à la classe "input-spacing"
4. Un peu de taille:
  - 4.1. Affectez l'identifiant "login-page" à la balise <div> parente du formulaire
  - 4.2. Appliquez la hauteur "100vh" à l'id "login-page"
5. Utilisez l'inspecteur de votre navigateur pour examiner le code



## ● CSS : Premiers styles (2)

### Connexion

Nom d'utilisateur:

Mot de passe:

☐ Se souvenir

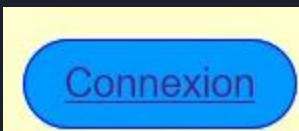
[Connexion](#)

C'est beau non ?

## ● CSS : Premiers styles (3)

Donnez l'apparence d'un bouton au lien de connexion:

1. Affectez l'identifiant "connection-button" à la balise <div> parente de la balise <a>
2. Appliquez les propriétés suivantes sur l'id "connection-button":
  - 2.1. width: 6vw
  - 2.2. margin-top: 1em
  - 2.3. padding: 0.5em
  - 2.4. border: solid, 1px, blue
  - 2.5. border-radius: 2em
  - 2.6. background-color: #0099ff
  - 2.7. display: flex
  - 2.8. justify-content: center
3. Appliquez la propriété "background-color" avec la valeur "#97d3fc" au sélecteur "#connection-button:hover" (sélecteur de pseudo-classe :hover sur l'identifiant #connection-button)
4. Votre bouton doit changer de couleur quand la souris passe dessus



Connexion





# CSS : Les textes

Propriété	Description	Ex. de valeurs
text-decoration	souligner, barrer	none, underline, overline, line-through
text-transform	majuscule, minuscule	none, uppercase, lowercase, capitalize
font-weight	graisse du texte	normal, bold, 100–900
font-style	Italique	normal, italic, oblique



# CSS : Les textes

Propriété	Description	Ex. de valeurs
text-align	alignement	left, right, center, justify
line-height	hauteur de ligne	1.5, 2
letter-spacing	espacement des lettres	2px, 4px
word-spacing	espacement des mots	2px, 4px



# ● CSS : Les textes (1)

- Lien utile: [le lorem ipsum](#)

1. Dans votre répertoire de travail, créez deux nouveaux fichiers "logged.html" (avec le HTML minimum) et "logged.css"
2. Modifiez votre fichier "index.html" pour que le lien du bouton "Connexion" pointe vers la page "logged.html"
3. Modifiez votre fichier "logged.html" pour qu'il se voit appliquer les styles de "logged.css"
4. Dans le body de "logged.html":
  - 4.1. Créez une balise <div> ayant l'identifiant "container" et dans cette balise:
    - 4.1.1. Créez un titre <h1> et un titre <h2>
    - 4.1.2. Créez quatre paragraphes précédés par des titres <h3>, avec suffisamment de contenu pour qu'ils fassent plusieurs lignes



## ● CSS : Les textes (2)

- Lien utile: [le lorem ipsum](#)

1. Dans "logged.css":
  - 1.1. Créez une règle qui justifie le texte des balises ayant la classe "justify"
  - 1.2. Créez une règle qui barre et met en gras le texte des balises ayant la classe "barrer"
  - 1.3. Créez une règle qui met en majuscules rouges, souligne, centre et espace un peu les lettres de tous les niveaux de titres
  - 1.4. Créez une règle qui augmente la hauteur de ligne des paragraphes et leur affecte la taille de police 1em
  - 1.5. Créez une règle qui affecte la taille de police 2em aux titres h1
  - 1.6. Créez une règle qui affecte la taille de police 1.5em aux titres h2
  - 1.7. Créez une règle qui affecte la taille de police 1.25em aux titres h3
  - 1.8. Créez une règle qui affiche une bordure bleue, ajoute un margin (20px) et un padding (20px) à l'identifiant "container"
  - 1.9. Créez une règle qui affecte la taille de police 2em aux paragraphes quand la souris passe dessus (pseudo-classe ":hover")
2. Justifiez tous les paragraphes en utilisant la classe créée en 1.1
3. Dans chaque paragraphe, utilisez la classe créée en 1.2 sur un mot (indice: vous aurez besoin de la balise <span>)



# CSS, mise en page : La Flexbox

Il existe plusieurs moyens d'organiser les balises entre elles au sein d'une page:

- En utilisant les tableaux (ça, c'était avant, et c'était pas mieux avant)
- En utilisant les boîtes flottantes (old school)
- En utilisant la Flexbox
- La Flexbox, bien que pratique et populaire, est maintenant concurrencée par la disposition en grille (Grid)

Flexbox:

- Flexible Box Layout Module = Mise en page de boîtes modulaires
- Détermine le placement des balises enfants dans le parent
- Pour activer la Flexbox d'une balise, il faut lui donner la propriété "display" avec la valeur "flex". Les balises enfants vont alors s'organiser selon le modèle Flexbox



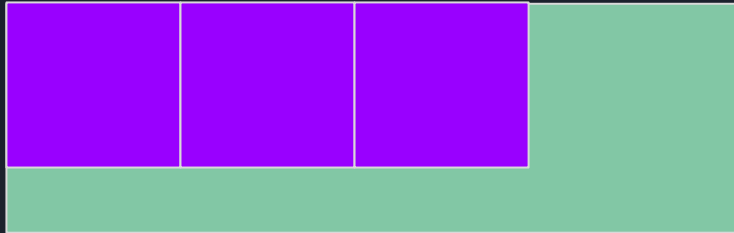
# CSS, mise en page : La Flexbox

Au sein d'une flexbox:

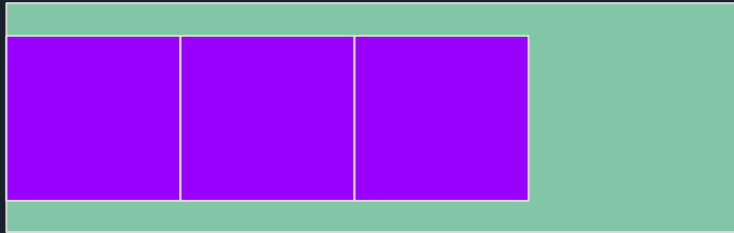
- Les enfants vont s'organiser soit horizontalement, soit verticalement. Ce comportement est défini par la propriété du parent "flex-direction" avec la valeur "column" ou "row", la valeur par défaut. Il existe aussi "column-reverse" et "row-reverse"
- Les enfants vont se partager l'espace de manière égale. Un poids peut-être affecté à chaque enfant avec la propriété "flex" et une valeur numérique

# CSS, mise en page : La Flexbox

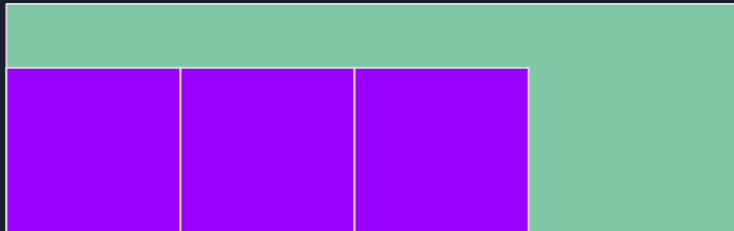
La propriété align-items (propriété du parent)



align-items: flex-start



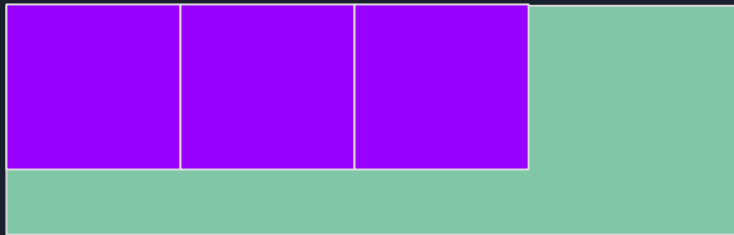
align-items: center



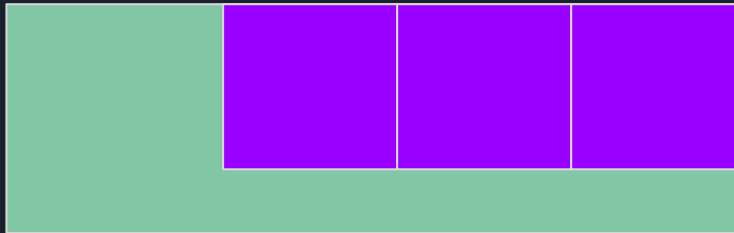
align-items: flex-end

# CSS, mise en page : La Flexbox

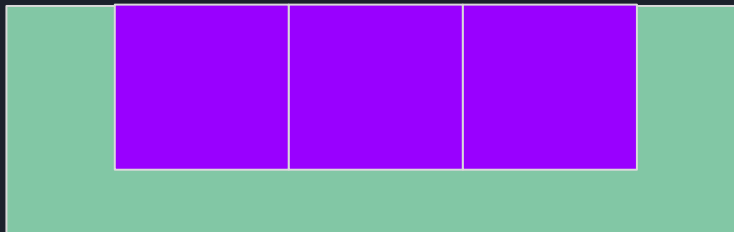
La propriété justify-content (propriété du parent)



justify-content: flex-start



justify-content: flex-end

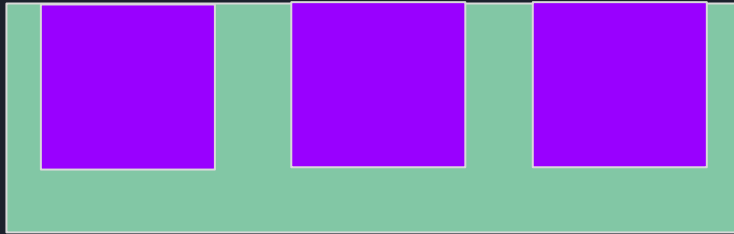


justify-content: center

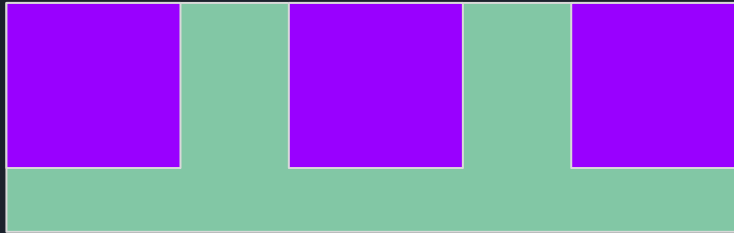


# CSS, mise en page : La Flexbox

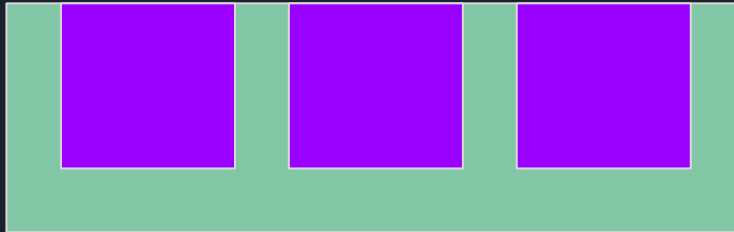
La propriété justify-content (propriété du parent)



justify-content: space-around



justify-content: space-between



justify-content: space-evenly

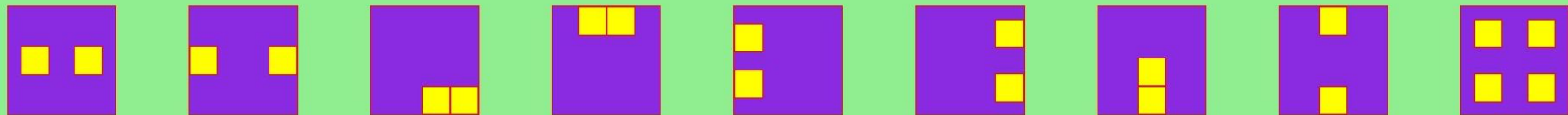


# ● CSS : Comprendre la flexbox (1)

1. Dans votre répertoire de travail, créez deux nouveaux fichiers "flexbox.html" (avec le HTML minimum) et "flexbox.css"
2. Modifiez votre fichier "flexbox.html" pour qu'il se voit appliquer les styles de "flexbox.css"
3. Dans "flexbox.html" ou "flexbox.css" le cas échéant
  - 3.1. Créez une balise <div> à laquelle vous affecterez l'id "container"
  - 3.2. Dans cette balise, créez 3 <div> auxquelles vous affecterez la classe "child"
  - 3.3. Créez une règle pour l'id "container" contenant les propriétés suivantes:
    - 3.3.1. display: flex
    - 3.3.2. background-color: lightgreen
    - 3.3.3. height: 150px
    - 3.3.4. align-items: flex-start
    - 3.3.5. justify-content: flex-start
  - 3.4. Créez une règle pour la classe "child" contenant les propriétés suivantes:
    - 3.4.1. background-color: blueviolet
    - 3.4.2. height: 100px
    - 3.4.3. width: 100px
    - 3.4.4. border: solid red 1px
4. Jouez avec les attributs "align-items" et "justify-content" pour visualiser leur fonctionnement

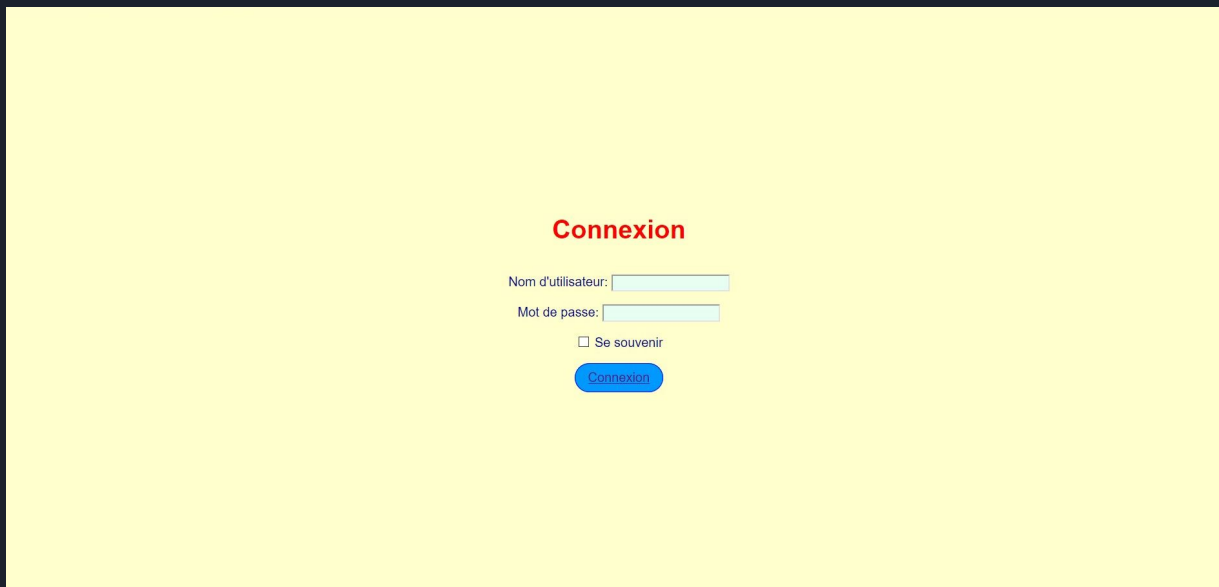
## ● CSS : Comprendre la flexbox (2)

1. Modifiez l'exercice précédent pour obtenir l'affichage suivant:



## ● CSS : Comprendre la flexbox (3)

1. En utilisant les flexbox, modifiez vos pages "index.html" et "index.css" (la page de login) pour centrer le formulaire de login au milieu de la page et centrer tous les éléments qu'il comporte (indice: vous aurez besoin de la propriété "flex-direction" avec la valeur "column")



**Connexion**

Nom d'utilisateur:

Mot de passe:

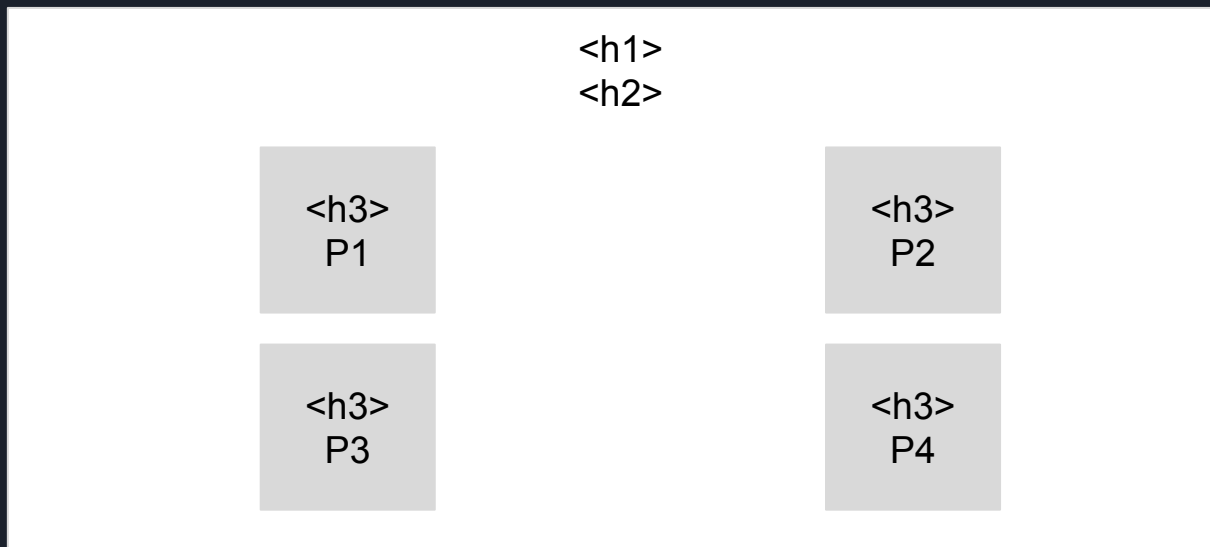
☐ Se souvenir

[Connexion](#)



## ● CSS : Comprendre la flexbox (4)

1. En utilisant les flexbox, modifiez vos pages "logged.html" et "logged.css" pour que les quatre paragraphes soient disposés en carrés





# CSS, mise en page : La Flexbox

- Comme vu dans l'exercice précédent, vous pouvez imbriquer autant de flexbox que vous voulez entre elles
- Cela permet de créer des mises en page complexes



# CSS et compatibilité

- Tous les navigateurs et un même navigateur entre ses différentes versions n'auront pas la même prise en charge du CSS
- Vous pouvez vérifier la compatibilité avec le site <https://www.caniuse.com>



# ● CSS, exercice: Rendez votre CV stylé

- Reprenez votre CV en HTML et rendez-le stylé grâce au CSS:
  - Ajoutez quelques couleurs (utilisez le [sélecteur de couleurs](#) pour choisir des couleurs sobres et pas des couleurs "flash"), aux titres par exemple
  - Utilisez des tailles de police variée en fonction des différents niveaux de texte (titre, titres de sections, paragraphes...)
  - Utilisez le soulignage de texte
  - Utilisez des border variés pour encadrer vos sections de manière visible
  - Tout votre CV devra être organisé en utilisant des flexbox imbriquées