

# Les préprocesseurs CSS

Merci de vous installer  
par binôme de niveau  
différent



# Installation de LESS

- Télécharger less:
  - Le fichier less.js:  
<http://lesscss.org/#download-options>
- Insérer dans le head:  

```
<link rel="stylesheet/less" type="text/css"  
      href="styles.less">  
<script src="less.js" type="text/javascript"></  
script>
```

# Les concepts derrière LESS

## ➤ Les variables

➤ S'écrit et s'appelle sous la forme:  
@mavar: valeur;  
#id { propriete: @mavar; }

➤ Exemple:  
@blue: #17B6FF;  
#header { background: @blue; }

# Les concepts derrière LESS

## ➤ Les fonctions

### ➤ Définition de la fonction:

```
.nom-fonction{  
}
```

### ➤ Appel de la fonction (ici à l'intérieur d'un id):

```
#id{  
.nom-fonction  
}
```

# Les concepts derrière LESS

## ➤ Les fonctions

### ➤ Exemple:

```
.border-radius {  
-moz-border-radius: 10px;  
-webkit-border-radius: 10px;  
border-radius: 10px;  
}
```

```
#foo {  
.border-radius;  
}
```

# Les concepts derrière LESS

## ➤ Les fonctions **avec arguments**

### ➤ Définition de la fonction:

```
.nom-fonction(argument){  
propriete:argument;  
}
```

### ➤ Appel de la fonction (à l'intérieur d'un id):

```
#id{  
.nom-fonction(valeurArgument)  
}
```

# Les concepts derrière LESS

## ➤ Les fonctions **avec arguments**

### ➤ Exemple:

```
.border-radius(@radius) {  
-moz-border-radius: @radius;  
-webkit-border-radius: @radius;  
border-radius: @radius;}
```

### ➤ Appel de la fonction (ici à l'intérieur d'un id):

```
#foo {  
.border-radius(30px);  
}
```

# Les concepts derrière LESS

- Les fonctions avec arguments **par défaut**

- Définition de la fonction:

```
.nom-fonction(argument:valParDefaut){  
propriete:argument;  
}
```

- Appel de la fonction (ici à l'intérieur d'un id):

```
#id{  
.nom-fonction();  
}
```

# Les concepts derrière LESS

- Les fonctions avec arguments **par défaut**

- Exemple:

```
.border-radius(@radius:10px) {  
-moz-border-radius: @radius;  
-webkit-border-radius: @radius;  
border-radius: @radius;}
```

- Appel de la fonction (à l'intérieur d'un id):

```
#foo {  
.border-radius();  
}
```

# Les concepts derrière LESS

- L'imbrication
  - Permet d'imbriquer des sélecteurs comme des balises HTML
  - Exemple:

```
#foo {  
background: lemon;  
bar { color: lightblue; }  
baz { color: orange; }  
}
```

# Les concepts derrière LESS

- L'imbrication
- Va générer:

```
#foo {  
#foo { background: lemon; }  
#foo bar { color: lightblue; }  
#foo baz { color: orange; }
```

# Les concepts derrière LESS

## ➤ L'imbrication

### ➤ Utilisation du &. Exemple:

```
#foo {  
background: lightgrey;  
&:hover { color: orange; }  
}
```

### ➤ Va générer:

```
#foo { background: lightgrey; }  
#foo:hover { color: orange; }
```

# Les concepts derrière LESS

## ➤ Les calculs

### ➤ Exemple:

```
@size_image: 100px;  
.box {  
width: @size_image + 10px;  
height: @size_image + 10px;
```

```
img {  
width: @size_image;  
height: @size_image;  
}  
}
```

# Passage rapide sur SASS

- Les variables commencent par un \$
- Les fonctions sont appelées mixins
- Il faut compiler le .scss en .css via un outils ou la console