

1. Éléments de type *block* et *inline* : <https://player.vimeo.com/video/243807666?color=7451eb>

Exemple :

```
<p>
  Bonjour, <u>je </u>suis un <em>héro</em><br>
  La <strong>vie</strong> est <span style='color: ■pink;'>trop belle</span> .
  <img src='smiley.png'>
</p>
<div>
  <img src='smiley.png'><img src='smiley.png'><img src='smiley.png'><img src='smiley.png'>
</div>
```

Bonjour, je suis un *héro*
La vie est **trop belle** 😊

Règles :

- Les éléments *block* reviennent à la ligne, pas les éléments *inline*
- On peut centrer les éléments *inline* avec la propriété : `text-align : center`
- Les principaux éléments *inline* sont : `<input>` ; `<label>` ; `` ; `<iframe>` ; `` et aussi toutes ces petites balises qui s'insèrent dans un texte : `` ; `
` ; `` ; `<u>` ; `<sub>` ; `<sup>` ; ...



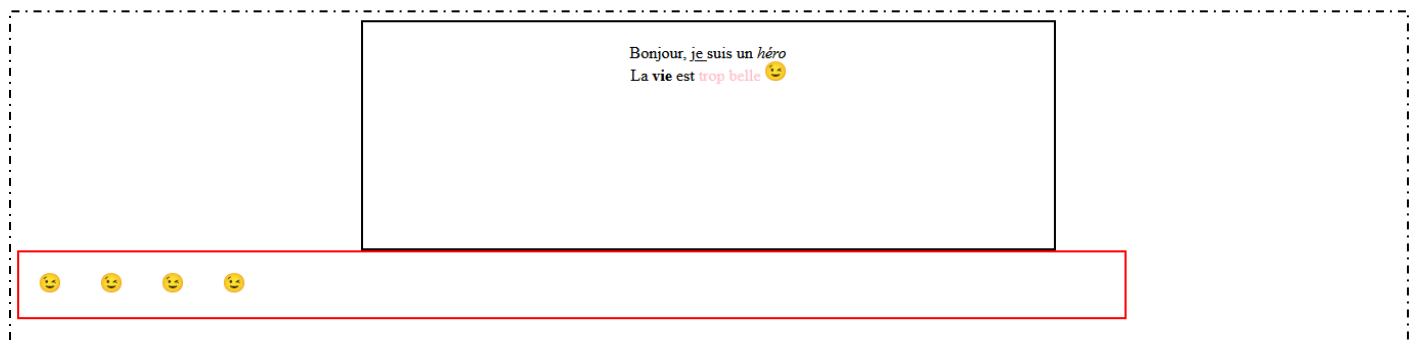
2. Gestion des éléments de type *block*

Exemple :

```
p{
  border:2px solid ■black;
  width:50%;
  height:200px;
  margin:auto;
  padding-top: 20px;
  text-align: center;
  word-wrap: break-word;
}
div{
  border:2px solid ■red;
  width:80%;
}
div img{
  margin:20px;
}
```

→ Permet de centrer le bloc `<p>` dans son conteneur parent

→ Permet de centrer les éléments *inline* contenus dans le bloc `<p>`



3. Gestion du positionnement des éléments de type *block*

3.1. Utilisation des propriétés **flex** : <https://player.vimeo.com/video/243807888?color=7451eb>

Exemple :

```
<body>
  <div class='je_suis_le_conteneur'>
    <div class='t cc'>Je suis un texte</div>
    <div class='t cc'>Je suis aussi un texte</div>
    <div class='t ii'><img src='smiley.png'></div>
    <div class='t'>Je suis une liste
      <ul>
        <li>choix 1</li>
        <li>choix 2</li>
      </ul>
    </div>
    <div class='t'><iframe width="560" height="315" src="https://www.youtube.com/embed/Vx4ReBkMpP0"
  </div>
</body>
```

```
.je_suis_le_conteneur {
  border:2px solid red;
  display:flex;
  flex-direction: row;
  justify-content: space-between;
  align-items:center;
  flex-wrap: wrap;
}
.t {border:1px solid black;min-width: 100px;}
.cc {color:green;word-wrap: break-word;}
.ii img{max-width: 100%;width:200px;}
```



Règles : Le CONTENEUR doit avoir la propriété flex : **display : flex** ;

- **flex-direction** : permet de gérer le positionnement en ligne (row) ou en colonne (column),
- **justify-content** : permet de gérer les écartements entre les blocks suivant la flex-direction choisie, (peut prendre les valeurs : flex-start , flex-end , center , space-between , space-around)
- **align-items** : permet de gérer les écartements entre les blocks suivant la perpendiculaire à la flex-direction choisie sur une ligne (si row) ou une colonne (si column) (peut prendre les valeurs : stretch , flex-start , flex-end , center)

- **align-content** : permet de gérer les écartements entre les différentes lignes (si row), s'il y en a plusieurs (peut prendre les valeurs : flex-start , flex-end , center , space-between , space-around)
- **flex-wrap** : si cette propriété prend la valeur wrap, si tous les blocs ne tiennent pas sur une ligne, ils se répartissent automatiquement sur plusieurs lignes (permet de s'adapter aux différentes tailles d'écran)

Si un block contenu dans le CONTENEUR a la propriété : **margin : auto** ; il va se centrer dans ce block.

D'une manière générale, **margin : auto** permet de centrer les éléments **blocks**.

3.2. Utilisation des propriétés display

```
<body>
  <div class='je_suis_le_conteneur'>
    <div class='t cc'>Je suis un texte</div>
    <div class='t cc'>Je suis aussi un texte</div>
    <div class='t ii'><img src='smiley.png'></div>
    <div class='t'>Je suis une liste
      <ul>
        <li>choix 1</li>
        <li>choix 2</li>
      </ul>
    </div>
    <div class='t'><iframe width="560" height="315" src="https://www.youtube.com/embed/Vx4ReBkMpP0"
  </div>
</body>
```

```
.je_suis_le_conteneur {
  border:2px solid red;
  text-align:center;
}
.t {border:1px solid black;display:inline-block;}
.cc {color:green;word-wrap: break-word;}
.ii img{max-width: 100%;width:200px;}
```



Règles :

- On applique la propriété **display : inline-block** aux éléments contenus dans le conteneur. Ainsi les blocks se comportent comme des éléments **inline** et ne vont pas revenir à la ligne.
- Pour gérer le positionnement des éléments entre eux, on peut utiliser :
 - text-align : center
 - margin et padding

3.3. Utilisation de la propriété `display : none`

Si un block se voit attribuer la propriété `display : none` il ne sera pas du tout affiché dans la page. En mettant par exemple un `display : none` dans un `:hover`, l'élément disparaît au passage de la souris.

3.4. Utilisation de la propriété `float`

Exemple :

```
<body>
  <div class='je_suis_le_conteneur'>
    <div class='t cc'>Je suis un texte</div>
    <div class='t cc'>Je suis aussi un texte</div>
    <div class='t ii'><img src='smiley.png'></div>
    <div class='t'>Je suis une liste
      <ul>
        <li>choix 1</li>
        <li>choix 2</li>
      </ul>
    </div>
    <div class='t'><iframe width="560" height="315" src="https://www.youtube.com/embed/Vx4ReBkMpP0"
  </div>
</body>
```

```
.je_suis_le_conteneur {
  border:2px solid red;
  text-align:center;
}
.t {border:1px solid black;}
.cc {color:green;word-wrap: break-word;}
.ii img{max-width: 100%;width:200px;float:right}
```



Règles :

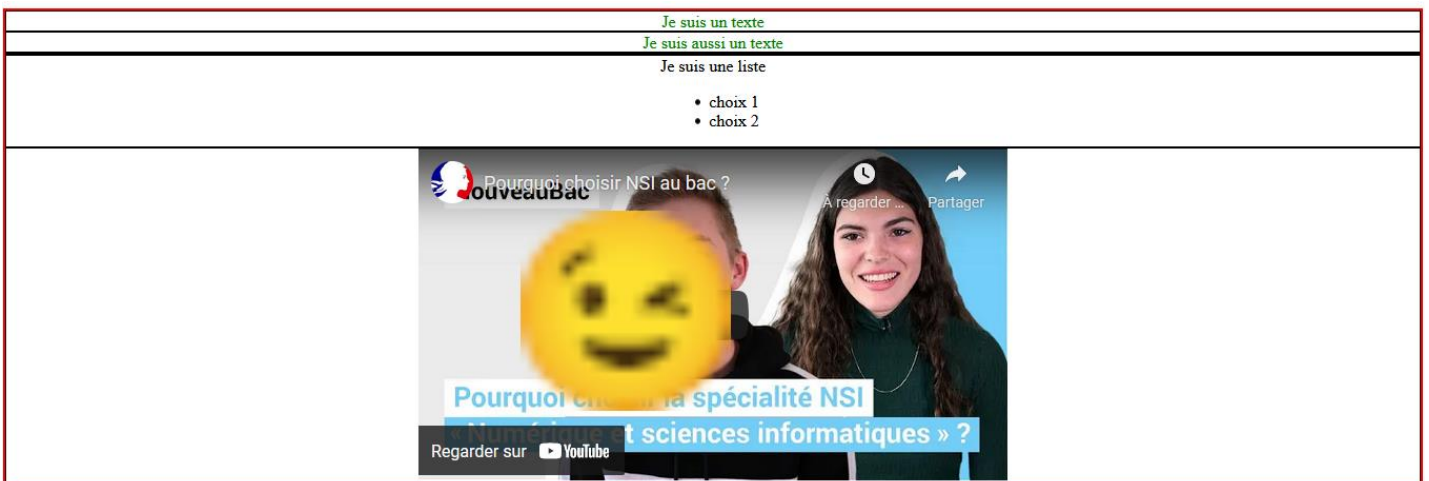
- la propriété `float` appliquée à un block, (valeurs : left, right) affiche l'élément en priorité dans son conteneur, il ne gêne pas le positionnement des autres éléments.
- Pour annuler cette propriété (en cas d'héritage), on utilise `clear` (options : left, right).

3.5. Utilisation de la propriété `position`

Exemple :

```
<body>
  <div class='je_suis_le_conteneur'>
    <div class='t cc'>Je suis un texte</div>
    <div class='t cc'>Je suis aussi un texte</div>
    <div class='t ii'><img src='smiley.png'></div>
    <div class='t'>Je suis une liste
      <ul>
        <li>choix 1</li>
        <li>choix 2</li>
      </ul>
    </div>
    <div class='t'><iframe width="560" height="315" src="https://www.youtube.com/embed/Vx4ReBkMpP0"
  </div>
</body>
```

```
.je_suis_le_conteneur {
  border:2px solid red;
  text-align:center;
}
.t {border:1px solid black;}
.cc {color:green;word-wrap: break-word;}
.ii img{max-width: 100%;width:200px;position:absolute; left:500px ; top:200px;}
```



Règles :

- La propriété *position* appliqué à un block, (valeurs : *absolute* , *fixed* , *relative*) affiche l'élément en priorité dans son conteneur, il ne gêne pas le positionnement des autres éléments.
- Pour positionner cet élément dans son CONTENEUR, on utilise les propriétés *left*, *top*,
- Si des éléments se superposent, on peut utiliser la propriété *z-index* pour gérer les conflits.
- La valeur de *position* normale, attribuée par défaut est *static*.