

# Introduction aux technologies WEB



**Pr. Hajar IGUER**  
**[hajar.iguer@uic.ac.ma](mailto:hajar.iguer@uic.ac.ma)**

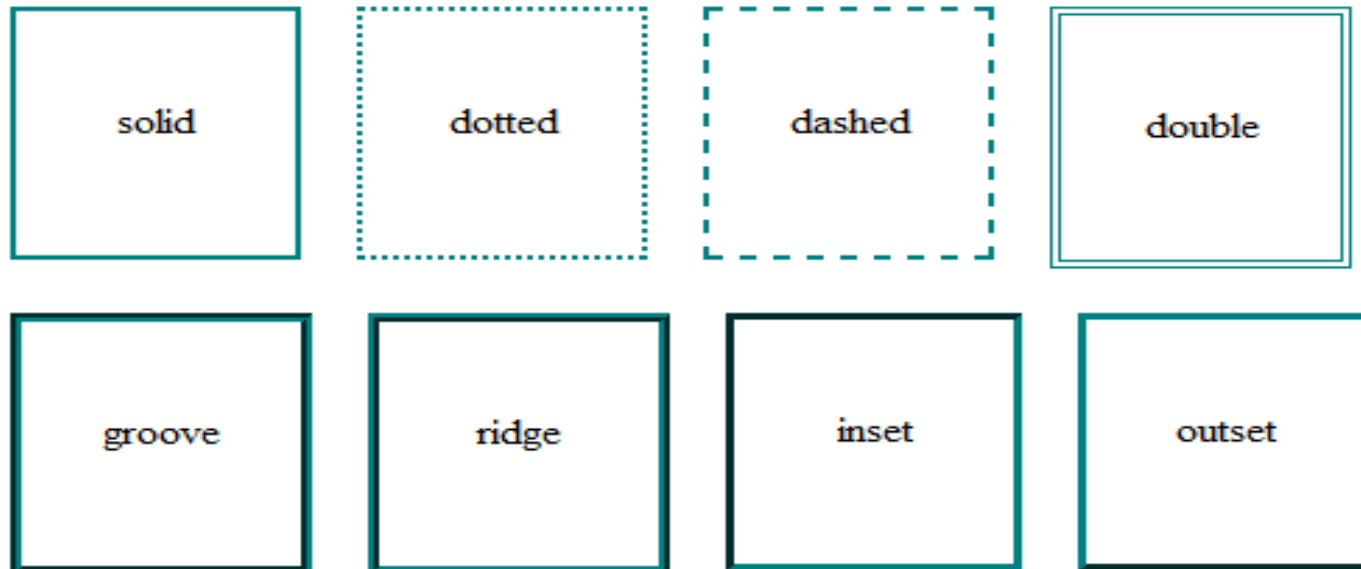


# CSS : CASCADING STYLE SHEET II



# Les Bordures

- border: 3px blue dashed;
- border-left: 2px solid black;
- border-right: 2px solid black;



# Les Bordures

- `border-radius: 10px;` : Arrondir la bordure d'un élément
- `border-radius: 10px 5px 10px 5px;` : Définir la forme arrondi de plusieurs coins du haut vers le bas et de gauche à droite
- `box-shadow: 6px 6px 0px black;` : Mettre une ombre pour un contour
  - le décalage horizontal de l'ombre ;
  - le décalage vertical de l'ombre ;
  - l'adoucissement du dégradé ;
  - la couleur de l'ombre.
- `text-shadow: 2px 2px 4px black;`

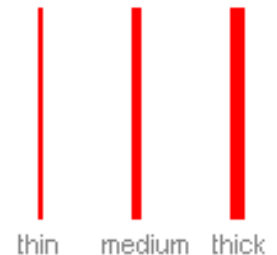
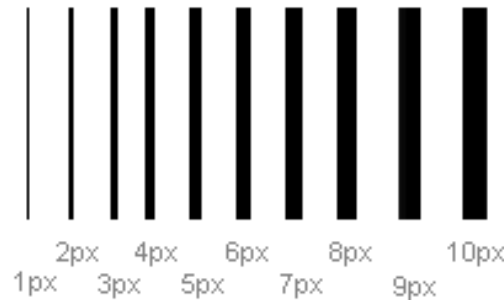


# Les Bordures

- La propriété `border-color`, ou `border-width` ne s'applique qu'en complément d'un style (`border-style`) ou d'une épaisseur de bordure. Sans cela, la bordure est inexistante.

- Exemples:

```
border-width: 3px;  
border-style: dotted;  
border-color: gray;
```



- Ou

```
border: 3px dotted gray;  
border-style: none;
```



# Les Bordures

- Les ombres font partie des nouveautés récentes proposées par CSS3.
- **box-shadow** : les ombres des boîtes
- La propriété box-shadow s'applique à tout le bloc et prend quatre valeurs dans l'ordre suivant :
  - le décalage horizontal de l'ombre ;
  - le décalage vertical de l'ombre ;
  - l'adoucissement du dégradé ;
  - la couleur de l'ombre.



# Les Bordures

- Exemple:

```
p
```

```
{
```

```
  box-shadow: 6px 6px 6px black inset;
```

```
  text-shadow: 2px 2px 4px black;
```

```
}
```

- Une 5<sup>ème</sup> valeur inset peut être rajouté. Dans ce cas, l'ombre sera placée à l'intérieur du bloc pour donner un effet enfoncé.



# Bordures

- **border-width**: Largeur de la bordure
- **border-color** : Couleur de la bordure
- **border-top** : bordure du haut ;
- **border-radius**: Bordures Arrondies

```
p  
{
```

```
border: 3px blue double;  
border-left: 2px solid black;  
border-right: 2px solid black;  
border-radius: 10px;
```

```
}
```





# Ombres

- `box-shadow`: ombre des boîtes
- `box-webkit-box-shadow` : Pour navigateurs mobile
- `text-shadow: 2px 2px 4px black;`
- La propriété `box-shadow` s'applique à tout le bloc et prend quatre valeurs dans l'ordre suivant :
  - le décalage horizontal de l'ombre ;
  - le décalage vertical de l'ombre ;
  - l'adoucissement du dégradé ;
  - la couleur de l'ombre.

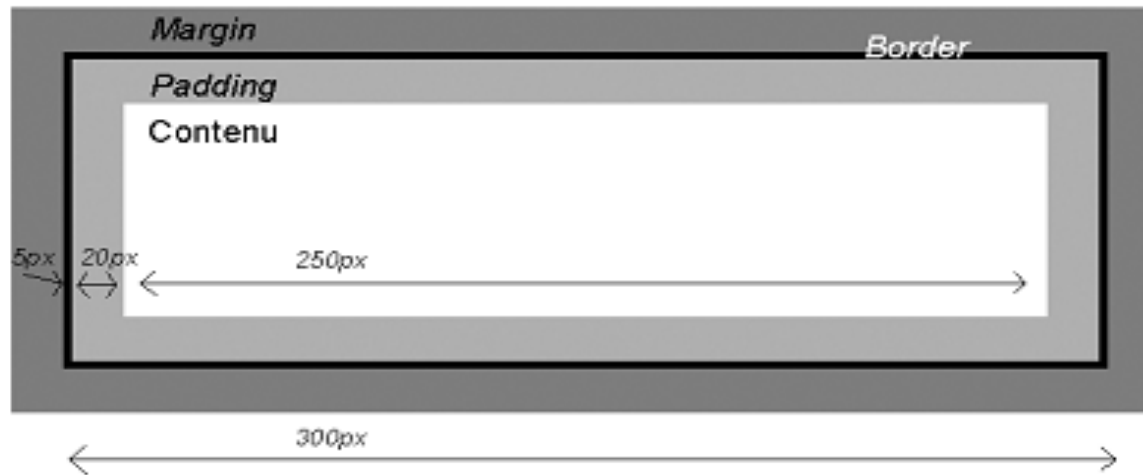
```
p  
{  
  box-shadow: 6px 6px 0px black;  
}
```



# Positionnement en CSS

Nous innovons pour votre réussite !

- Afin de positionner les balises dans notre page web, nous utilisons le modèle en boîte qui permet de préciser plusieurs propriétés CSS.



- La largeur et la hauteur d'une boîte sont calculées en ajoutant celles du contenu aux dimensions des marges internes et à l'épaisseur de la bordure



# Positionnement en CSS

Nous innovons pour votre réussite !

- En CSS, on peut modifier la taille des marges avec les deux propriétés suivantes :
- **padding** : indique la taille de la marge intérieure. À exprimer en général en pixels (px).
- **margin** : indique la taille de la marge extérieure. Là encore, on utilise le plus souvent des pixels.
- Exemple:

```
p
{
  width: 50%;
  min-width: 400px;
  Max-height: 700px;
  border: 1px solid black;
  text-align: justify;
  padding: 12px;
  Margin: 50 px;
}
```



# Positionnement en CSS

Nous innovons pour votre réussite !

- Voici la liste de propriétés pour margin :
  - **margin-top** : marge extérieure en haut ;
  - **margin-bottom** : marge extérieure en bas ;
  - **margin-left** : marge extérieure à gauche ;
  - **margin-right** : marge extérieure à droite.
- Et la liste pour padding :
  - **padding-top** : marge intérieure en haut ;
  - **padding-bottom** : marge intérieure en bas ;
  - **padding-left** : marge intérieure à gauche ;
  - **padding-right** : marge intérieure à droite.



# Positionnement relatif

- Un élément en position relative garde sa place dans le document, sa position est **calculée d'après sa position originale dans le document.**
- Exemple:

```
#test1 { position:relative; left: 350px; bottom: 150px; }
```



# Positionnement absolu

- Un élément en position absolue ne reçoit aucune place dans le document. Il ne laisse aucun espace vide après qu'il est positionné.
- Exemple:

```
#box1 { position:absolute; top: 50px; left: 50px; right: 50px; bottom: 50px;}
```



# Police personnalisée : Exemple

```
@font-face { /* Définition d'une nouvelle police nommée
LearningCurveProRegular */
  font-family: 'LearningCurveProRegular';
  src: url('LearningCurve_OT-webfont.eot');
  src: url('LearningCurve_OT-webfont.eot?#iefix') format('embedded-
opentype'),
      url('LearningCurve_OT-webfont.woff') format('woff'),
      url('LearningCurve_OT-webfont.ttf') format('truetype'),
      url('LearningCurve_OT-webfont.svg#LearningCurveProRegular')
format('svg');
}

h1 /* Utilisation de la police qu'on vient de définir sur les titres */
{
  font-family: 'LearningCurveProRegular', Arial, serif;
}
```



# Apparence dynamique

- C'est une de ses forces : le CSS nous permet aussi de modifier l'apparence des éléments de façon dynamique.
- Les pseudo formats servent à rendre ou créer une apparence dynamique d'un élément en CSS.





# Les pseudo-formats

- Des éléments peuvent changer de forme une fois que la page a été chargée.
  - au survol ;
  - lors du clic ;
  - lors du focus (élément sélectionné) ;
  - lorsqu'un lien a été consulté.
- Un pseudo-format est une précision que l'on rajoute après le nom d'une balise



# Apparence dynamique

- Hover : Lorsque la souris survole l'élément
- Exemple:

```
a /* Liens par défaut (non survolés) */  
{  
  color: red;  
  font-style: italic;  
}
```

```
a:hover /* Apparence au survol des liens */  
{  
  text-decoration: underline;  
  color: green;  
}
```



# Apparence dynamique

- **Active** : Lorsque du clic de la souris
  - **Focus**: Lors de la sélection de l'élément
- ```
a:active /* Quand le visiteur clique sur le lien */  
{  
    background-color: #FFCC66;}  
a:visited /* Quand le lien a été consulté*/  
{  
    color: blue;}  
a:focus /* Quand le visiteur sélectionne le lien */  
{  
    background-color: #FFCC66;  
}
```



# Les pseudo-formats

- Au survol:
  - `a: hover { text-decoration: underline;`
  - `Font-style: italic; }`
- On peut avoir deux codes CSS relatifs aux liens, le premier pour les liens par défaut et le deuxième pour les liens survolés
- Au clic:
  - `a: active{`
  - `background-color: #FFAA77;}`
- À la sélection ou au focus
  - `a: focus{`
  - `text-decoration: overligne;}`
- Lorsqu'un lien a été consulté
  - `a: visited {color: #AAA;}`



# Standard du W3C

- Le concept des standards est de trouver un dénominateur commun pour l'utilisation des technologies Web.
- Pour votre code fonctionne dans la plupart des plateformes , vous devez en respecter les standards. **Ainsi, je vous recommande de vous appuyer sur les travaux du W3C.**



# Validation CSS

- Afin de faciliter le respect du standard CSS, le W3C a développé un validateur, lequel lit les feuilles de style et renvoie un état listant les erreurs et les avertissements au cas où votre code CSS ne serait pas valide.

<https://validator.w3.org/>

