



HTML | CSS



Aïcha Bakki



aicha.bakki@univ-angers.fr



Application Web

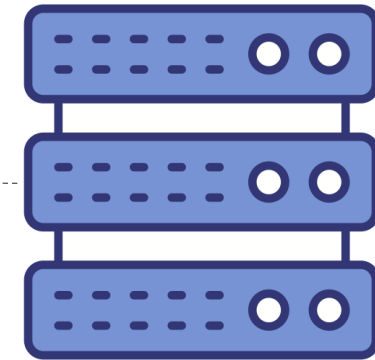
Client



Quand vous utilisez un navigateur Web pour accéder à un site



Serveur



Ce sont les ordinateurs qui délivrent les site Web aux internautes (clients)



APACHE

Requête

Internet

Réponse



Application Web : technologies

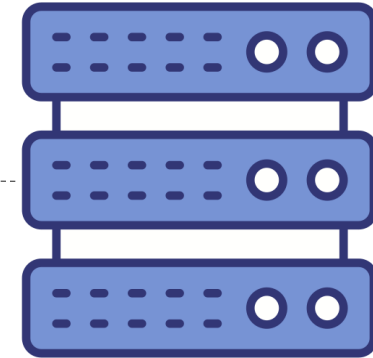
Client



Internet



Serveur





Application Web : technologies

Client

HTML



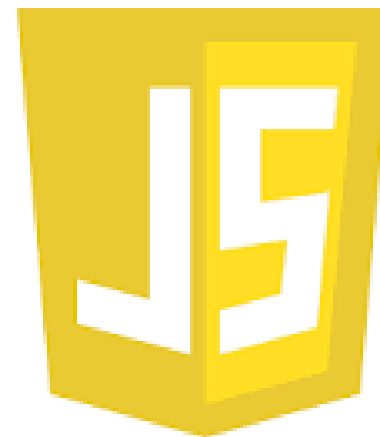
Markup Language
Content

CSS



Style Sheet Language
Presentation

JS

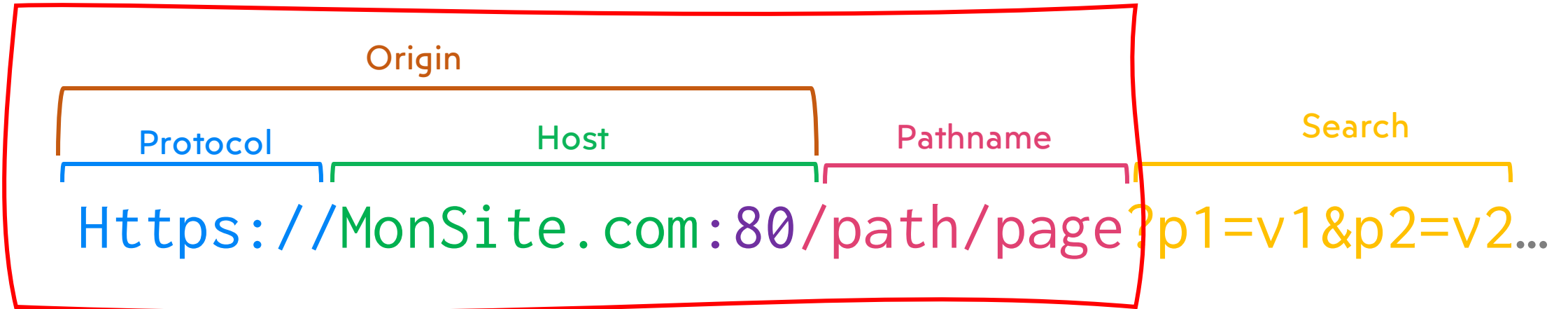


Programming Language
Behavior

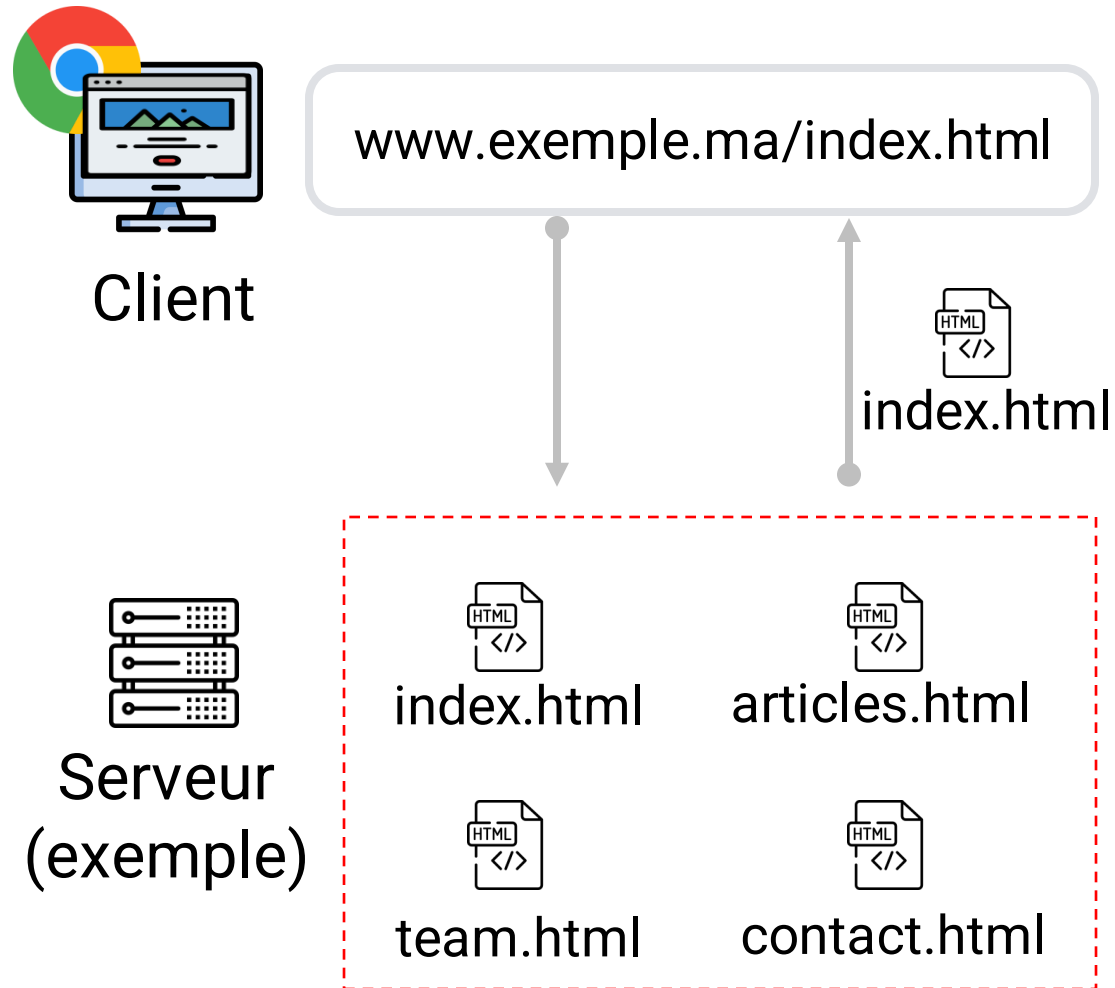




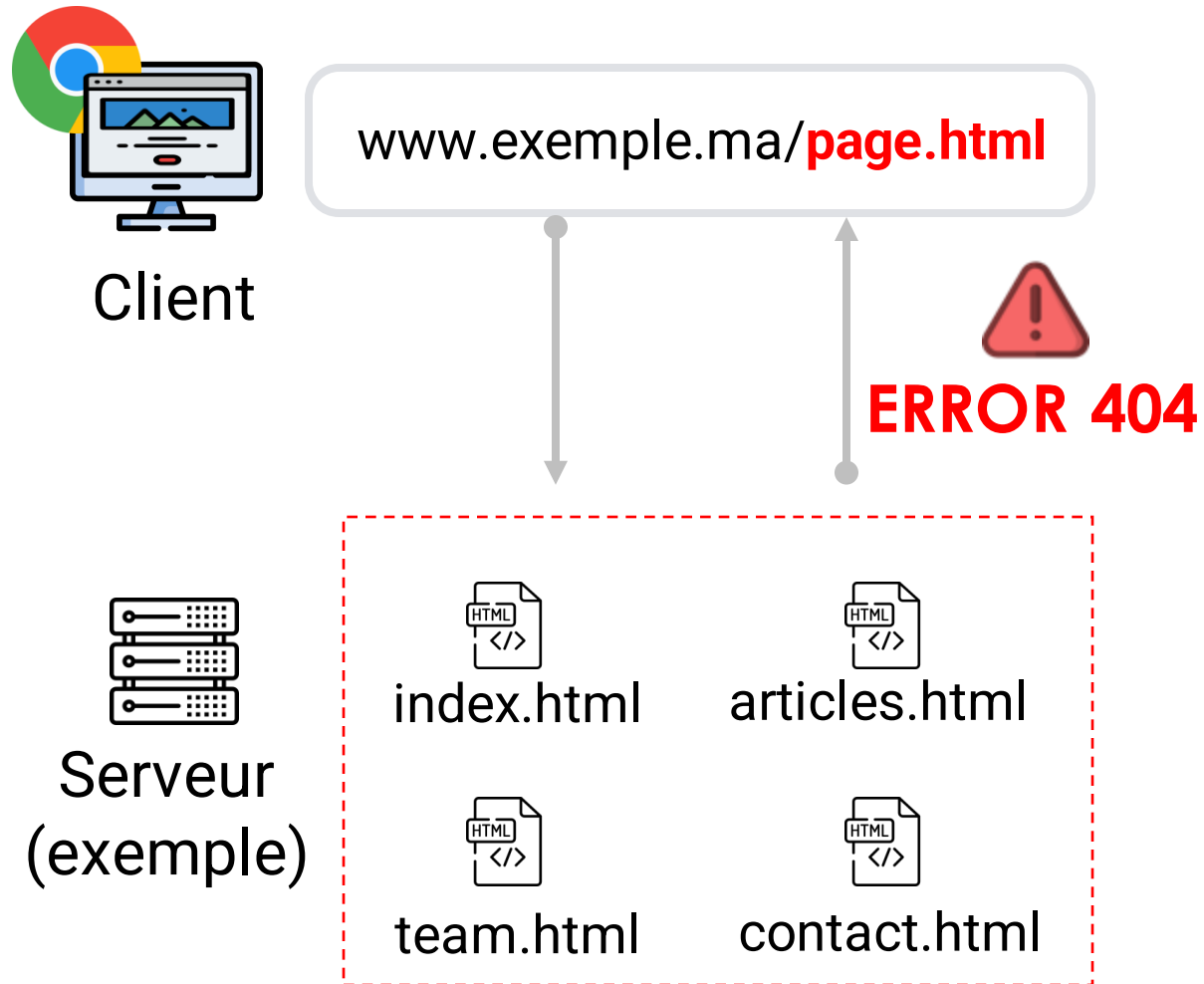
Demande de ressources : URL



Demande de ressources : URL



Demande de ressources : URL





Web Statique

HTML

HyperText Markup Language

- Décrire le contenu d'une page web – Contenu Statique.
- HTML 5 – Sorti en octobre 2014
- + Tous les sites web utilisent HTML
- Texte Formaté
- Langage descriptif – langage balisé



HTML

Structure minimale d'une page web

```
<!DOCTYPE html>
```

précise au navigateur quelle version de HTML est utilisée

```
<html>
```

```
  <head>
```

```
    <title> MonSite: Inscription </title>
```

```
    <meta name="" content="" />
```

```
  </head>
```

```
  <body>
```

```
    <img src = "header_logo.jpg">
```

```
  </body>
```

```
</html>
```



HTML

Structure minimale d'une page web



```
<!DOCTYPE html>
```

La ou on va insérer tout le contenu de notre page

```
<html dir="ltr" lang="fr-FR">
```

```
<head>
```

```
<title> MonSite: Inscription </title>
```

```
<meta name="" content="" />
```

```
</head>
```

```
<body>
```

```
<img src = "header_logo.jpg">
```

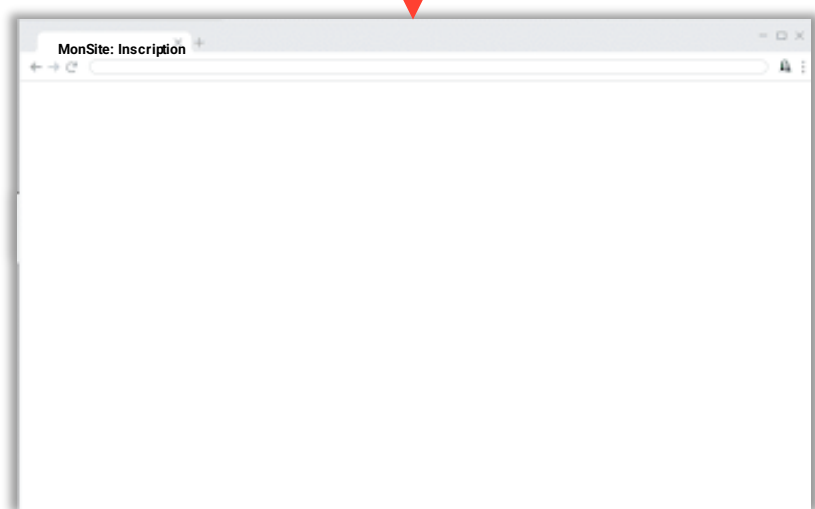
```
</body>
```

```
</html>
```



HTML

Structure minimale d'une page web



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

fournir des informations sur la page au navigateur

```
<title> MonSite: Inscription </title>
```

```
<meta name="" content="" />
```

```
</head>
```

```
<body>
```

```
<img src = "header_logo.jpg">
```

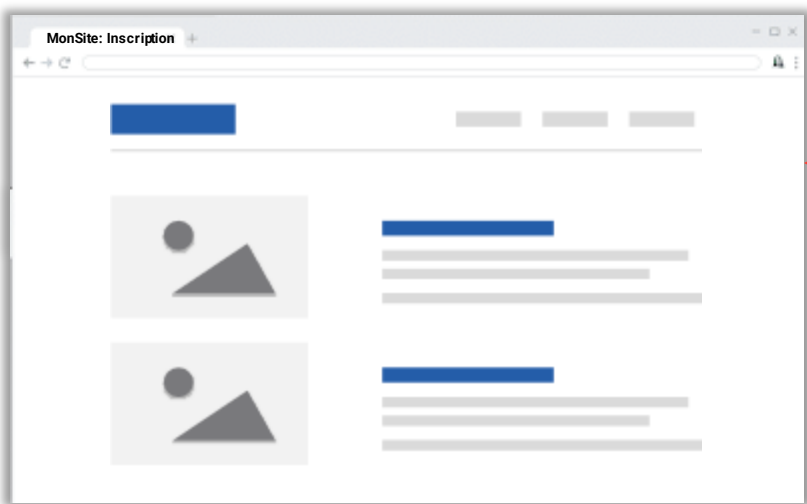
```
</body>
```

```
</html>
```



HTML

Structure minimale d'une page HTML valide



```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title> MonSite: Inscription </title>
```

```
    <meta name="" content="" />
```

```
  </head>
```

```
  <body>
```

Décrit le contenu « visible » de la page

```
    <img src = "header_logo.jpg">
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>

<html>

  <head>

    <title> MonSite: Inscription </title>

    <meta name="keyword" content="UA" />

  </head>

  <body>

    <img src = "header_logo.jpg">

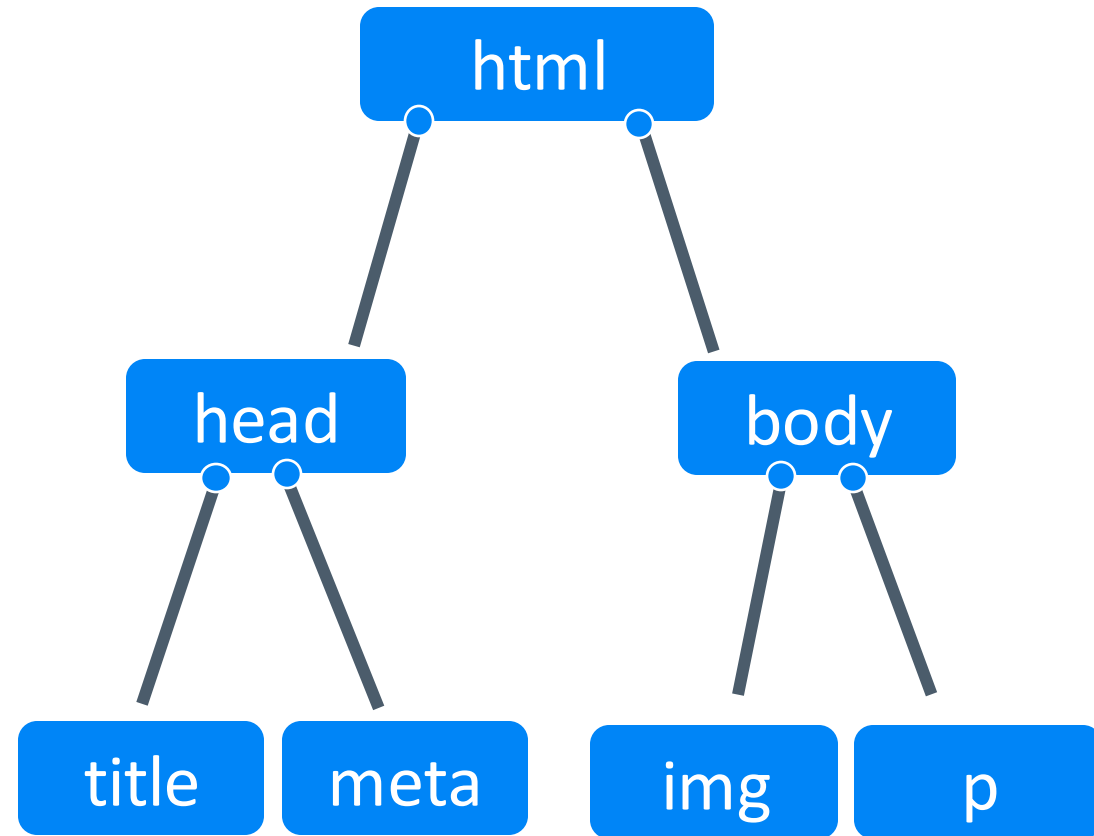
    <p> HTML is fun! </p>

  </body>

</html>
```

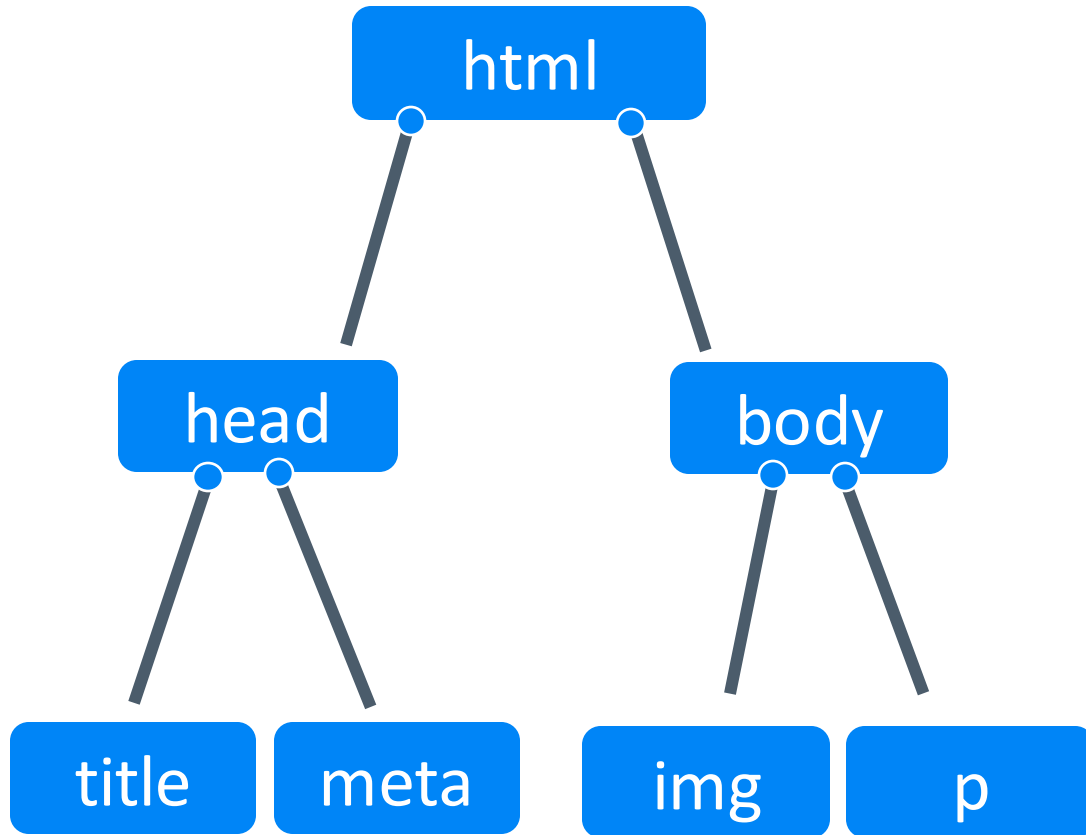
Arbre DOM

(Document Object Model)



Arbre DOM

(Document Object Model)



> Nœud = Boîte

> Nœuds Père | Nœuds fils
→ Boîte dans boîte

> Nœuds frère
→ Comportement dépend de la nature de la balise

BALISE
BLOCK

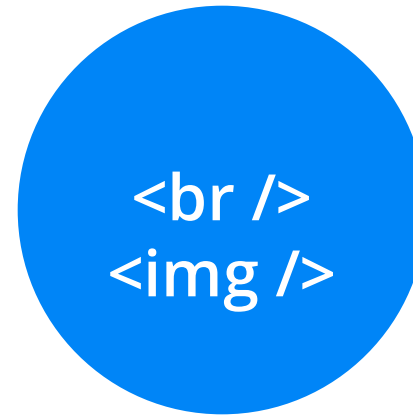
BALISE
INLINE

BALISE
INLINE-
BLOCK

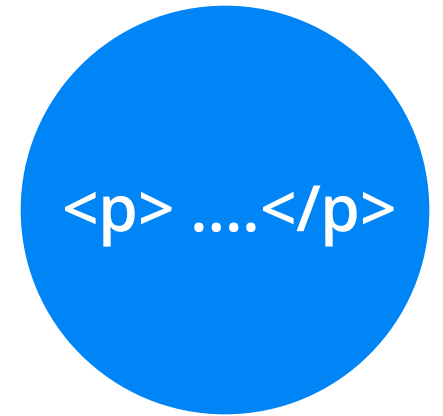
Un aperçu plus clair des balises

Les balises sont utilisées
pour marquer le début et
la fin des éléments HTML

<balise> Contenu **</balise>**

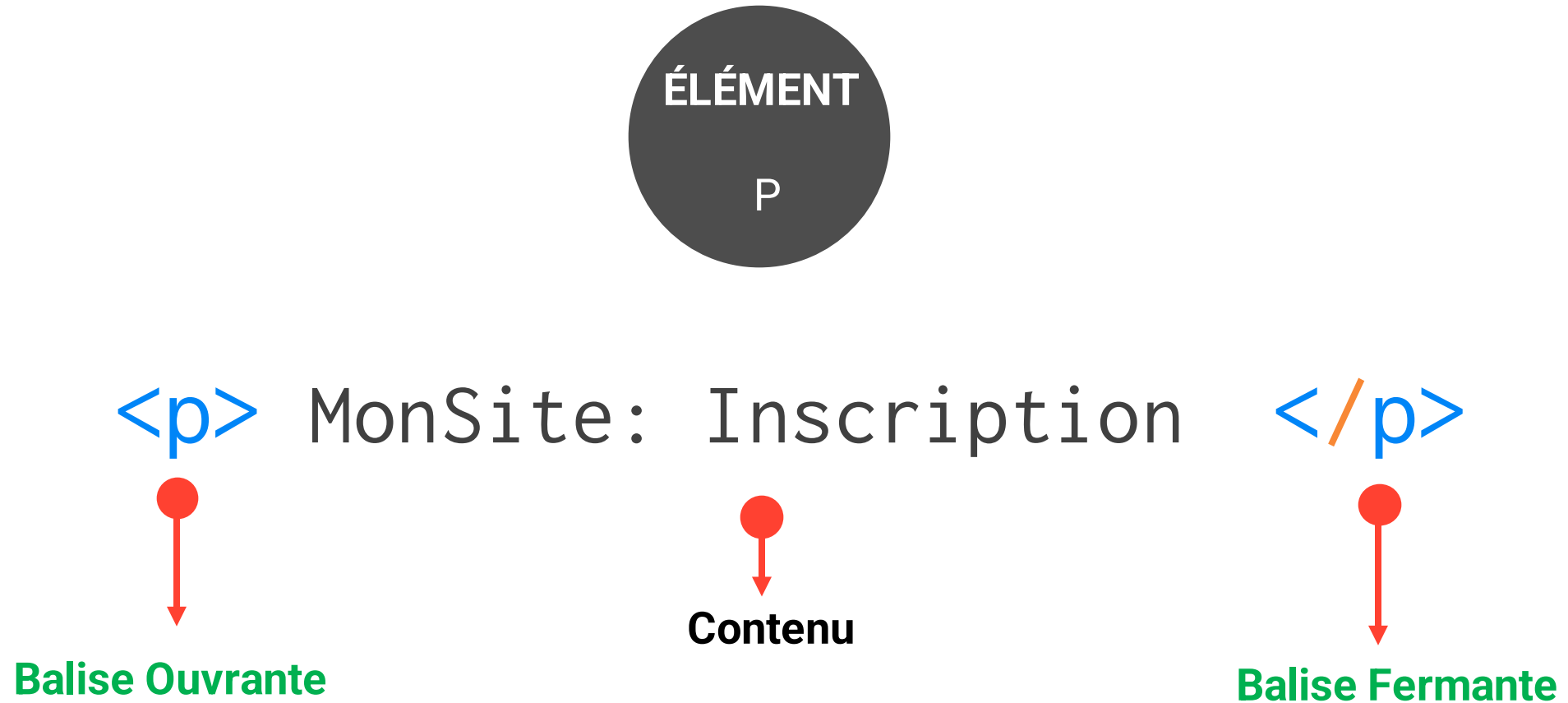


Balise unitaire
Auto fermante



Balise en paire

Balise en paire



La **grande majorité** des balises HTML sont des balises en paires

Balise unitaire

Élément br

Balise unitaire

```
<div>
```

```
  <h1> Contact info </h1>
```

```
  <br/>
```

```
  <p> Université d'Angers </p>
```

```
</div>
```

HTML : Plan général



Sections |
paragraphes



Liens |
Ancres



Images



Listes



Tableaux



Formulaires

HTML : Chapitre 1



Sections |
paragraphes



Liens |
Ancres



Images



Listes



Tableaux



Formulaires

Les balises liées à la structuration



NON
SEMANTIQUE

<div> ... </div>

 ...

Les balises liées au Texte



`<h1> Mon Titre </h1>`

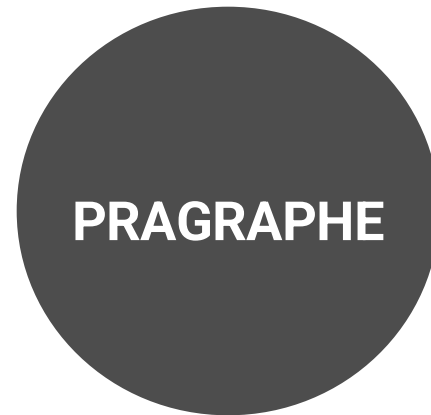
`<h2> Mon Titre </h2>`

...

`<h6> Mon Titre </h6>`

Les balises `<h1>` à `<h6>`

Les balises liées au Texte



<p> Mon Texte </p>

Les balises liées au Texte



FORMATAGE

`<q> Texte </q>` Citation

` Texte ` Texte en gras

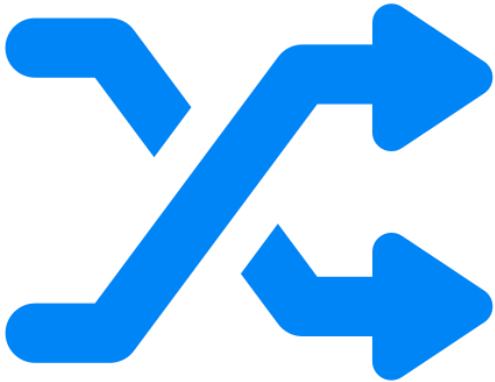
` Texte ` Texte en italique

`_{ind}` Texte en indice

`^{exp}` Texte en exposant

`<blockquote> ... </blockquote>` Block de citation

`<acronym title=""> Abbr </acronym>` Définir un acronyme (HTML5)



Quelques Bonnes pratiques

Les balises doivent être imbriquées, **sans se chevaucher** :

```
<p> This is <strong> correct </strong> </p>
```



```
<p> This is <strong> correct </p> </strong>
```



Quelques Bonnes pratiques

Utiliser l' **indentation** pour mettre en évidence les blocs de code

```
<div>
  <ul>
    <li>
      
      <h3>HTML5</h3>
    </li>
    <li>
      
      <h3>CSS3</h3>
    </li>
  </ul>
</div>
```





Sections |
paragraphes



Liens |
Ancres



Images



Listes



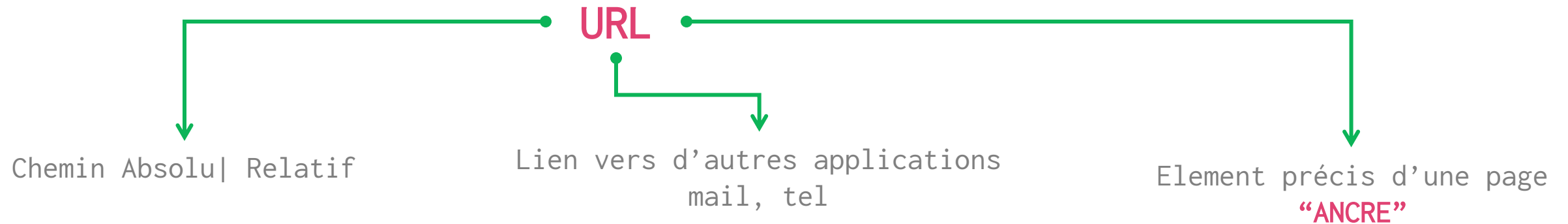
Tableaux



Formulaires

Balise <a> - Lien

` Université d'Angers `

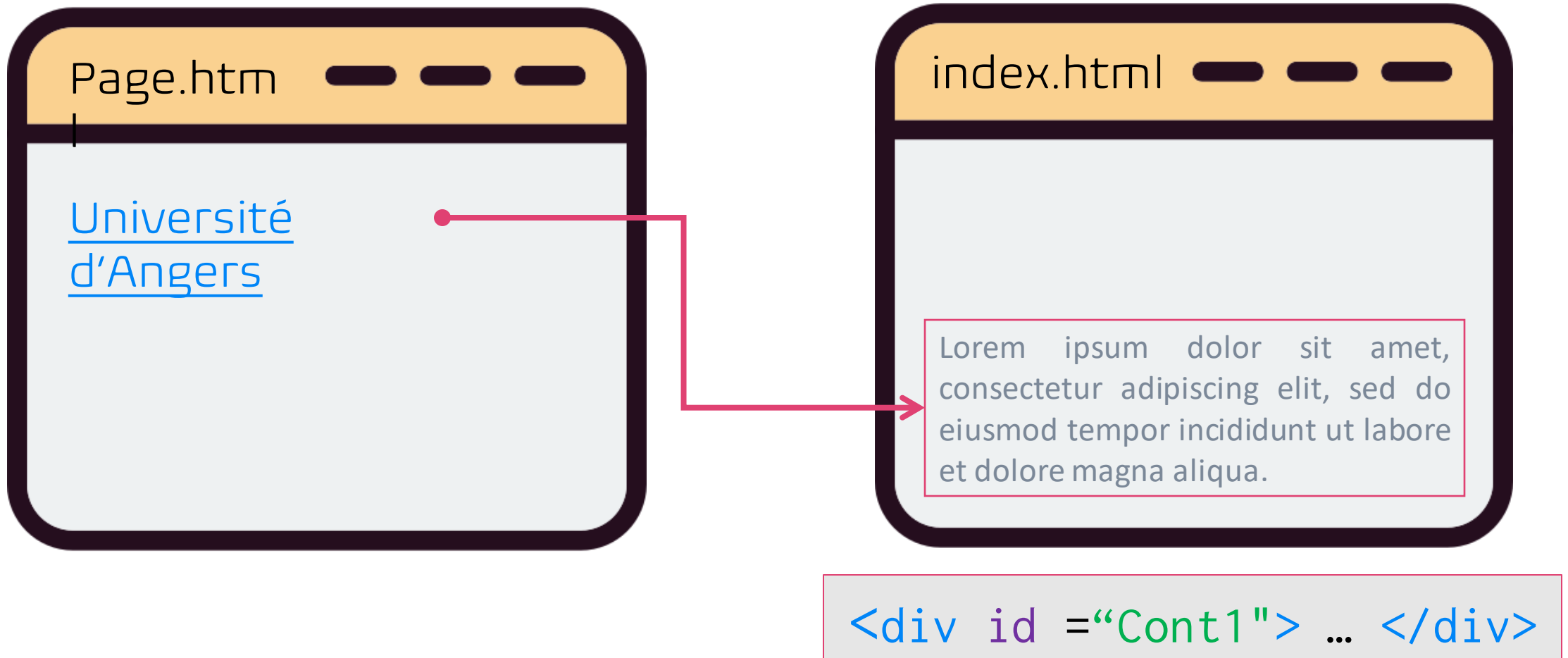


```
<a href = "www.unviv-angers.fr"> UA </a>  
<a href = "Accueil/page.html"> Accueil </a>
```

```
> Contact </a>  
.ephone </a>
```

Balise <a> - Ancre

` Université d'Angers `



Balise <a> - Ancre



```
<div id = "Haut"> ... </div>
```

```
<a href = "#Haut">Haut</a>
```



**Sections |
paragraphes**



**Liens |
Ancres**



Images



Listes



Tableaux

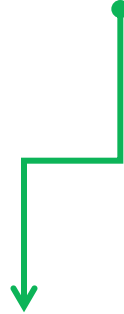


Formulaires

Balise

```

```



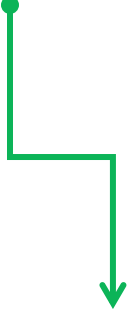
Chemin vers l'image



Texte alternatif



Hauteur



Largeur



**Sections |
paragraphes**



**Liens |
Ancres**



Images



Listes



Tableaux



Formulaires

Les listes en HTML



Les listes simples – non ordonnées

 Elm1

 Elm2

 Elm3

liste non ordonnée

- Elm1
- Elm2
- Elm3

Les listes simples – non ordonnées

Il existe plusieurs styles possibles aux puces d'une liste non-ordonnée, voici quelques possibilités :

- › Disque noir : **list-style-type: disc;**
- › Cercle : **list-style-type: circle;**
- › Carré noir : **list-style-type: square;**
- › Aucune puce : **list-style-type: none;**

Les listes simples – non ordonnées

```
<ul style="list-style-type: disc;">
```

```
  <li> Elm1 </li>
```

```
</ul>
```

```
<ul style="list-style-type: circle;">
```

```
  <li> Elm2 </li>
```

```
</ul>
```

```
<ul style="list-style-type: square;">
```

```
  <li> Elm3 </li>
```

```
</ul>
```

```
<ul style="list-style-type: none;">
```

```
  <li> Elm4 </li>
```

```
</ul>
```

- Elm1
- Elm2
- Elm3
- Elm4

Les listes numérotées – ordonnées

 Elm1

 Elm2

 Elm3

liste ordonnée

1. Elm1
2. Elm2
3. Elm3

Les listes numérotées – ordonnées

Il existe plusieurs styles possibles aux puces d'une liste ordonnée, voici quelques possibilités :

- › Chiffres décimaux : **list-style-type: decimal;**
- › Chiffres décimaux avec un zéro devant : **list-style-type: decimal-leading-zero;**
- › Chiffres romains en majuscules : **list-style-type: upper-roman;**
- › Chiffres romains en minuscules : **list-style-type: lower-roman;**
- › Alphabet en majuscules : **list-style-type: upper-alpha;**
- › Alphabet en minuscules : **list-style-type: lower-alpha;**

Les listes simples – non ordonnées

```
<ul style="list-style-type: decimal-leading-zero">
```

```
  <li> Elm1 </li>
```

```
  <li> Elm2 </li>
```

```
</ul>
```

```
<ul style="list-style-type: upper-roman;">
```

```
  <li> Elm3 </li>
```

```
  <li> Elm4 </li>
```

```
</ul>
```

```
<ul style="list-style-type: upper-alpha;">
```

```
  <li> Elm5 </li>
```

```
  <li> Elm6 </li>
```

```
</ul>
```

01. Elm1

02. Elm2

I. Elm3

II. Elm4

A. Elm5

B. Elm6

Les listes de définition

<dl>

<dt> Elm1 </dt>

<dd> description1 <dd>

<dt> Elm2 </dt>

<dd> description2 <dd>

</dl>

Elm1

description1

Elm2

description2



**Sections |
paragraphes**



**Liens |
Ancres**



Images



Listes



Tableaux



Formulaires

Les tableaux en HTML

Définit un tableau

```
<table> ... </table>
```

quelques
attributs

```
graph TD; A[quelques attributs] --- B[align = left, right, center]; A --- C[border = n]; A --- D[width = n]; A --- E[bgcolor = color];
```

align =
left, right,
center

border = n

width = n

bgcolor = color

Structure tableaux en HTML5

`<table> ... </table>`

Produit	Prix	Quantité	Prix * quantité
Pro1	60\$	2	120\$
Pro2	10\$	32	320\$
Total			440\$

`<thead> ... </thead>`

`<tbody> ... </tbody>`

`<tfoot> ... </tfoot>`

Les tableaux en HTML

Définit une ligne d'un tableau

`<tr> ... </tr>`

quelques
attributs

`align =`
left, right,
center

`valign =`
Top, middle,
bottom

`border = n`

`bgcolor = color`

Les tableaux en HTML

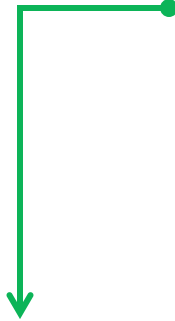
Définit une cellule d'un tableau

```
<td> ... </td>
```

Définit une cellule d'entete d'un tableau

```
<th> ... </th>
```

quelques
attributs



colspan

Fusion de gauche
à droite



rowspan

Fusion de haut
en bas

```

1 <table border="1">
2   <caption>Tableau des produits</caption>
3   <thead>
4     <tr>
5       <th>Produit</th>
6       <th>Prix</th>
7       <th>quantité</th>
8       <th>somme</th>
9     </tr>
10  </thead>
11  <tfoot>
12    <tr>
13      <td>Total</td>
14      <td></td>
15      <td></td>
16      <td>360</td>
17    </tr>
18  </tfoot>
19  <tbody>
20    <tr>
21      <td>Produit1</td>
22      <td>20</td>
23      <td>2</td>
24      <td>40</td>
25    </tr>
26    <tr>
27      <td>Produit2</td>
28      <td>32</td>
29      <td>10</td>
30      <td>320</td>
31    </tr>
32  </tbody>
33 </table>

```

Tableau des produits

Produit	Prix	quantité	somme
Produit1	20	2	40
Produit2	32	10	320
Total			360

Fusionner les cellules

```

1 <table border="1">
2   <caption>Tableau des produits</caption>
3   <thead>
4     <tr>
5       <th>Produit</th>
6       <th>Prix</th>
7       <th>quantité</th>
8       <th>somme</th>
9     </tr>
10  </thead>
11  <tfoot>
12    <tr>
13      <td colspan="3">Total</td>
14      <td>360</td>
15    </tr>
16  </tfoot>
17  <tbody>
18    <tr>
19      <td>Produit1</td>
20      <td>20</td>
21      <td>2</td>
22      <td>40</td>
23    </tr>
24    <tr>
25      <td>Produit2</td>
26      <td>32</td>
27      <td>10</td>
28      <td>320</td>
29    </tr>
30  </tbody>
31 </table>

```

Tableau des produits

Produit	Prix	quantité	somme
Produit1	20	2	40
Produit2	32	10	320
Total			360

```

3 <thead>
4 <tr>
5 <th>Ville</th>
6 <th>Département</th>
7 <th>Code</th>
8 </tr>
9 </thead>
10 <tbody>
11 <tr>
12 <td>Angers</td>
13 <td>Maine-et-loire</td>
14 <td>49</td>
15 </tr>
16 <tr>
17 <td>Cholet</td>*
18 <td>Maine-et-loire</td>
19 <td>49</td>
20 </tr>
21 <tr>
22 <td>Le Mans</td>
23 <td>Sarthe</td>
24 <td>72</td>
25 </tr>
26 <tr>
27 <td>Allonnes</td>
28 <td>Sarthe</td>
29 <td>72</td>
30 </tr>
31 <tr>
32 <td>Changé</td>
33 <td>Sarthe</td>
34 <td>72</td>
35 </tr>
36 </tbody>

```

Tableau des villes

Ville	Département	Code
Angers	Maine-et-loire	49
Cholet	Maine-et-loire	49
Le Mans	Sarthe	72
Allonnes	Sarthe	72
Changé	Sarthe	72

Fusionner les
cellules


```

1 <table border="1">
2   <caption>Tableau des villes</caption>
3   <thead>
4     <tr>
5       <th>Ville</th>
6       <th>Département</th>
7       <th>Code</th>
8     </tr>
9   </thead>
10  <tbody>
11    <tr>
12      <td>Angers</td>
13      <td rowspan="2">Maine-et-loire</td>
14      <td rowspan="2">49</td>
15    </tr>
16    <tr>
17      <td>Cholet</td>
18    </tr>
19    <tr>
20      <td>Le Mans</td>
21      <td rowspan="3">Sarthe</td>
22      <td rowspan="3">72</td>
23    </tr>
24    <tr>
25      <td>Allonnes</td>
26    </tr>
27    <tr>
28      <td>Changé</td>
29    </tr>
30 </table>

```

Tableau des villes

Ville	Département	Code
Angers	Maine-et-loire	49
Cholet		
Le Mans	Sarthe	72
Allonnes		
Changé		



**Sections |
paragraphes**



**Liens |
Ancres**



Images



Listes



Tableaux



Formulaires

Les formulaires en HTML

Définit un formulaire en HTML

`<form> ... </form>`

2 attributs



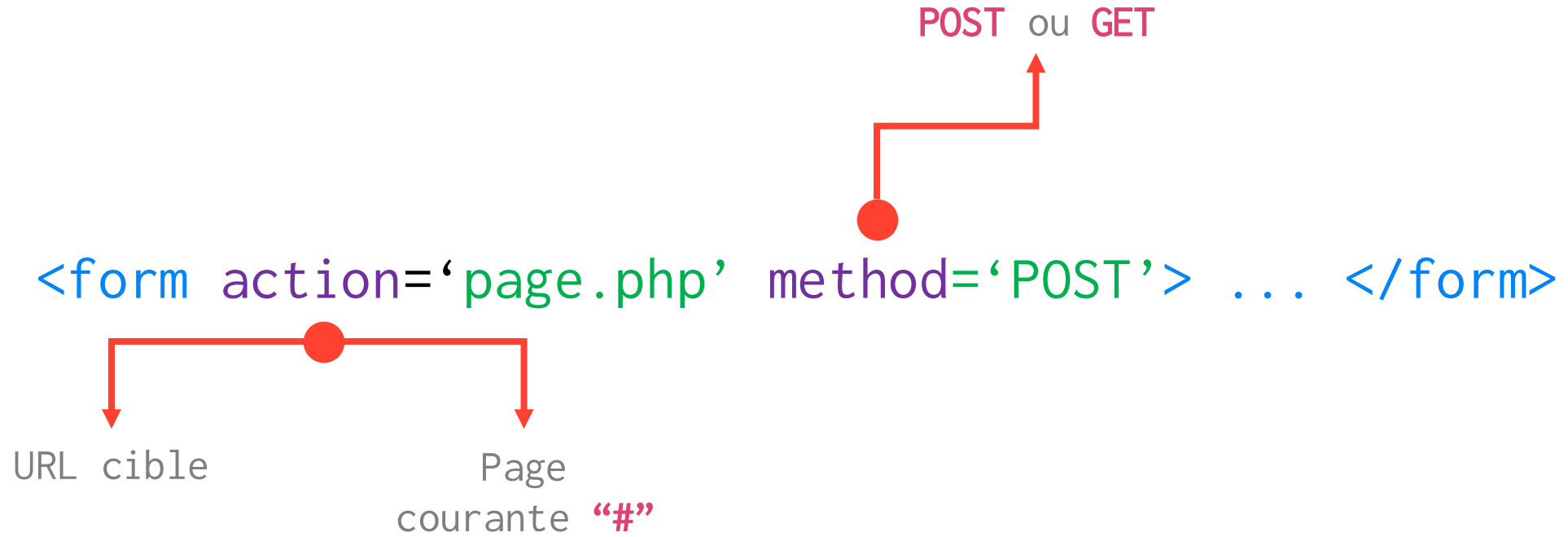
action

URL vers laquelle les
données seront envoyées

method

Méthode HTTP utilisée
pour envoyer les
données vers le serveur

Les formulaire



Éléments d'un formulaire

Il existe plusieurs champs possibles :

- › **champ de texte : en clair ou non (password)**
- › **champ de texte sur plusieurs lignes (textarea)**
- › **liste déroulante (select)**
- › **bouton radio : un seul choix parmi plusieurs**
- › **case à cocher : plusieurs choix possibles**
- › **boutons : simple, submit, reset**

<fieldset> : l'élément pour les ensembles de champs


L'élément HTML **<fieldset>** est utilisé afin de regrouper plusieurs contrôles interactifs ainsi que des étiquettes dans un formulaire HTML.

```
<form action="#" method="GET">  
  <fieldset>  
    <legend> SECTION 1 </legend>  
  </fieldset>  
</form>
```

Définit le titre de la section

<fieldset> : l'élément pour les ensembles de champs

L'élément HTML **<fieldset>** est utilisé afin de regrouper plusieurs contrôles interactifs ainsi que des étiquettes dans un formulaire HTML.



SECTION 1

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>
  </fieldset>
</form>
```

Définit le titre de la section

<input>

Les éléments **<input>** représentent les champs de saisie du formulaire. Ces champs peuvent prendre plusieurs formes et récupérer différents types de données.

- Texte - password
- Button
- Checkbox
- Date
- Radiobox

Champs de texte via <input>

Attributs

- **Type** : text | password
- **Name**: identificateur
- **Value** : valeur par défaut
- **Size** : taille du champs
- **Maxlength**: taille maximale
- **Placeholder** : texte indicatif

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>
    <input type="text" placeholder="Entrez votre Nom">
    <br/><br/>
    <input type="password" >
  </fieldset>
</form>
```

Définit le texte de remplacement

Champs contenu non visible

Champs de texte via `<input>`

Attributs

- **Type** : text | password
- **Name**: identificateur
- **Value** : valeur par défaut
- **Size** : taille du champs
- **Maxlength**: taille maximale
- **Placeholder** : texte indicatif

SECTION 1

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>
    <input type="text" placeholder="Entrez votre Nom">
    <br/><br/>
    <input type="password" >
  </fieldset>
</form>
```

Définit le texte de remplacement

Champs contenu non visible

<label>

L'élément **<label>** représente une légende pour un objet d'une interface utilisateur. Il peut être associé à un contrôle en utilisant l'attribut **for** ou en plaçant l'élément du contrôle à l'intérieur de l'élément **<label>**

```
<form action="#" method="GET">
  <fieldset>
    <input type="text" id="Nom" />

    <label for="Nom"> Nom : </label>
    <input id="Nom" type="text" placeholder="Votre Nom">
    <br/><br/>

    <label> Mdp:
      <input type="password" />
    </label>

  </fieldset>
</form>
```

Définit le champ concerné a travers son id

Requêtes imbriquées

<label>

L'élément **<label>** représente une légende pour un objet d'une interface utilisateur. Il peut être associé à un contrôle en utilisant l'attribut **for** ou en plaçant l'élément du contrôle à l'intérieur de l'élément **<label>**

SECTION 1

Nom :

Mdp:

```
<form action="#" method="GET">
  <fieldset>
    <label for="Nom"> Nom : </label>
    <input id="Nom" type="text" placeholder="Votre Nom">
    <br/><br/>

    <label> Mdp:
      <input type="password" >
    </label>

  </fieldset>
</form>
```

Définit le champ concerné a travers son id

Requêtes imbriquées

Bouton radio via <input>

Attributs

- **Type** : radio
- **Name**: identificateur
- **Value** : valeur
- **checked**: indique si le bouton est coché

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <label> Veuillez choisir une option : </label> <br/><br/>

    <input type="radio" name="Choix" value="Choix1"> Choix1
    <input type="radio" name="Choix" value="Choix2"> Choix2
    <input type="radio" name="Choix" value="Choix3"> Choix3

  </fieldset>
</form>
```

Même nom pour chaque groupe
d'option!!

Bouton radio via <input>

Attributs

- **Type** : radio
- **Name**: identificateur
- **Value** : valeur par défaut
- **checked**: indique si le bouton est coché

SECTION 1

Veillez choisir une option :

☐ Choix1 ☐ Choix2 ☐ Choix3

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <label> Veillez choisir une option : </label> <br/><br/>

    <input type="radio" name="Choix" value="Choix1"> Choix1
    <input type="radio" name="Choix" value="Choix2"> Choix2
    <input type="radio" name="Choix" value="Choix3"> Choix3

  </fieldset>
</form>
```

Même nom pour chaque groupe d'options!!

Bouton choix multiples via <input>

Attributs

- **Type** : checkbox
- **Name**: identificateur
- **Value** : valeur
- **checked**: indique si le bouton est coché

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <label> Veuillez cochez vos choix : </label> <br/><br/>

    <input type="checkbox" name="Choix1" value="Style1"> Style1
    <input type="checkbox" name="Choix2" value="Style2"> Style2
    <input type="checkbox" name="Choix3" value="Style2"> Style3

  </fieldset>
</form>
```

Bouton choix multiples via <input>

Attributs

- **Type** : checkbox
- **Name**: identificateur
- **Value** : valeur par défaut
- **checked**: indique si le bouton est coché

SECTION 1

Veillez cochez vos choix :

☐ Style1 ☐ Style2 ☐ Style3

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <label> Veillez cochez vos choix : </label> <br/><br/>

    <input type="checkbox" name="Choix1" value="Style1"> Style1
    <input type="checkbox" name="Choix2" value="Style2"> Style2
    <input type="checkbox" name="Choix3" value="Style2"> Style3

  </fieldset>
</form>
```


Bouton via <input>

Attributs

- **Type** : submit | reset
- **Name**: identificateur
- **Value** : valeur

Envoie le formulaire vers le serveur

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>
    <label> réinitialise les champs du formulaire : </label> <br/><br/>
    <input type="reset" name="vider" value="Réinitialiser">
    <input type="submit" name="Valider" value="Valider">
    <input type="button" name="perso" value="A personnaliser!!">
  </fieldset>
</form>
```

Aucune Action!! A définir via JS

On peut aussi utiliser la balise

<button>

Bouton via <input>

Attributs

- **Type** : submit | reset
- **Name**: identificateur
- **Value** : valeur par défaut

réinitialise les champs du formulaire

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>
    <label> Envoie le formulaire vers le serveur : </label> <br/><br/>
    <input type="reset" name="vider" value="Réinitialiser">
    <input type="submit" name="Valider" value="Valider">
    <input type="button" name="perso" value="A personnaliser!!">
  </fieldset>
</form>
```

Aucune Action!! A définir via JS

On peut aussi utiliser la balise

<button>

SECTION 1

Veuillez cochez vos choix :

Réinitialiser

Valider

A personnaliser!!

Liste déroulante via <select>

<select> représente un
contrôle qui fournit une liste
d'options parmi lesquelles
l'utilisateur pourra choisir.

<select>

<option>

<option>

</select>

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <select id="monselect">
      <option value="valeur1">Valeur 1</option>
      <option value="valeur2" selected>Valeur 2</option>
      <option value="valeur3">Valeur 3</option>
    </select>

  </fieldset>
</form>
```

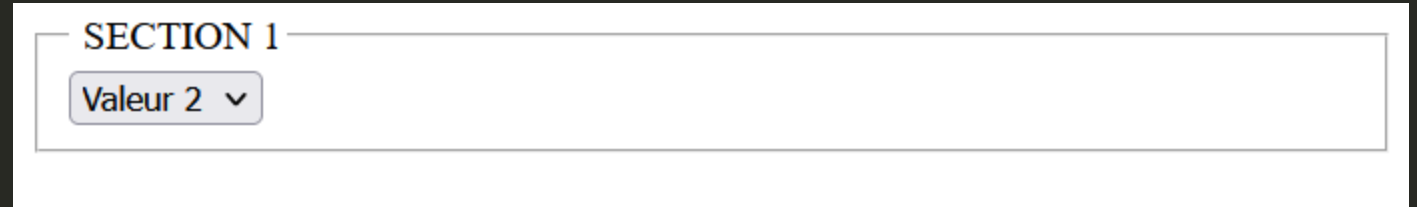
Option : définit l'ensemble des éléments de la liste

Selected: select une valeur par défaut

Liste déroulante via `<select>`

`<select>` représente un contrôle qui fournit une liste d'options parmi lesquelles l'utilisateur pourra choisir.

```
<select>
  <option>
  <option>
</select>
```

A screenshot of a web form. It features a legend titled "SECTION 1" with a horizontal line. Below the legend, there is a dropdown menu (select control) with the text "Valeur 2" and a downward-pointing arrow.

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <select id="monselect">
      <option value="valeur1">Valeur 1</option>
      <option value="valeur2" selected>Valeur 2</option>
      <option value="valeur3">Valeur 3</option>
    </select>

  </fieldset>
</form>
```

Option : définit l'ensemble des éléments de la liste

Selected: select une valeur par défaut

Liste déroulante option <optgroup>

<optgroup> permet de créer un groupe d'options parmi lesquelles on peut choisir dans un élément <select>.

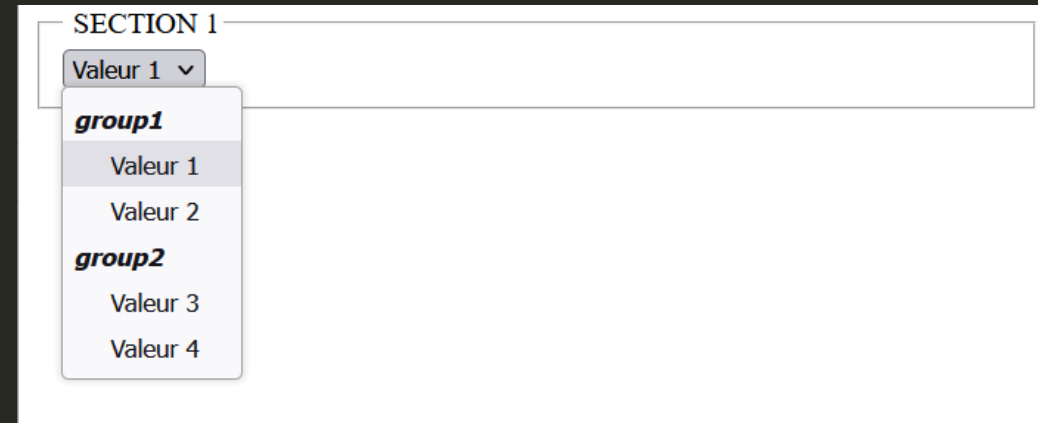
```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <select id="monselect">
      <optgroup label="group1">
        <option value="valeur1">Valeur 1</option>
        <option value="valeur2">Valeur 2</option>
      </optgroup>
      <optgroup label="group2">
        <option value="valeur3">Valeur 3</option>
        <option value="valeur4">Valeur 4</option>
      </optgroup>
    </select>

  </fieldset>
</form>
```

Liste déroulante option <optgroup>

<optgroup> permet de créer un groupe d'options parmi lesquelles on peut choisir dans un élément <select>.



The screenshot shows a web form with a legend titled "SECTION 1". Below the legend is a dropdown menu. The menu is currently open, showing a list of options. The options are grouped into two categories: "group1" and "group2". "group1" contains "Valeur 1" and "Valeur 2". "group2" contains "Valeur 3" and "Valeur 4". The dropdown menu is styled with a light blue background and a white border. The legend is a simple box with a title and a border.

```
<form action="#" method="GET">
  <fieldset>
    <legend> SECTION 1 </legend>

    <select id="monselect">
      <optgroup label="group1">
        <option value="valeur1">Valeur 1</option>
        <option value="valeur2">Valeur 2</option>
      </optgroup>
      <optgroup label="group2">
        <option value="valeur3">Valeur 3</option>
        <option value="valeur4">Valeur 4</option>
      </optgroup>
    </select>

  </fieldset>
</form>
```



CSS

Aïcha Bakki



aicha.bakki@univ-angers.fr



Application Web : technologies

Client

HTML



Markup Language
Content

CSS



Style Sheet Language
Presentation

JS



Programming Language
Behavior



CSS : Objectif

Séparer la structure d'un document de ses styles de présentation



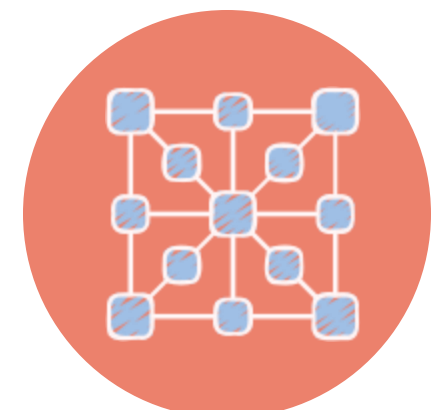
Gestion séparée



Présentation uniformisée



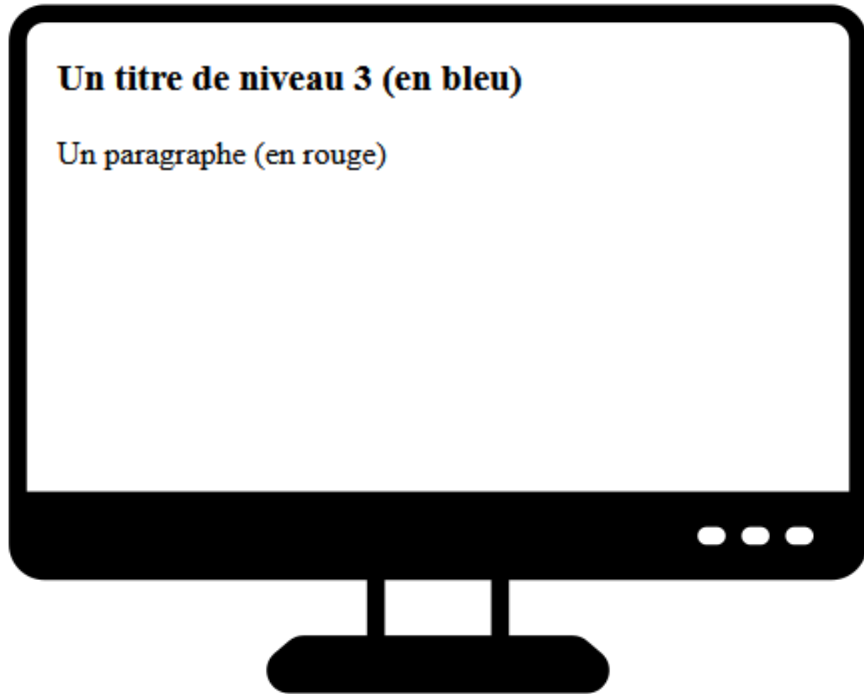
**Choix: 1 CSS
plusieurs supports**



**Réduire la
complexité du HTML**

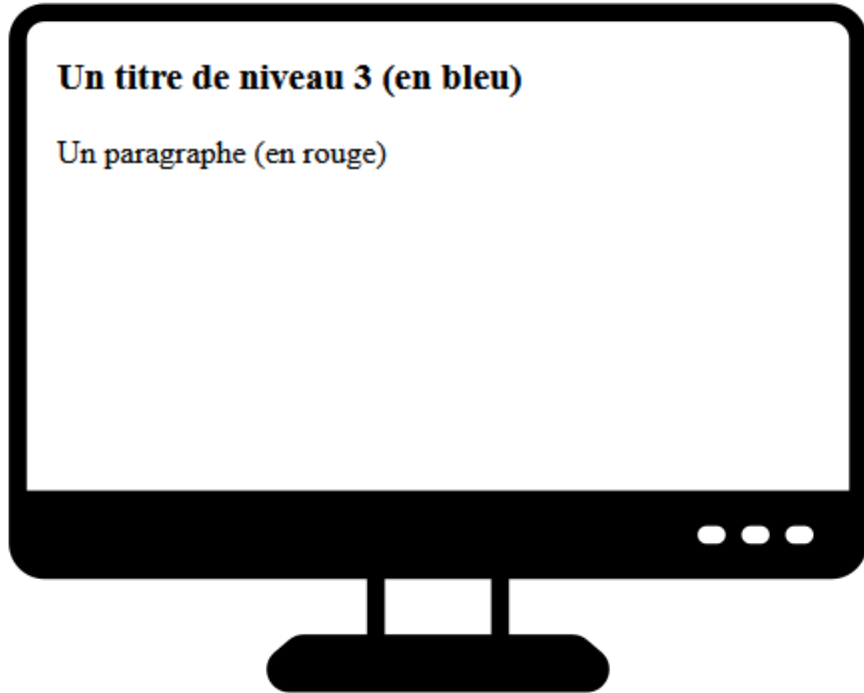


CSS : Principes





CSS : Principes





CSS : Principes

Insérer le style dans les
balises HTML

Insérer le style dans une
balise <style>

Créer une **feuille CSS**
séparée



CSS : Principes

Insérer le style dans les
balises HTML

```
<html>
  <head>
    <title>Un titre de document</title>
  </head>
  Argument style
  <body>
    <h3 style="color: blue;">Un titre de niveau 3 (en bleu)</h3>
    <p style="color: red;">Un paragraphe (en rouge)</p>
  </body>
  Liste des propriétés
</html>
```



CSS : Principes

```
<html>
  <head>
    <title>Un titre de document</title>
    <style type="text/css">
      h3 { color: blue;}
      p { color: red; }
    </style>
  </head>
  <body>
    <h3>Un titre de niveau 3 (en bleu)</h3>
    <p>Un paragraphe (en rouge)</p>
  </body>
</html>
```

Balise <style>

Style à appliquer



CSS : Principes

```
<html>
  <head>
    <title>Un titre de document</title>
    <link rel="stylesheet" type="text/css" href="style.css"/>
  </head>
  <body>
    <h3>Un titre de niveau 3 (en bleu)</h3>
    <p>Un paragraphe (en rouge)</p>
  </body>
</html>
```

Balise <link> + lien vers feuille CSS

```
h3 {
  color: blue;
}
```

```
p {
  color: red;
}
```

Feuille CSS « style.css »



CSS : jeu de règles

Règle CSS

Sélecteur

`h3` {

`color: blue;`

Déclaration

`font-family: "Arial Black", Arial;`

propriété

Valeur

}



CSS : jeu de règles

Règle CSS

Sélecteur

h3

{

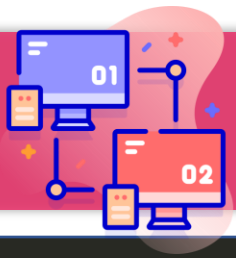
color: blue;

font-family : " Arial Black " , Arial ;

}

propriété

Valeur



CSS : les sélecteurs

Tous les éléments de la page

Sélecteur universel

```
*{  
  color: green;  
  font-family : " Arial Black " , Arial ;  
}
```

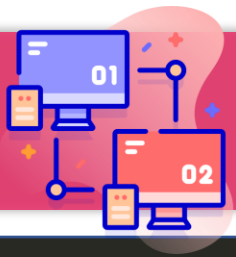
Un titre de niveau 3

Un premier paragraphe

Un deuxième paragraphe

Un troisième paragraphe Une balise SPAN imbriquée

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p>Un premier paragraphe</p>  
  <p>Un deuxième paragraphe</p>  
  <p>Un troisième paragraphe  
    <span> Une balise SPAN imbriquée </span>  
  </p>  
</body>
```



CSS : les sélecteurs

Toutes les instances d'un élément

Tous les pararaphes

```
p {  
  color: red;  
  font-family : " Arial Black " , Arial ;  
}
```

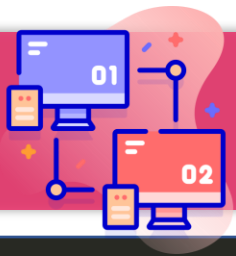
Un titre de niveau 3

Un premier paragraphe

Un deuxième paragraphe

Un troisième paragraphe Une balise SPAN imbriquée

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p>Un premier paragraphe</p>  
  <p>Un deuxième paragraphe</p>  
  <p>Un troisième paragraphe  
    <span> Une balise SPAN imbriquée </span>  
  </p>  
</body>
```



CSS : les sélecteurs

A différents éléments
simultanément

Balises h3 et h4 et span

```
h3, h4, span {  
  color: purple;  
  font-family : " Arial Black " , Arial ;  
}
```

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p>Un premier paragraphe</p>  
  <p>Un deuxième paragraphe</p>  
  <h4>Un titre de niveau 4 </h4>  
  <p>Un troisième paragraphe  
    <span> Une balise SPAN imbriquée </span>  
  </p>  
</body>
```

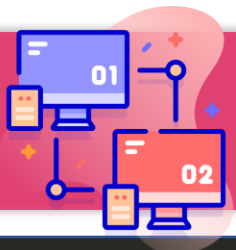
Un titre de niveau 3

Un premier paragraphe

Un deuxième paragraphe

Un titre de niveau 4

Un troisième paragraphe Une balise SPAN imbriquée



CSS : les sélecteurs

A un sous élément
sélecteur (directement
ou indirectement)
descendant

Balise span descendante de p

```
p span {  
    color: purple;  
    font-family : " Arial Black " , Arial ;  
}
```

Un titre de niveau 3

Un premier paragraphe *Une 1ère balise SPAN imbriquée*

Un titre de niveau 4

Un troisième paragraphe *Une 2ème balise SPAN imbriquée*

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p> Un premier paragraphe  
    <em>  
      <span> Une 1ère balise SPAN imbriquée </span>  
    </em>  
  </p> Descendant indirect  
  <h4>Un titre de niveau 4 </h4>  
  <p class="vert">Un troisième paragraphe  
    <span> Une 2ème balise SPAN imbriquée </span>  
  </p> Descendant direct  
</body>
```



CSS : les sélecteurs

A un élément **adjacent**
(frère)

Balise span frère (adjacent) de p

```
p+span {  
    color: purple;  
    font-family : " Arial Black " , Arial ;  
}
```

Un titre de niveau 3

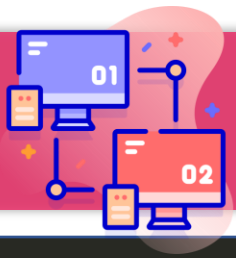
Un premier paragraphe

Un premier span

Un titre de niveau 4

Un troisième paragraphe Une balise SPAN imbriquée

```
<body>  
    <h3>Un titre de niveau 3 </h3>  
    <p>Un premier paragraphe</p>  
    <span>Un premier span</span>  
    <h4>Un titre de niveau 4 </h4>  
    <p>Un troisième paragraphe  
        <span> Une balise SPAN imbriquée </span>  
    </p>  
</body>
```



CSS : les sélecteurs

Au selecteur d'id ou de classe

Sélecteur balise avec id = « para1 »

```
#para1 {  
    color: purple;  
    font-family : " Arial Black " , Arial ;  
}
```

```
<body>    Définition de balise avec id=« para 1»  
    <h3>Un titre de niveau 3 </h3>  
    <p id='para1'>Un premier paragraphe</p>  
    <p>Un deuxième paragraphe</p>  
    <h4>Un titre de niveau 4 </h4>  
    <p class="vert">Un troisième paragraphe  
        <span> Une balise SPAN imbriquée </span>  
    </p>  
</body>
```

Un titre de niveau 3

Un premier paragraphe

Un deuxième paragraphe

Un titre de niveau 4

Un troisième paragraphe Une balise SPAN imbriquée



CSS : les sélecteurs

Au selecteur d'id ou de classe

Sélecteur balise avec class = « vert »

```
.vert {  
    color: green;  
}
```

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p id='para1'>Un premier paragraphe</p>  
  <p class="vert">Un deuxième paragraphe</p>  
  <h4>Un titre de niveau 4 </h4>  
  <p class="vert">Un troisième paragraphe  
    <span> Une balise SPAN imbriquée </span>  
  </p>  
</body>
```

Définition des balises avec
class=« vert »

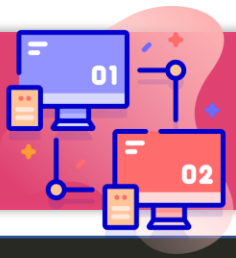
Un titre de niveau 3

Un premier paragraphe

Un deuxième paragraphe

Un titre de niveau 4

Un troisième paragraphe Une balise SPAN imbriquée



CSS : les sélecteurs

Au selecteur d'id ou de classe

Sélecteur balise avec class = « vert »

```
.vert {  
    color: green;  
}
```



- › Possibilité d'avoir plusieurs balises avec le même nom de classe
- › L'ID doit être unique

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p id='para1'>Un premier paragraphe</p>  
  <p class="vert">Un deuxième paragraphe</p>  
  <h4>Un titre de niveau 4 </h4>  
  <p class="vert">Un troisième paragraphe  
    <span> Une balise SPAN imbriquée </span>  
  </p>  
</body>
```

Définition des balises avec
class=« vert »



CSS : les sélecteurs

A tout élément descendant direct d'id de balise ou de classe

Span descendant direct de p

```
p>span {  
    color: purple;  
}
```

Un titre de niveau 3

Un premier paragraphe *Une 1ère balise SPAN imbriquée*

Un titre de niveau 4

Un troisième paragraphe *Une 2ème balise SPAN imbriquée*

```
<body>  
  <h3>Un titre de niveau 3 </h3>  
  <p> Un premier paragraphe  
    <em>  
      <span> Une 1ère balise SPAN imbriquée </span>  
    </em>  
  </p>  
  <h4>Un titre de niveau 4 </h4>  
  <p class="vert">Un troisième paragraphe  
    <span> Une 2ème balise SPAN imbriquée </span>  
  </p>  
</body>
```

Descendant indirect

Descendant direct



CSS : les sélecteurs

Au pseudo-classe d'une balise

Une **pseudo-classe** ou **pseudo-élément** est un mot-clé qui peut être ajouté à un sélecteur afin d'indiquer l'état spécifique dans lequel l'élément doit être pour être ciblé par la déclaration

sélecteur : **pseudo-classe** { propriété: valeur; }

- > **first-child** : premier enfant
- > **link, visited** : s'applique au lien et ancre
- > **hover** : élément survolé par la souris
- > **active** : élément activé par clique
- > **focus** : entrée au clavier par exemple
- > **checked** : élément sélection d'un bouton radio ou case à cocher
- > **first-letter** : première lettre d'un block
- > **first-line** : première ligne d'un block
- > **nth-child()** : permet de cibler tous les n-ième enfants dans la liste des enfants directs



CSS : les sélecteurs

Au pseudo-classe d'une balise

Une **pseudo-classe** ou **pseudo-élément** est un mot-clé qui peut être ajouté à un sélecteur afin d'indiquer l'état spécifique dans lequel l'élément doit être pour être ciblé par la déclaration

```
a : visited {  
    color: green;  
}  
p : first-letter {  
    color : red;  
    font-weight : normal, bold  
}
```

CSS : les sélecteurs

pseudo-élément:
last-child | **first-child**

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

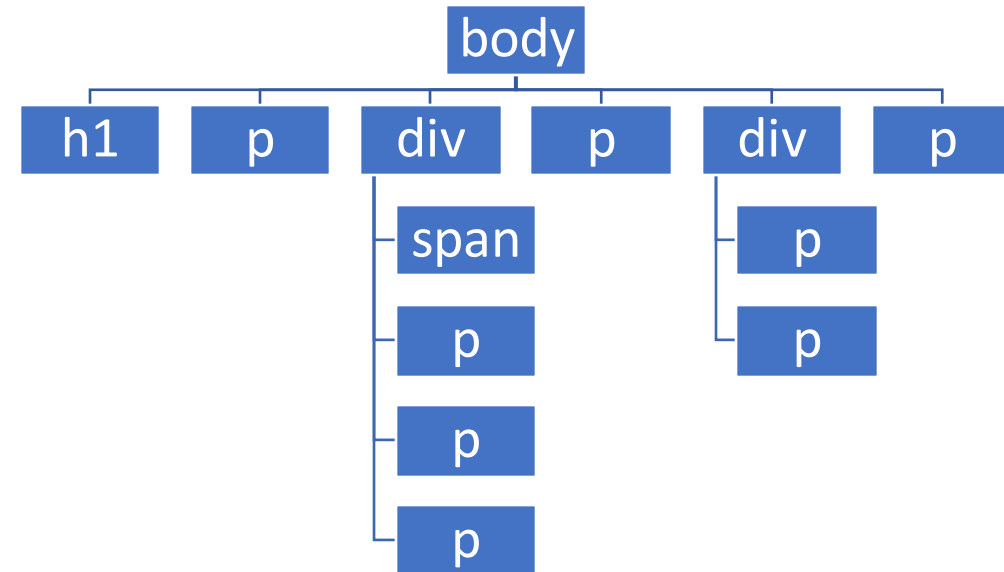
```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

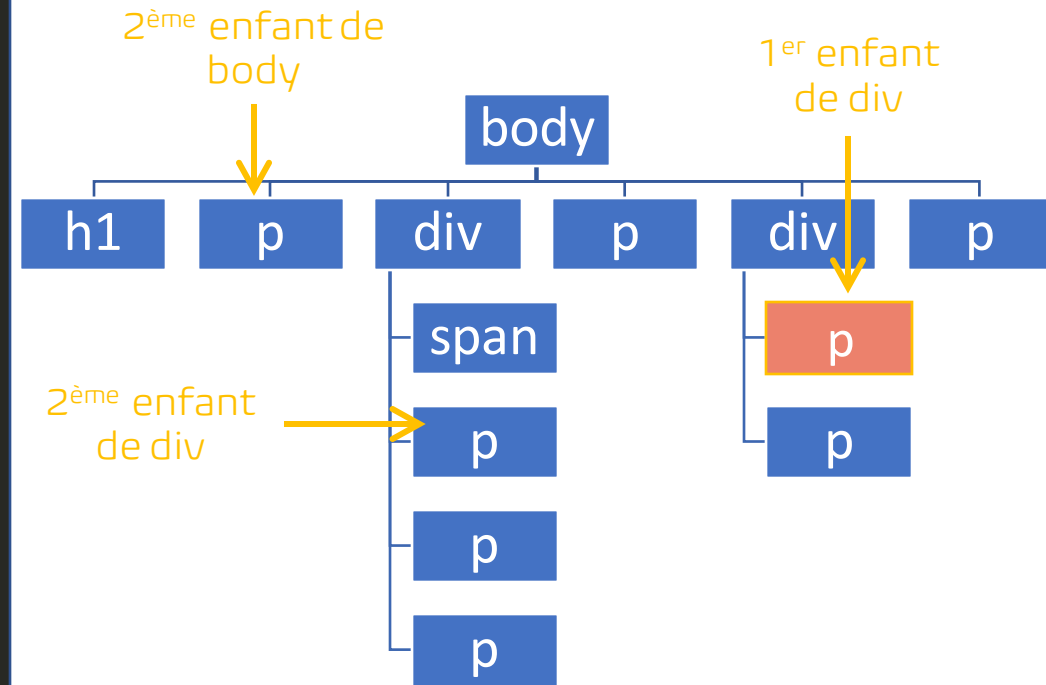


CSS : les sélecteurs

Toute balise <p> première
fille d'un élément de la page

```
p:first-child{  
  color: orange;  
  text-decoration: underline;  
}  
  
p:last-child{  
  color: green;  
  text-decoration: underline;  
}
```

pseudo-élément:
last-child | first-child





CSS : les sélecteurs

pseudo-élément:
last-child | **first-child**

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
  <span>Ce span est le premier enfant du 1er div</span>
```

```
  <p>Premier élément p dans le 1er div</p>
```

```
  <p>Deuxième élément p du div</p>
```

```
  <p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

Les pseudo-classes

Ce paragraphe est le premier élément p du body

Ce span est le premier enfant du 1er div

Premier élément p dans le 1er div

Deuxième élément p du div

Ce paragraphe est le dernier enfant du 1er div

Encore un paragraphe

Ce paragraphe est le premier enfant du 2è div

Ce paragraphe est le dernier enfant du 2è div

Ce paragraphe est le dernier élément du body

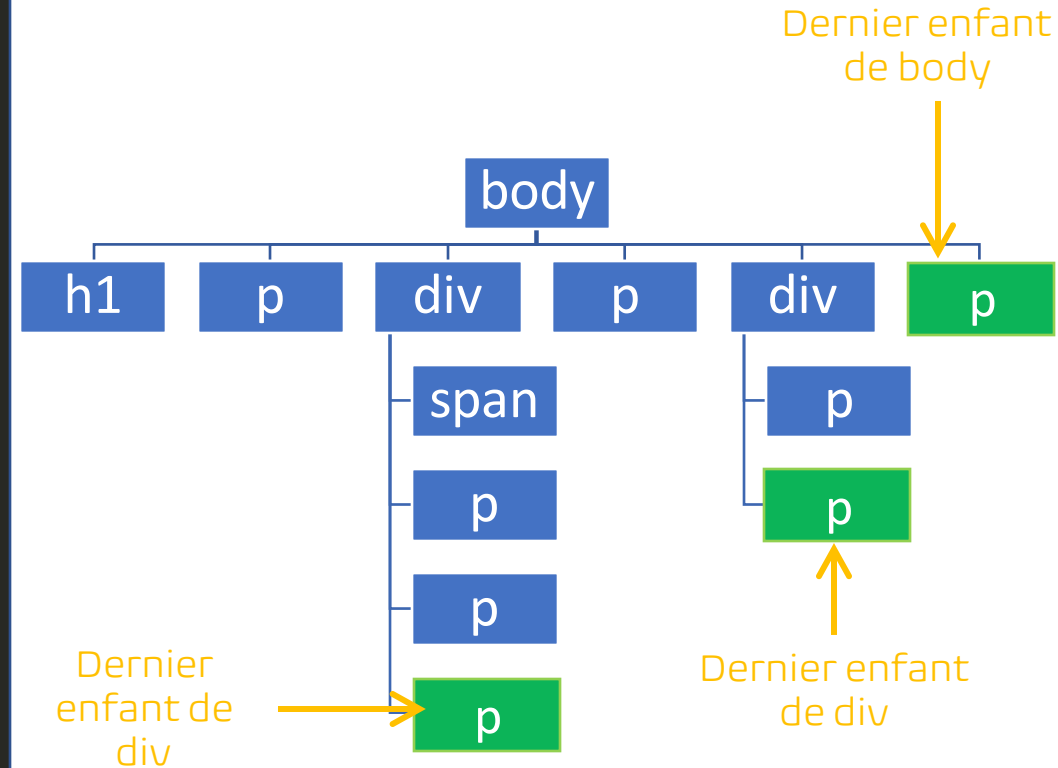
CSS : les sélecteurs

```
p:first-child{  
  color: orange;  
  text-decoration: underline;  
}
```

```
p:last-child{  
  color: green;  
  text-decoration: underline;  
}
```

Toute balise <p> dernière
fille d'un élément de la page

pseudo-élément:
last-child | **first-child**





CSS : les sélecteurs

pseudo-élément:
last-child | **first-child**

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

Les pseudo-classes

Ce paragraphe est le premier élément p du body

Ce span est le premier enfant du 1er div

Premier élément p dans le 1er div

Deuxième élément p du div

Ce paragraphe est le dernier enfant du 1er div

Encore un paragraphe

Ce paragraphe est le premier enfant du 2è div

Ce paragraphe est le dernier enfant du 2è div

Ce paragraphe est le dernier élément du body



CSS : les sélecteurs

Au pseudo-classe d'une balise

Pseudo-classe **nth-child()**. Plusieurs paramètres:

- odd | even
- valeur numérique
- an+b

```
p : nth-child(odd)
    color: green;
}
```

Toute balise <p> fille
impaire d'un élément de la
page

```
p : nth-child(even)
    color: purple;
}
```

Toute balise <p> fille paire
d'un élément de la page

CSS : les sélecteurs

nth-child(odd)
nth-child(even)

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

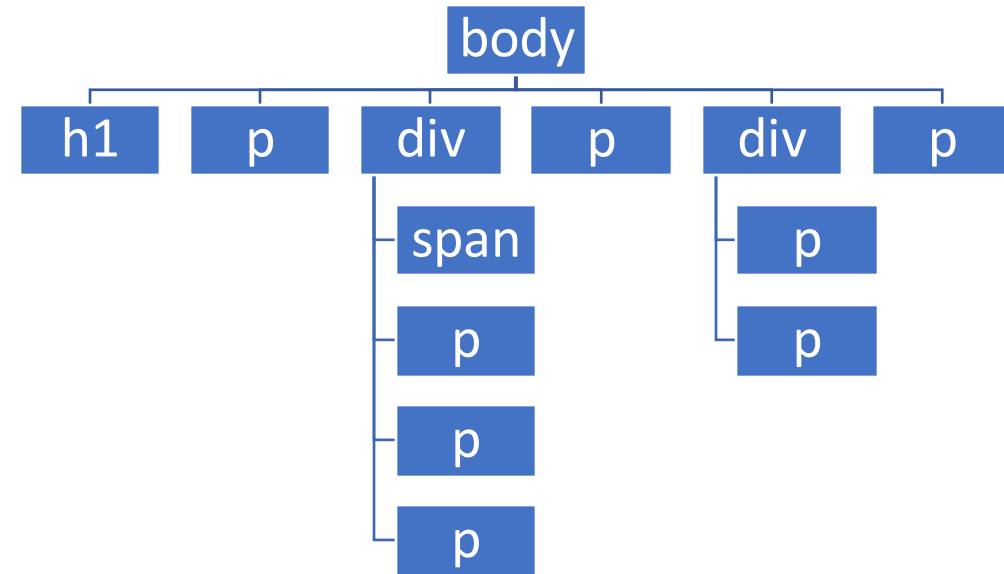
```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

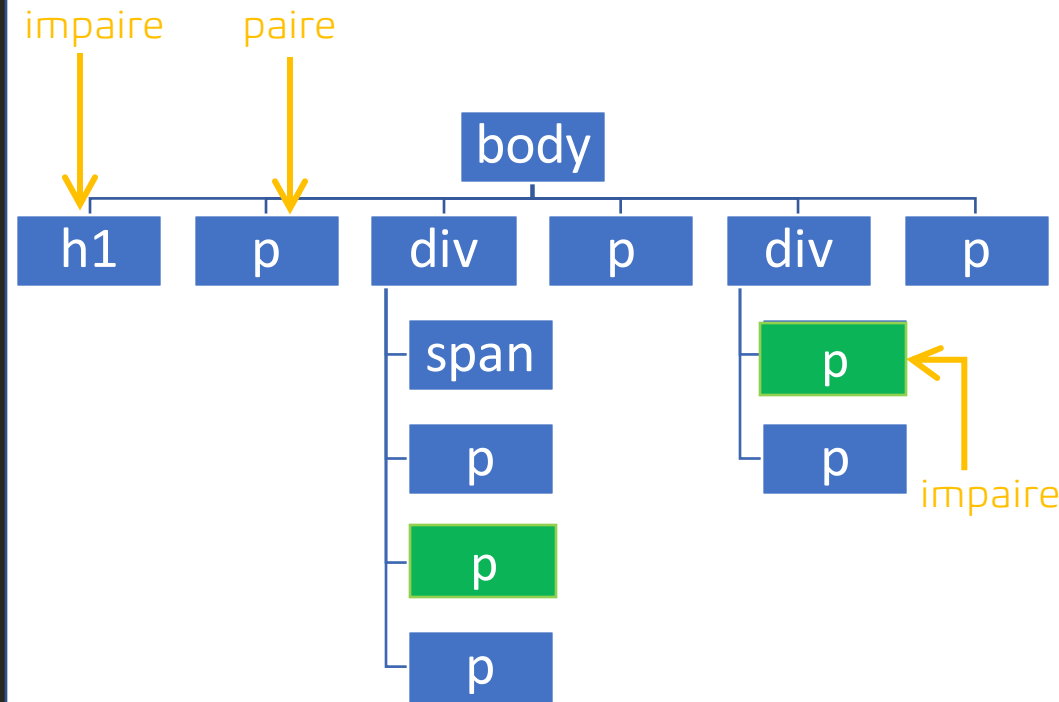


CSS : les sélecteurs

Toute balise <p> fille impaire d'un élément de la page

```
p:nth-child(odd){ /*impaire*/  
  color: green;  
  text-decoration: underline wavy;  
}  
  
p:nth-child(even){ /*paire*/  
  color: purple;  
  text-decoration: underline dotted;  
}
```

nth-child(odd)
nth-child(even)





CSS : les sélecteurs

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

nth-child(odd)
nth-child(even)

Les pseudo-classes

Ce paragraphe est le premier élément p du body

Ce span est le premier enfant du 1er div

Premier élément p dans le 1er div

Deuxième élément p du div

Ce paragraphe est le dernier enfant du 1er div

Encore un paragraphe

Ce paragraphe est le premier enfant du 2è div

Ce paragraphe est le dernier enfant du 2è div

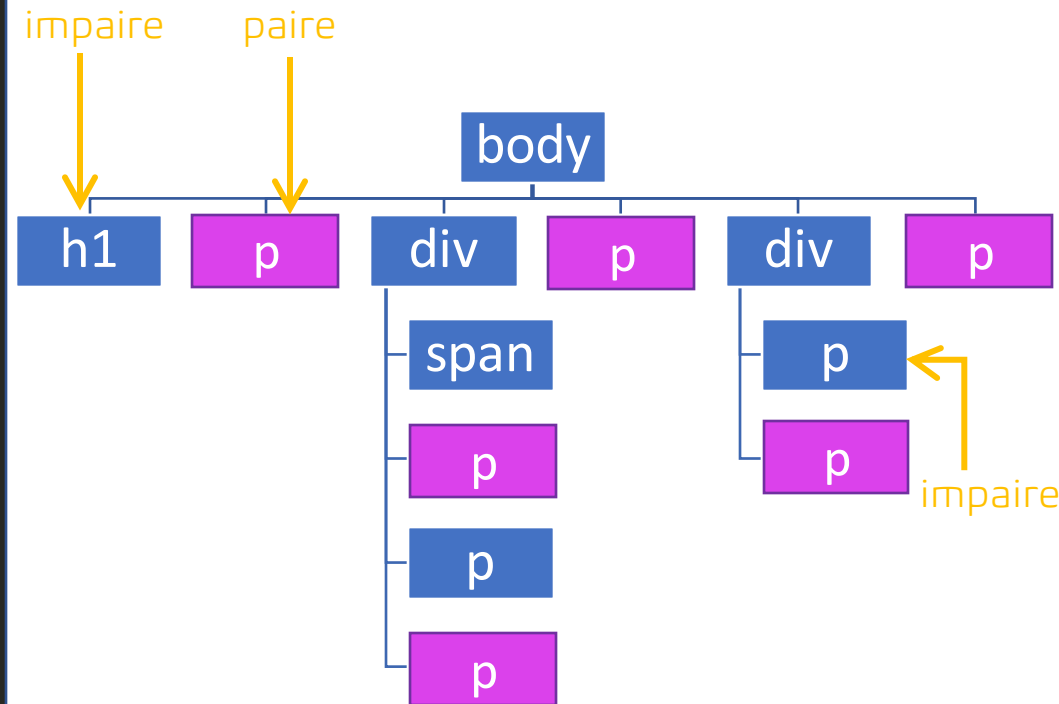
Ce paragraphe est le dernier élément du body

CSS : les sélecteurs

```
p:nth-child(odd){ /*impaire*/  
  color: green;  
  text-decoration: underline wavy;  
}  
  
p:nth-child(even){ /*paire*/  
  color: purple;  
  text-decoration: underline dotted;  
}
```

Toute balise <p> fille paire d'un élément de la page

nth-child(odd)
nth-child(even)





CSS : les sélecteurs

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

nth-child(odd)
nth-child(even)

Les pseudo-classes

Ce paragraphe est le premier élément p du body

Ce span est le premier enfant du 1er div

Premier élément p dans le 1er div

Deuxième élément p du div

Ce paragraphe est le dernier enfant du 1er div

Encore un paragraphe

Ce paragraphe est le premier enfant du 2è div

Ce paragraphe est le dernier enfant du 2è div

Ce paragraphe est le dernier élément du body



CSS : les sélecteurs

Au pseudo-classe d'une balise

Pseudo-classe **nth-child()**. Plusieurs paramètres:

- odd | even
- valeur numérique
- an+b

```
p : nth-child(3)  
    color: green;  
}
```

Toute balise <p> 3^{ème} fille
d'un élément de la page

```
p : nth-child(2)  
    color: purple;  
}
```

Toute balise <p> 2^{ème} fille
d'un élément de la page



CSS : les sélecteurs

nth-child(n)

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

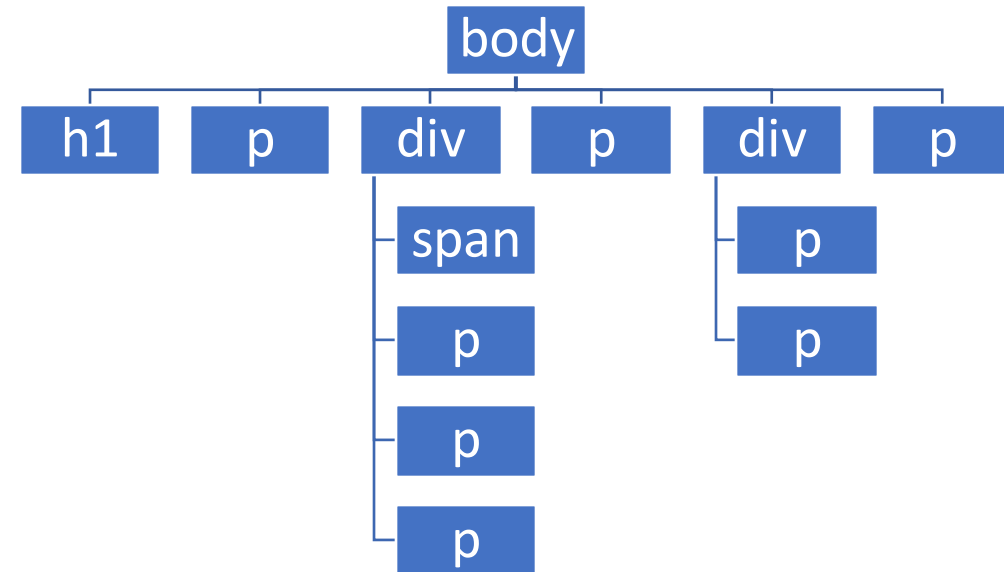
```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```



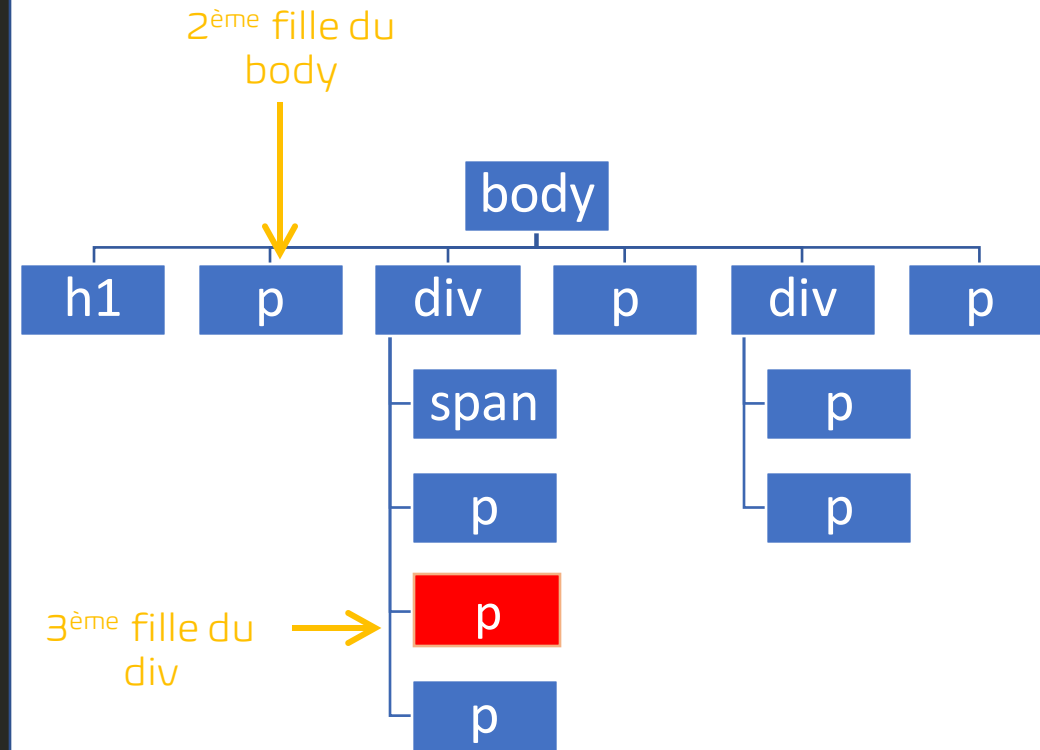


CSS : les sélecteurs

Toute balise <p> 3^{ème} fille d'un élément de la page

```
p:nth-child(3){  
  color: red;  
  text-decoration: underline wavy;  
}
```

nth-child(n)





CSS : les sélecteurs

```
<body>
```

```
<h1>Les pseudo-classes</h1>
```

```
<p>Ce paragraphe est le premier élément p du body</p>
```

```
<div>
```

```
<span>Ce span est le premier enfant du 1er div</span>
```

```
<p>Premier élément p dans le 1er div</p>
```

```
<p>Deuxième élément p du div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 1er div</p>
```

```
</div>
```

```
<p>Encore un paragraphe</p>
```

```
<div>
```

```
<p>Ce paragraphe est le premier enfant du 2è div</p>
```

```
<p>Ce paragraphe est le dernier enfant du 2è div</p>
```

```
</div>
```

```
<p>Ce paragraphe est le dernier élément du body</p>
```

```
</body>
```

nth-child(n)

Les pseudo-classes

Ce paragraphe est le premier élément p du body

Ce span est le premier enfant du 1er div

Premier élément p dans le 1er div

Deuxième élément p du div

Ce paragraphe est le dernier enfant du 1er div

Encore un paragraphe

Ce paragraphe est le premier enfant du 2è div

Ce paragraphe est le dernier enfant du 2è div

Ce paragraphe est le dernier élément du body



CSS : les sélecteurs

Au pseudo-classe d'une balise

Pseudo-classe **nth-child()**. Plusieurs paramètres:

- odd | even
- valeur numérique
- $an+b$

p : **nth-child**(**n**)

Toute balise <p> fille d'un élément

p : **nth-child**(**2n**)

Toutes les 2/n balises <p> filles d'un élément

p : **nth-child**(**3n**)

Toutes les 3/n balises <p> filles d'un élément

p : **nth-child**(**2n+1**)

Toutes les 2/n balises <p> filles d'un élément à partir du 1^{er}

p : **nth-child**(**n+4**)

Toutes les balises <p> filles d'un élément à partir du 4^{ème}



CSS : les sélecteurs

```
<body>
  <h3>Un titre de niveau 3 </h3>
  <ul>
    <li class="elm"> element 1 </li>
    <li class="elm"> element 2 </li>
    <li class="elm"> element 3 </li>
    <li class="elm"> element 4 </li>
    <li class="elm"> element 5 </li>
    <li class="elm"> element 6 </li>
    <li class="elm"> element 7 </li>
    <li class="elm"> element 8 </li>
  </ul>
</body>
```

`nth-child(an+b)`



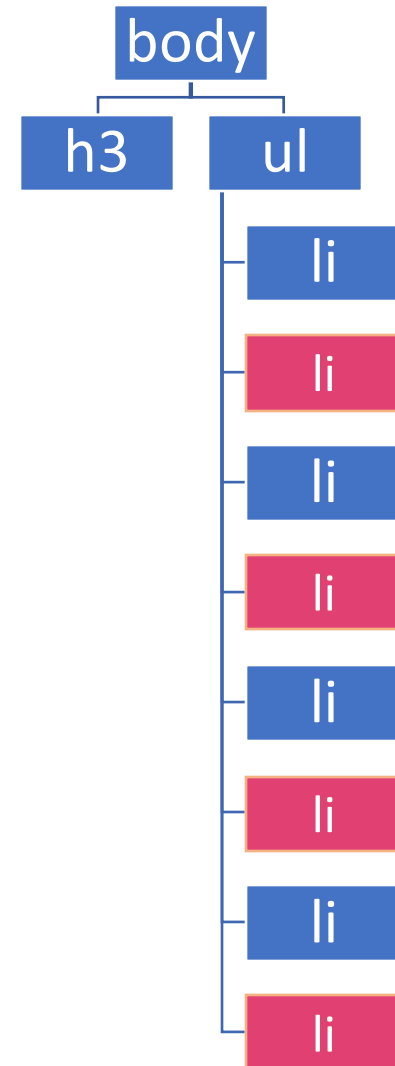
CSS : les sélecteurs

1 balise avec class = 'elm' / 2 fille d'un élément de la page

```
.elm:nth-child(2n){  
  color: hotpink;  
  text-decoration: underline wavy;  
}
```

- element 1
- element 2
- element 3
- element 4
- element 5
- element 6
- element 7
- element 8

`nth-child(an+b)`



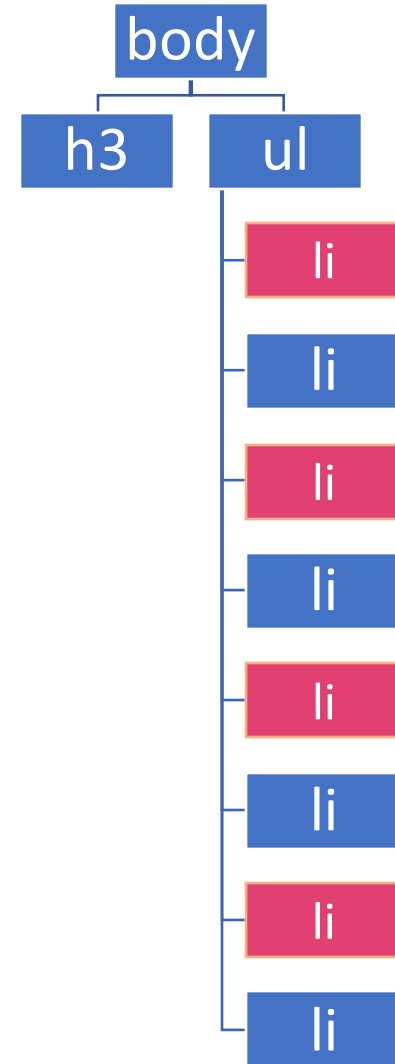
CSS : les sélecteurs

1 balise `<p>` / 2 fille d'un élément de la page à partir du 1^{er} élément

```
.elm:nth-child(2n+1){  
  color: hotpink;  
  text-decoration: underline wavy;  
}
```

- element 1
- element 2
- element 3
- element 4
- element 5
- element 6
- element 7
- element 8

`nth-child(an+b)`



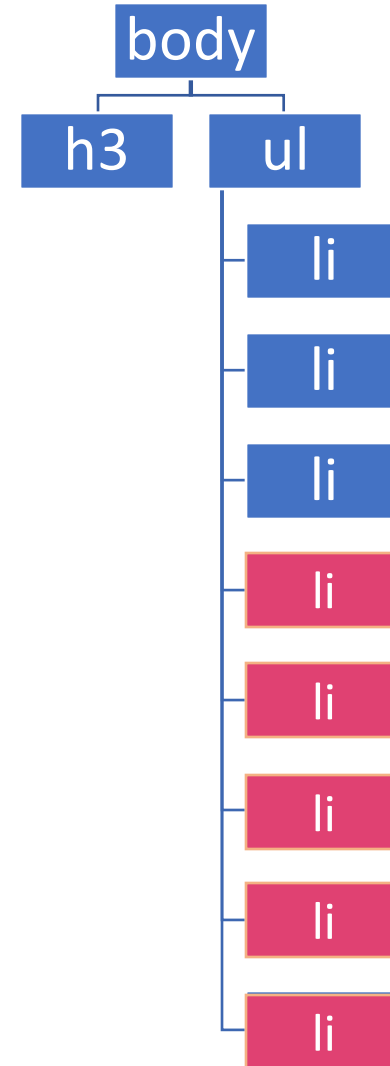
CSS : les sélecteurs

Toute balise <p> fille d'un élément de la page à partir du 4^{er} élément

```
.elm:nth-child(n+4){  
  color: hotpink;  
  text-decoration: underline wavy;  
}
```

- element 1
- element 2
- element 3
- element 4
- element 5
- element 6
- element 7
- element 8

`nth-child(an+b)`





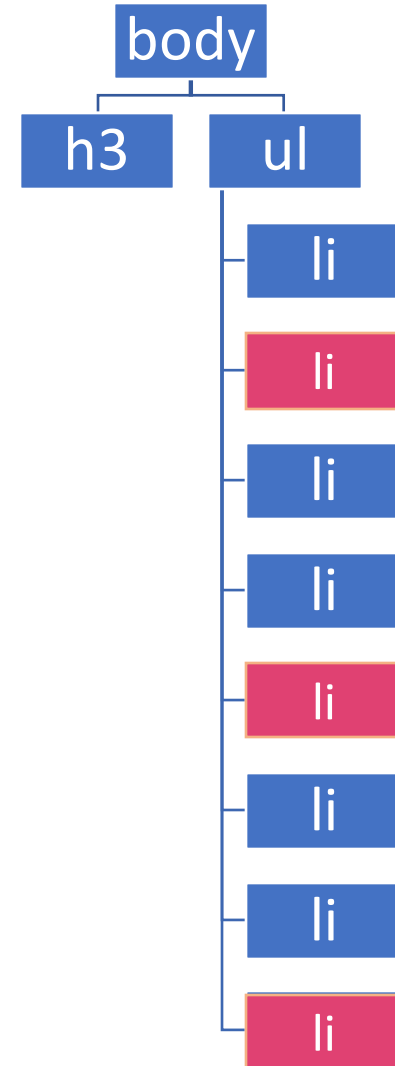
CSS : les sélecteurs

1 balise `<p>` / 3 fille d'un élément de la page à partir du 2^{ème} élément

```
.elm:nth-child(3n+2){  
  color: hotpink;  
  text-decoration: underline wavy;  
}
```

- element 1
- element 2
- element 3
- element 4
- element 5
- element 6
- element 7
- element 8

`nth-child(an+b)`





CSS : jeu de règles

Règle CSS

```
h3 {                               Propriétés et valeurs
  color: blue;
  font-family : " Arial Black " , Arial ;
}
```



CSS : les propriétés

Propriétés texte

- › **font-family** : police de caractères – [consultez lien w3c](#)
- › **font-size** : taille de la police, peut être exprimé en %, pt, em ou px | medium, large, small, etc.
- › **font-style** : normal, italic, etc.
- › **font-weight** : normal, bold, lighter, bolder, etc.
- › **color** : couleur du texte - couleur en anglais, code hexadécimal.
- › **text-transform** : la façon d'écrire les lettres majuscules – capitalize, uppercase, lowercase, etc.
- › **text-decoration** : décoration du texte – underline, underline dotted | weavy, overline, etc.



CSS : les propriétés

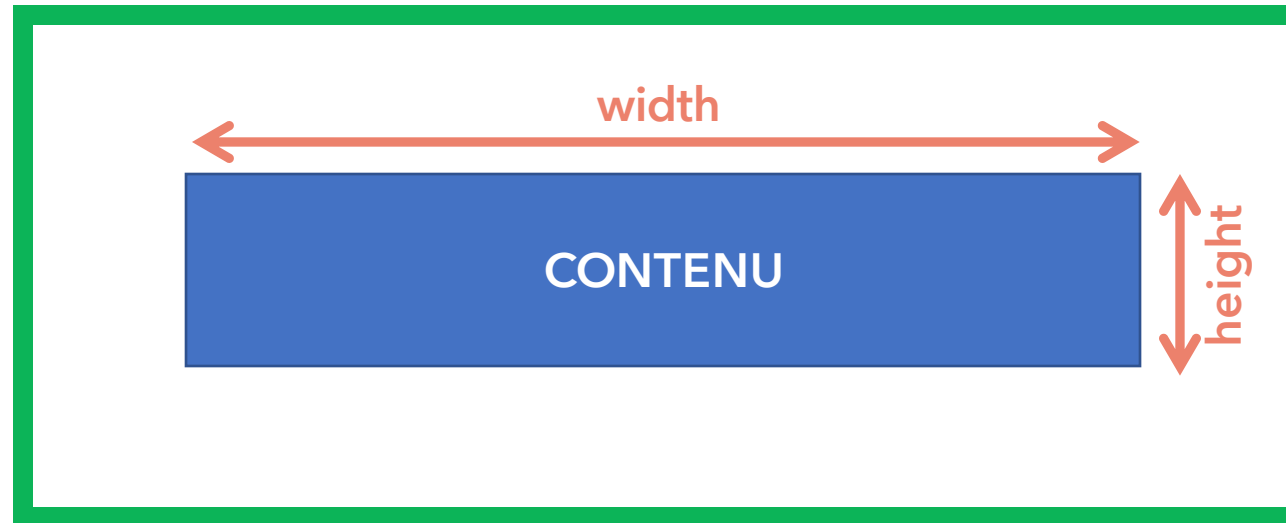
Propriétés d'un tableau

- › **border** : définir la taille le type et la couleur d'une bordure
- › **align** : alignement du tableau dans la page
- › **bgcolor** : couleur de fond
- › **cellpadding – padding** : espace entre le contenu et la bordure
- › **cellspacing** espace entre deux cellules
- › **valign | vertical-align** : alignement vertical – top | middle | bottom
- › **border-collapse**: fusion des bordures - collapse, separate
- › **table-layout**: géré la largeur : auto | fixed
- › **width** : largeur de la table
- › **empty-cells**: déterminer l'apparence des cellules vides – hide ou show
- › **caption-side** : emplacement de la légende top | bottom



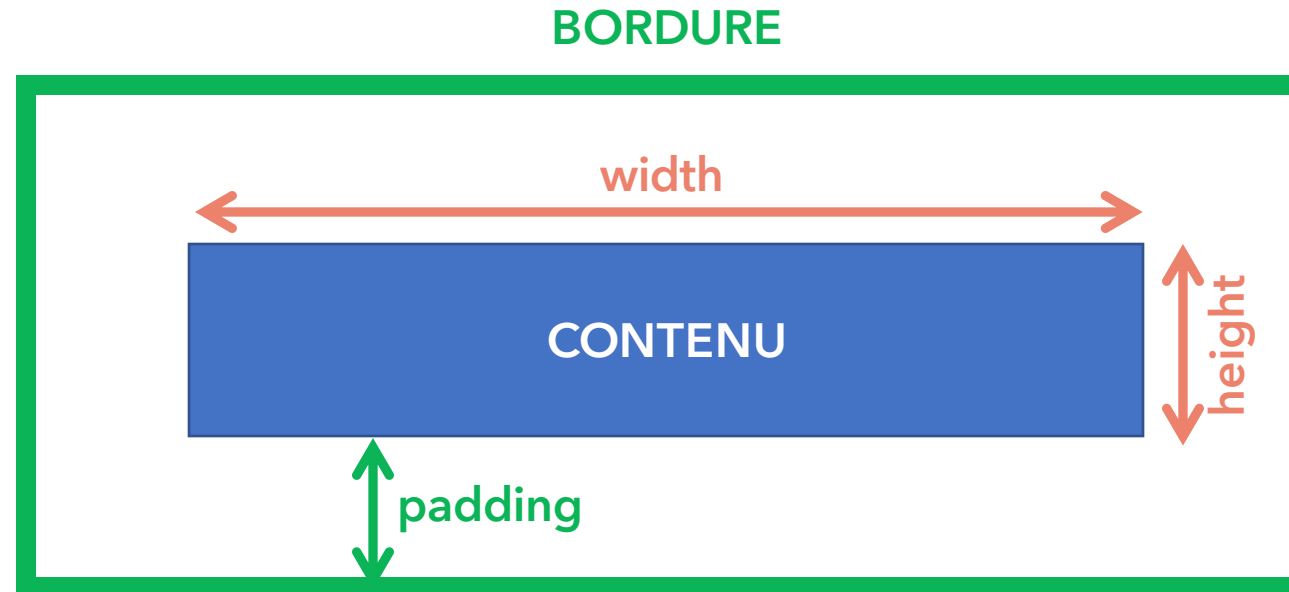
CSS : le modèle des layouts

BORDURE





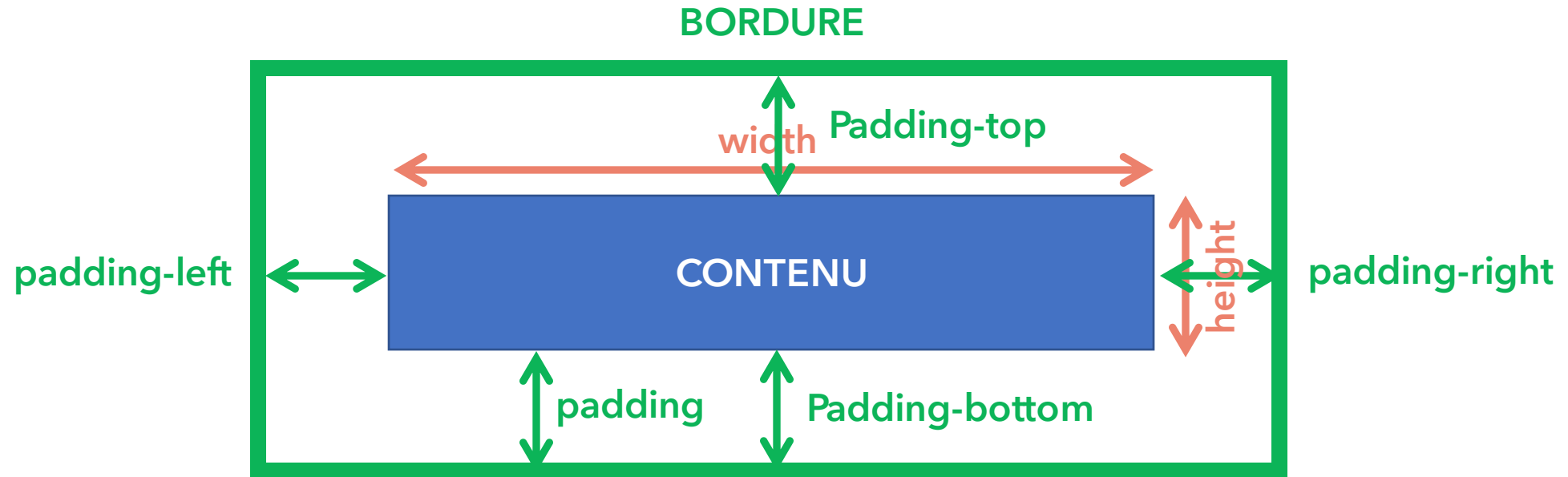
CSS : le modèle des layouts



`padding: 10px;`



CSS : le modèle des layouts



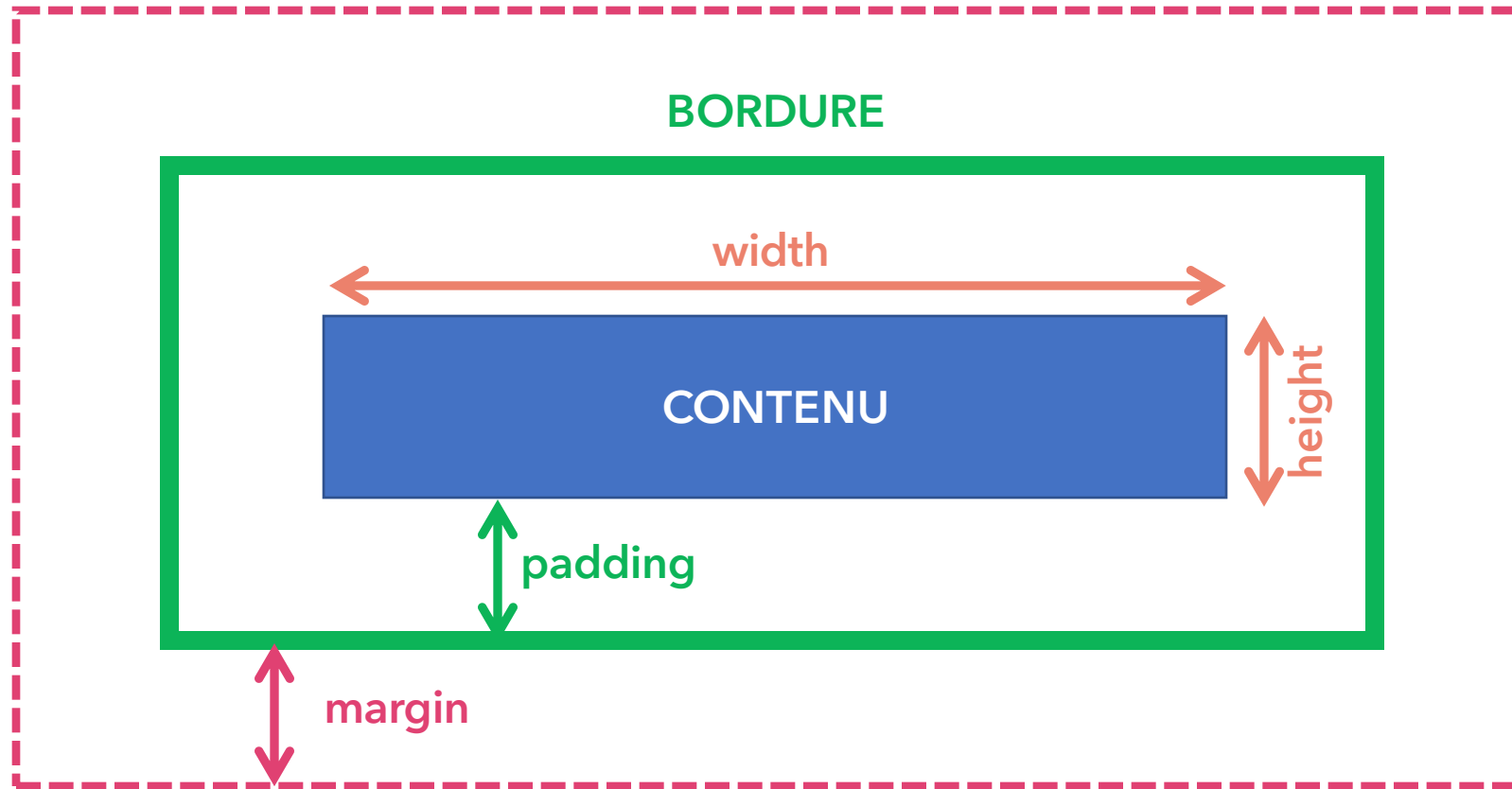
`padding: 10px;`



`padding-top: 10px;`
`padding-bottom: 10px;`
`padding-left: 10px;`
`padding-right: 10px;`



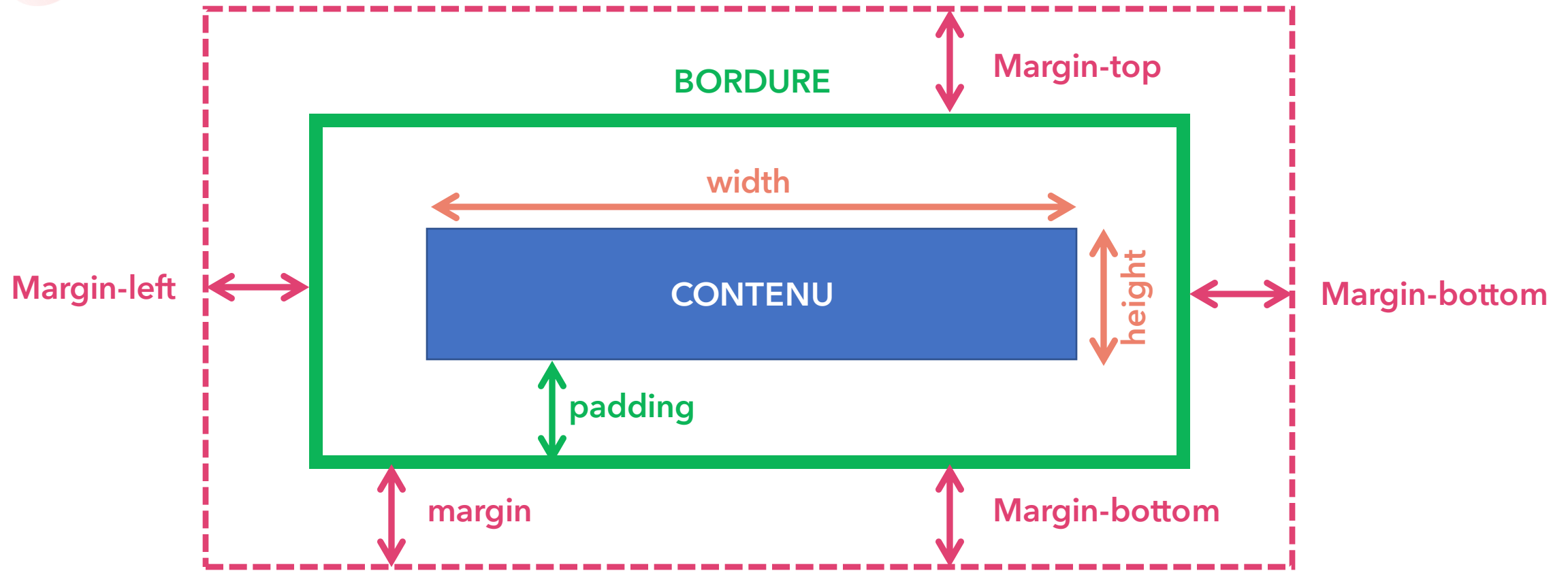
CSS : le modèle des layouts



`margin : 10px ;`



CSS : le modèle des layouts



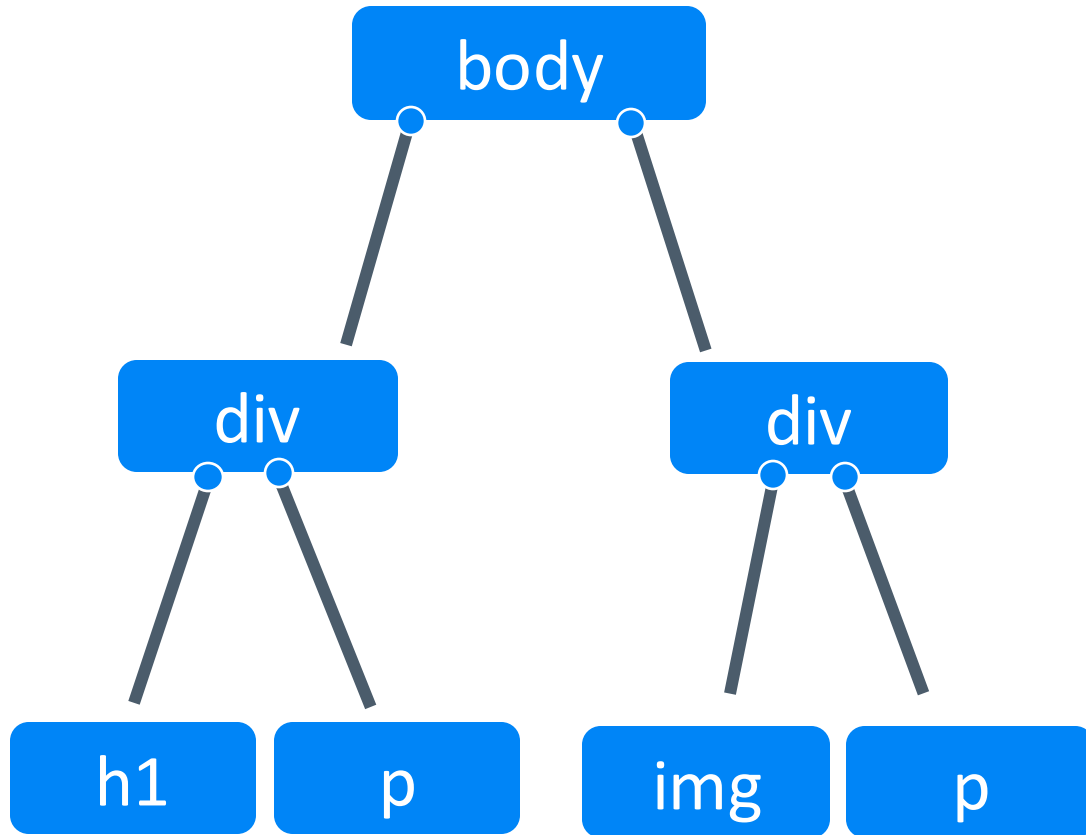
`margin : 10px ;`



`margin-top: 10px ;`
`margin-bottom: 10px ;`
`margin-left: 10px ;`
`margin-right: 10px ;`

Arbre DOM

(Document Object Model)



> Nœud = Boite

> Nœuds Père | Nœuds fils
→ Boite dans boite

> Nœuds frère
→ Comportement dépend de la nature de la balise

BALISE
BLOCK

BALISE
INLINE

BALISE
INLINE-
BLOCK



CSS : comportement des layouts

BLOCK

Les éléments concernés sont affichés les uns sous les autres.

Exemples:

- > titre h1 – h6
- > div
- > p
- > liste
- > ...

INLINE

Les éléments concernés sont affichés les uns à côté des autres.

Exemples:

- > span
- > a

On ne peut pas modifier la largeur et la hauteur

INLINE-BLOCK

Les éléments concernés sont affichés les uns à côté des autres **avec** possibilité de modifier la hauteur et la largeur.

Exemples:

- > img
- > button



CSS : propriété « display »

C'est la propriété display qui gère le type d'affichage des boîtes et qui va nous permettre de changer de type, si besoin.

Valeur : inline, block, inline-block et none

```
p { display: inline; }
```

Les balises <p> ne seront plus affichées l'une au dessous de l'autre mais une à côté de l'autre (par rapport aux autres balises de type inline).

```
span { display : block; }
```

Les balises ne seront plus affichées l'une à côté de l'autre mais une en dessous de l'autre



CSS : propriété « position »

Pour positionner une boîte là où vous le souhaitez, vous devez utiliser la propriété position.

Static

permet de placer la boîte dans le flux normal de la page

relative

permet de placer la boîte dans le flux normal avec une position relative indiquée par les propriétés de positionnement

absolute

permet de placer la boîte en position absolue par rapport au parent

fixed

permet de placer la boîte en position fixe



CSS : propriété « position »

Pour positionner une boîte là où vous le souhaitez, vous devez utiliser la propriété position.

Static

permet de placer la boîte dans le flux normal de la page

relative

permet de placer la boîte dans le flux normal avec une

absolute

permet de placer la boîte en position absolue par rapport

fixed

permet de placer la boîte en position fixe

Pour positionner la boîte là où vous souhaitez, utilisez les propriétés **top**, **right**, **bottom**, **left**. Chaque valeur spécifie les distances haute, droite, basse et gauche, entre la boîte et son bloc conteneur.

Propriété 'z-index' pour les éléments qui se chevauchent (démonstration)



CSS : propriété « position »

Pour positionner une boîte là où vous le souhaitez, vous devez utiliser la propriété position.

```
p { position : relative;  
      left : 20px }
```

Les balises <p> seront affichées en position relative par rapport à leur position par défaut
Position à 20px à gauche de la page

```
span { display : absolute;  
        right : 20px }
```

Les balises seront affichées en position absolue par rapport à la balise mère
Position à 20px à droite de la position de l'élément père



CSS : propriété « float »

Le flottement d'une boîte permet de sortir celle-ci du flux normal de la page et de la positionner relativement à son conteneur

```
p { float : right ; }
```

L'élément p va flotter à gauche par rapport à son conteneur

```
span { float : left ; }
```

L'élément span va flotter à droite par rapport à son conteneur



variable globale et calcul

```
:root{  
  --couleur: lightpink;}
```

Déclaration d'une variable global root avec la valeur « lightpink »

```
p{ color : var(--couleur);}
```

La couleur du texte des balise <p> sera en lightpink couleur défini par la variable globale '--color'



variable globale et calcul

```
:root{ --malargeur: 50 px;}
```

Déclaration d'une variable globalz 'malargeur' avec la valeur '50px'

```
p { width : var(--malargeur); }
```

La largeur de toute les balises sera = à 50px

```
div { width : calc(var(--malargeur) * 2); }
```

La largeur de toute les balises div sera = à 100 px



CSS : les grilles

Le module **CSS Grid layout** (modèle de disposition en grille) est un module de la spécification CSS qui permet de créer des mises en page en divisant l'espace d'affichage en régions



CSS : les grilles

```
<body>
```

```
  <div class="grid">  
    <div class="D1"> </div>  
    <div class="D2"></div>  
    <div class="D3"></div>  
    <div class="D4"></div>  
    <div class="D5"></div>  
  </div>|
```

```
</body>
```



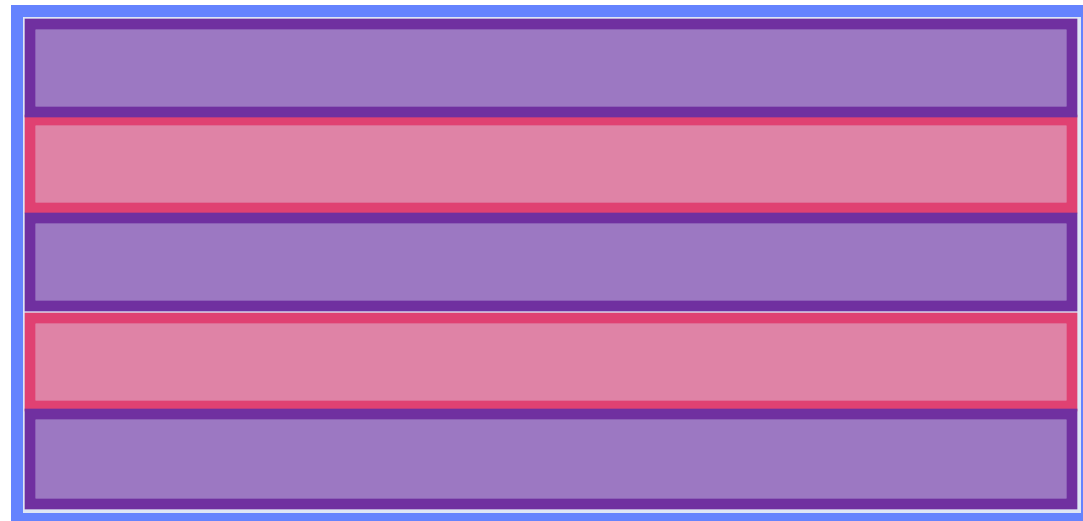


CSS : les grilles

```
.grid{
```

```
/* Créer la grille */  
display: grid;
```

```
}
```



CSS : les grilles

```
.grid{
```

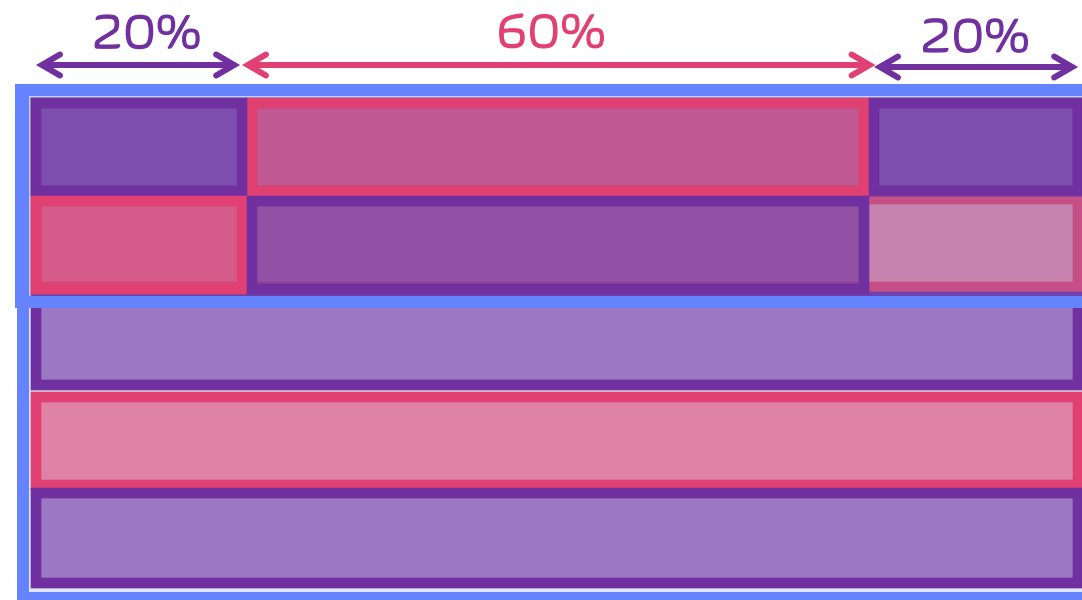
```
/* Créer la grille */
```

```
display: grid;
```

```
/* taille des Colonnes */
```

```
grid-template-columns: 20% 60% 20%;
```

```
}
```





CSS : les grilles

```
.grid{
```

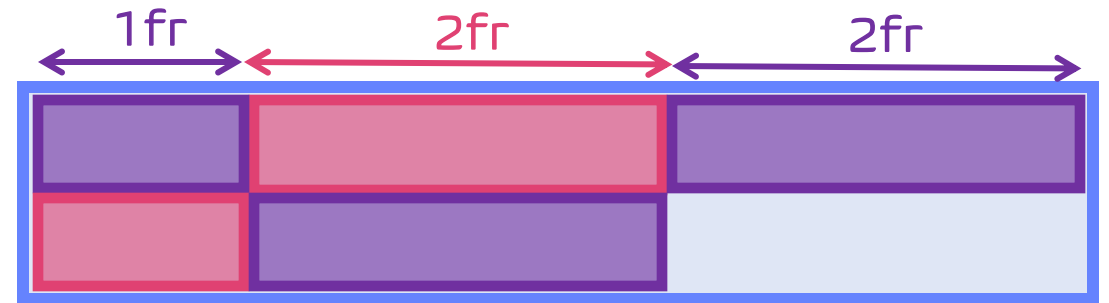
```
/* Créer la grille */
```

```
display: grid;
```

```
/* taille des colonnes */
```

```
grid-template-columns: 1fr 2fr 2fr;
```

```
}
```





CSS : les grilles

```
.grid{
```

```
/* Créer la grille */
```

```
display: grid;
```

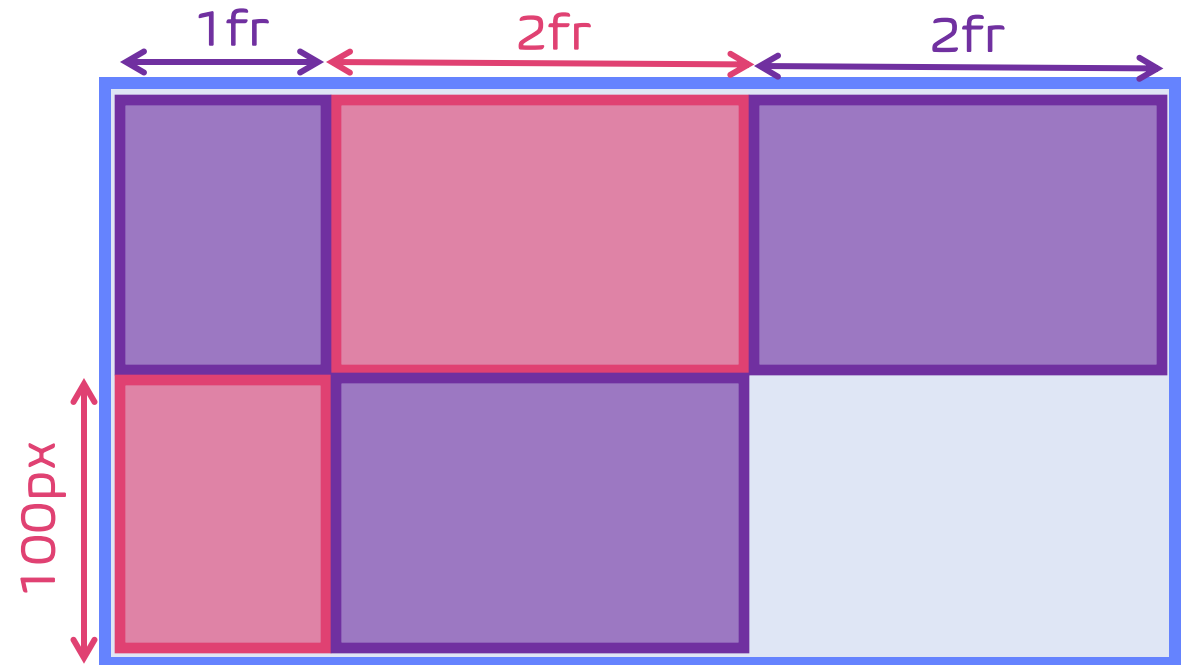
```
/* taille des colonnes */
```

```
grid-template-columns: 1fr 2fr 2fr;
```

```
/* taille des lignes */
```

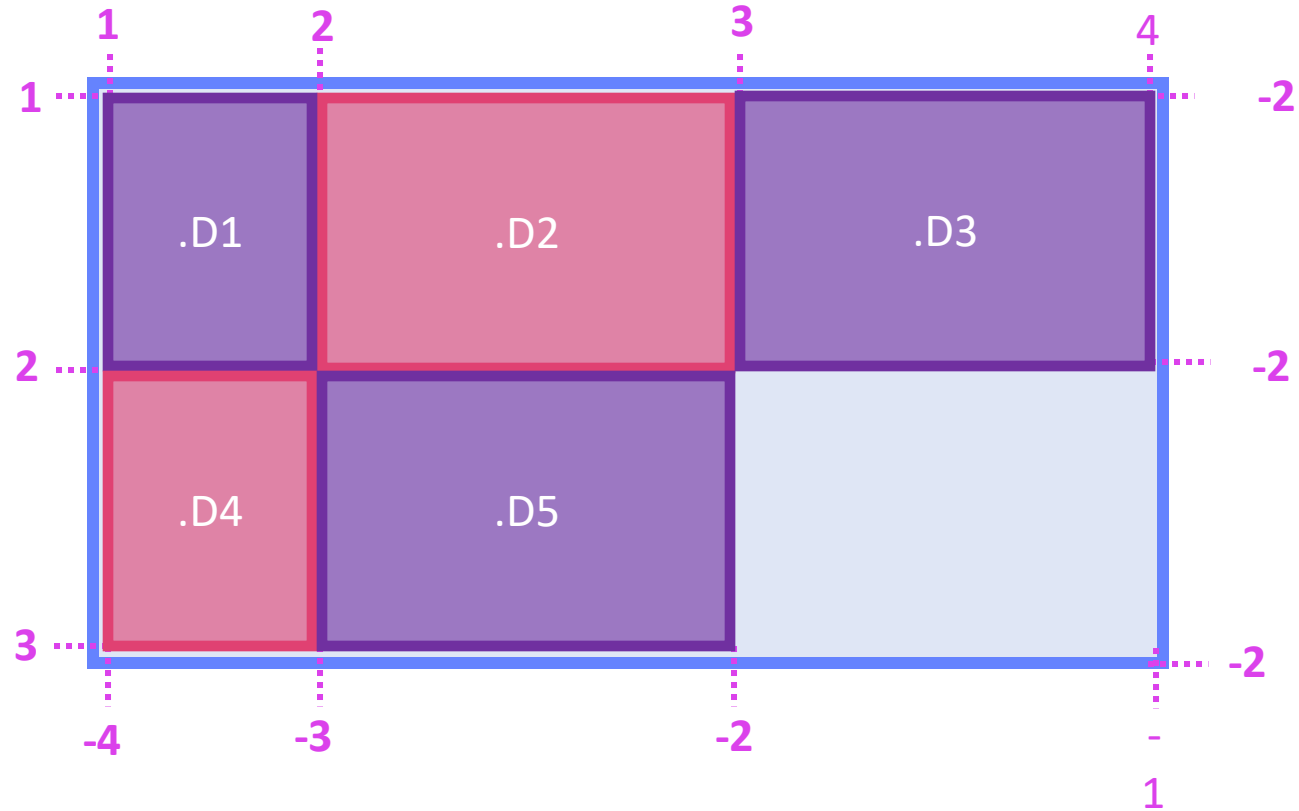
```
grid-template-rows: 100px;
```

```
}
```





CSS : les grilles



- > **grid-row-start** | **grid-row-end** | **grid-row** : définir la position de placement sur les lignes de la grille
- > **grid-column-start** | **grid-column-end** | **grid-column** : définir la position de placement sur les colonnes de la grille
- > **grid-column-gap** | **grid-row-gap** | **grid-gap** : définir l'espace entre les cases



CSS : flexbox

Le module **CSS Flexible box layout** (modèle de disposition de boîtes flexible) est un module CSS qui permet de créer des dispositions flexibles et adaptables. Il permet de gérer l'alignement, l'orientation et l'échelle des éléments d'une page web de manière efficace. Il est particulièrement utile pour les mises en page réactives (responsive)





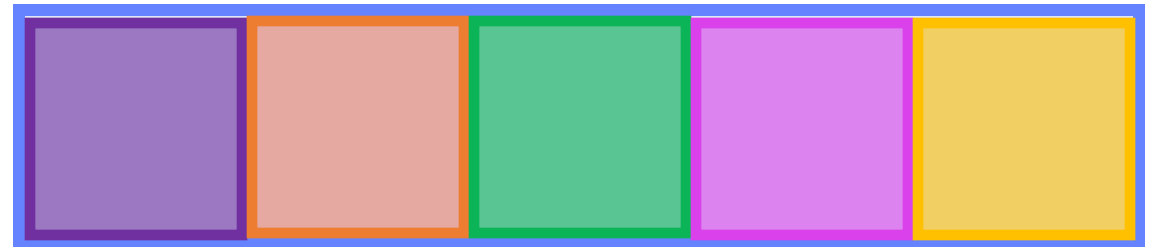
CSS : flexbox

```
.grid{
```

```
/* Créer la grille */  
display: flex;
```

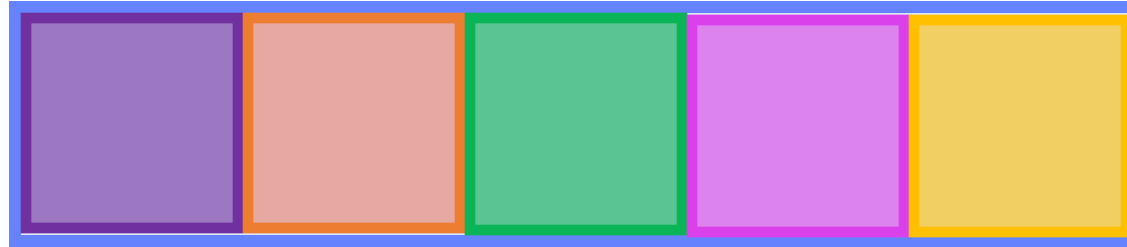
```
}
```

Création d'une grille sur une seule ligne
contenant tous les sous-éléments





CSS : flexbox



Pour le conteneur

- › **flex-direction**: définir l'axe principal des éléments flexibles | **column**, **row**, **column-reverse**, **row-reverse**.
- › **flex-wrap** : définir la possibilité aux éléments flexibles d'aller à la ligne | **wrap**, **nowrap**, **wrap-reverse**.
- › **flex-flow** : combinaison de flex-direction et flex-wrap.
- › **justify-content** : aligner les éléments d'un conteneur flexibles | **center**, **flex-start**, **space-between**, **etc.**
- › **align-items** : gérer l'alignement des selon l'axe secondaire (row ou column) | **stretch**, **flex-start**, **flex-end**, **etc.**

Pour les sous-éléments

- › **order** : définir l'ordre d'affichage élément par élément (par défaut = 0)
- › **justify-self** : définir l'alignement d'une boîte sur l'axe principal
- › **flex-grow**: définir la capacité des éléments à s'étirer dans leur conteneur pour remplir l'espace vide.



CSS : flexbox - grid

Flexbox

Démonstration : https://leria-info.univ-angers.fr/~a.jamin/l1_dw/demos-ctd/css-flexbox-layout.html

Documentation : https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Flexible_Box_Layout

Grid

Démonstration : https://leria-info.univ-angers.fr/~a.jamin/l1_dw/demos-ctd/css-grid-layout.html

Documentation : https://developer.mozilla.org/fr/docs/Web/CSS/CSS_Grid_Layout

Démonstration block (pour comparer)

https://leria-info.univ-angers.fr/~a.jamin/l1_dw/demos-ctd/css-block-layout.html