

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**MINISTERE DE L'ENSEIGNEMENT ET DE LA FORMATION PROFESSIONNELS**  
**INSTITUT NATIONAL SPECIALISE DE FORMATION PROFESSIONNELLE**  
**« El chahid Ahmed Mahdi »**

Direction des études :

Chapitre 2 : Effectuer les mises en forme avec CSS

Formateur : M<sup>me</sup> BENAMMOUR

Spécialité : D v web et mobile S1 .

lecon3

Mati re : les interfaces utilisateurs statiques

## LES UNITES DE MESURE EN CSS

### I.DEFINIION :

---

Les unit s servent   d crire diff rentes expressions comme des longueurs ou les couleurs (width, height, font-size, ...).

Il existe deux types d'unit s :

- ✓ **les unit s absolues** : leur grandeur est fixe
- ✓ **les unit s relatives** : leur grandeur est relative   un autre  l ment   savoir :
  - Les unit s relatives au texte
  - Les unit s en pourcentage
  - Les unit s relatives au viewport

### II.UNITES DE LONGUEUR ABSOULUES

---

Elles sont g n ralement utilis es pour traduire une longueur, par exemple un padding, une hauteur de bloc, la taille des caract res, etc.

#### 1. Unit s de longueur absolues (recommand es)

##### a) Les mots-cl s :

Les mots-cl s sont une m thode simple et efficace pour d terminer la taille de fonte   utiliser. En utilisant un mot-cl  sur l' l ment <body>, on peut d finir des tailles relatives sur toute la page et r duire ou augmenter la taille de la fonte en cons quence.

Les mots clefs suivants correspondent à des valeurs absolues de police mais qui sont relatives aux réglages faits par l'utilisateur dans son navigateur.

- xx-small : la taille de la police sera égale à la moitié de celle définie dans le navigateur ;
- x-small : la taille de la police sera égale à 60% de celle définie dans le navigateur
- small : la taille de la police sera égale à 80% de celle définie dans le navigateur
- medium : la taille de la police sera égale à celle définie dans le navigateur
- Large : la taille de la police sera 10% plus grande que celle définie dans le navigateur ;
- x-large : la taille de la police sera 50% plus grande que celle définie dans le navigateur ;
- xx-large : la taille de la police sera deux fois plus grande que celle définie dans le navigateur.
- smaller : la taille de la police de l'élément sera plus petite que celle de son élément parent ;
- larger : la taille de la police de l'élément sera plus grande que celle de son élément parent.

### Exemple :

#### Html

```
<h1>Un titre de niveau 1</h1>

<p id="p1">Un paragraphe p1</p>
<p id="p2">Un paragraphe p2</p>
<p id="p3">Un paragraphe p3</p>
<p id="p4">Un paragraphe p4</p>
<p id="p5">Un paragraphe p5</p>
<p id="p6">Un paragraphe p6</p>
<p id="p7">Un paragraphe p7</p>
```

#### Css

```
body{
  font-family: Verdana, sans-serif;
  font-size: xx-large;
}
/*La taille de la police de #p1 et de #p2 dépend de celle fixée dans l'élément
parent (ici body)*/
#p1{font-size: smaller;}
#p2{font-size: larger;}
/*La taille de #p3, #p4, ... #p9 dépend des réglages du navigateur et pas d'un
autre élément de la page*/
```

```
#p3{font-size: xx-small;}  
#p4{font-size: x-small;}  
#p5{font-size: small;}  
#p6{font-size: medium;}  
#p7{font-size: large;}  
#p8{font-size: x-large;}  
#p9{font-size: xx-large;}
```

# Un titre de niveau 1

Jn paragraphe p1

Jn paragraphe  
p2

**NOTE :** notre titre h1 hérite de la valeur de la propriété de son parent, c'est-à-dire font-size: xx-large et va donc s'afficher en très gros puisque le réglage par défaut des navigateurs pour un titre h1 définit généralement une taille de police deux fois plus grande que pour un paragraphe.

**b) Les pixels :**

L'utilisation des pixels (px) permet d'obtenir un résultat statique, absolu et indépendant du système d'exploitation et du navigateur.

**Exemple :**

```
body{
  font-family: Verdana, sans-serif;
  font-size: 16px;
}

/*La taille de police du h1 est définie de manière absolue et fixe*/
h1{
  font-size: 16px
```

**2. unités de longueur absolues - (a éviter)**

- cm (centimètre = 1cm = 10 mm) Risque d'affichage différent sur plusieurs écrans.
- mm (millimètre) Même remarque.
- pc (pica = 1pc = 12pt) Un autre terme de typographie, donc même remarque.
- pt (point = 1pt = 1/72 in) Le point est une mesure typographique utilisée par les imprimeurs (d'où les "points par pouce" de certaines résolutions) et les traitements de texte.
- Les pouces (in)

**NOTE :** Il ne peut y avoir d'espace à l'intérieur de cette valeur et pour les nombres décimaux, la virgule est impérativement remplacée par un point.

**En accessibilité, ces valeurs ne sont pas recommandées, car elles empêchent l'internaute de redimensionner la taille d'affichage des caractères.**

**Ces valeurs sont donc à éviter côté web ! (Mais restent éventuellement utilisables pour une CSS dédiée à l'impression).**

### III.UNITES DE LONGUEUR RELAIVES

---

#### 1. Les unités relatives au texte :

Ce sont les unités qui font varier la taille d'un élément en fonction de la taille de la police ou de l'élément parent. Ils sont de plus en plus utilisés pour harmoniser les éléments d'un design. On retrouve :

- **Ex** : **très rarement utilisée**, cette unité est relative à la hauteur de la police actuelle en minuscule.
- **Ch** : cette unité est elle aussi **peu utilisée**, elle est relative à la largeur du caractère "0".
- **Em** : elle est proportionnelle à la taille de la police de l'élément parent ou du document. Par défaut, 1 em = 16 px si aucune taille de police n'est définie.

#### Exemple :

```
<body>
  <h1>Un titre de niveau 1</h1>
  <p>Un paragraphe p1</p>
  <p>Un paragraphe p2</p>
  <p>Un paragraphe <span>avec un span qui <span>contient</span> lui-même un
span</span></p>
</body>
```

#### Css:

```
body {
  font-family: Verdana, sans-serif;
  font-size: 16px;
}
/*Ici, 2em = 32px (2 fois la taille de l'élément parent qui est l'élément body)*/
h1{
  font-size: 2em;
}
p{
  font-size: 0.8em;
}
span{
  font-size: 1.5em;
}
```

Ici, la taille de la police du premier **span** dans notre paragraphe est égale à 1,5 fois celle du paragraphe (qui est son parent) tandis que la taille de la police du **span** enfant dans le **span** parent est égale à 1,5 fois celle du **span** parent.

Résultat :



Pour pallier ce problème, nous pouvons plutôt utiliser l'unité **rem**. Les valeurs fournies en **rem** vont être relatives à la taille définie pour la propriété de l'élément racine **html** et non pas à la taille définie pour l'élément parent.

**Donc :**

— **Rem** : c'est l'unité qui fait toujours référence à la taille de la police de l'élément racine. En d'autres termes, elle dépend du font-size définit par défaut.

**Exemple (exemple précédent) :**

```
body{
  font-family: Verdana, sans-serif;
  font-size: 16px;
}
/*Ici, 2em = 32px (2 fois la taille de l'élément parent qui est l'élément body)*/
h1{
  font-size: 2em;
}
p{
  font-size: 0.8em;
}

/*La taille de la police des span dépend de la taille réglée dans l'élément html ou de celle par
défaut du navigateur si rien n'a été précisé (16px = 1em généralement)*/
span{
  font-size: 1.5rem;
}
```

## Résultat

# Un titre de niveau 1

Un paragraphe p1

Un paragraphe p2

Un paragraphe avec un span qui contient lui même un span

## 2. Les unités en pourcentage

Les valeurs exprimées en pourcentage vont être proportionnelles à la valeur renseignée pour la propriété dans l'élément parent.

La plupart du temps, cette unité est utilisée pour définir la hauteur et la largeur d'un élément. Vous la retrouverez notamment dans le framework Bootstrap.

### Même Exemple :

```
body{
  font-family: Verdana, sans-serif;
  font-size: 16px;
}
/*Ici, 2em = 32px (2 fois la taille de l'élément parent qui est l'élément body)*/
h1{
  font-size: 2em;
}
/*80% = 0.8e*/
p{
  font-size: 80%;
}

span{
  font-size: 150%;
}
```

## Résultat

# Un titre de niveau 1

Un paragraphe p1

Un paragraphe p2

Un paragraphe avec un span qui contient lui même un span

### 3. Les unités relatives au viewport

D'autres unités permettent aux éléments de s'adapter à la taille du viewport, c'est-à-dire à la taille de la fenêtre du navigateur.

Ces unités **relatives au viewport** sont essentielles pour mettre en place **un design responsive**, et on retrouve :

- La hauteur du viewport (**vh**)
- La largeur du viewport (**vw**)
- Le viewport minimum (**vmin**)
- Le viewport maximum (**vmax**)

#### a) Les unités vh et vw

Les unités vh et vw sont similaires, à la seule différence qu'elles dépendent respectivement de la hauteur et de la largeur de la fenêtre de navigation.

Retenez que  $1 \text{ vh} = 1 \% (1/100)$  de la hauteur du viewport et que  $1 \text{ vw} = 1 \%$  de la largeur du viewport.

L'unité de mesure vh est généralement utilisée pour permettre à un élément de s'adapter à la hauteur totale de la fenêtre :

**Exemple :**



```
.box {  
height :100vh ;  
background : red ;  
}
```

L'élément **box** a une hauteur de 100 % par rapport à la hauteur de la fenêtre de navigation.

### Exemple :

```
h1{  
font-size : 6 vw ;  
}
```

Ici la taille du titre H1 correspond à 6% de la largeur de la fenêtre. Par conséquent, si cette dernière est de 1000 px, la taille de la police sera de 60 px.

## b) Les unités vmin et vmax

Ces deux unités de mesure en CSS fonctionnent selon le **même principe**.

Avec le **vmin**, c'est la dimension minimale du viewport qui est prise en compte. **Par exemple**, si ce dernier est de 1000 px de haut par 800 px de large, les éléments ayant une unité vmin s'adapteront en fonction de la largeur de la fenêtre.

Avec **le vmax**, les éléments s'adaptent en fonction de la dimension maximale du viewport. Selon l'exemple précédent, 1 vmax = 10 px.