

Chapitre 9 : Les grilles de mise en page

A. Les objectifs279
B. La grille Gridlex279
C. La grille Grillade.....	.289
D. La grille Simple Grid.....	.296

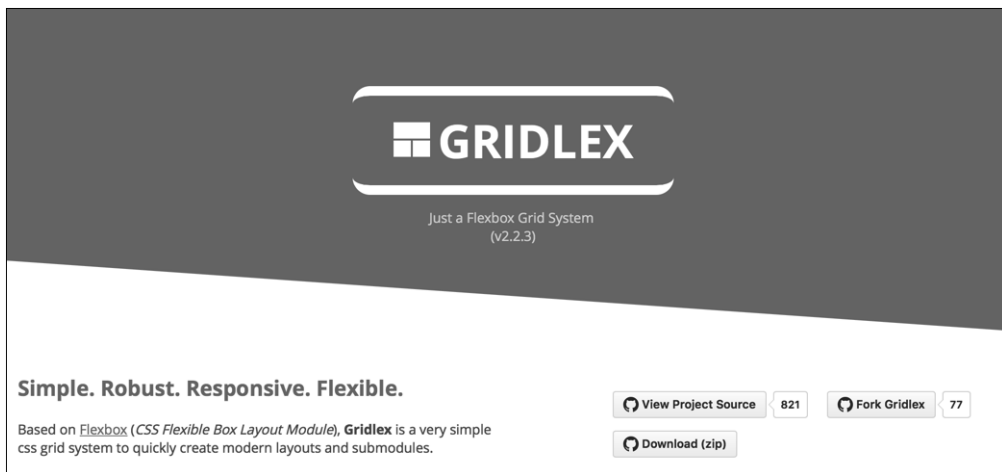
A. Les objectifs

Dans les chapitres précédents, nous avons vu les différentes techniques modernes pour créer des grilles de mise en page élaborées et responsive. Plutôt que de partir de zéro, *ex nihilo* ou *from scratch* comme disent les Anglo-Saxons, nous pouvons utiliser des grilles déjà conçues et prêtes à l'emploi. Cela nous permet de gagner du temps et ainsi d'avancer plus rapidement dans la réalisation de nos projets.

B. La grille Gridlex

1. Implémenter la grille

La grille **Gridlex** est "**Simple. Robust. Responsive. Flexible.**" selon son slogan. Voici son URL : <http://gridlex.devlint.fr>. Cette grille est basée sur le module **Flexible Box Layout** du W3C.



Pour implémenter **Gridlex** dans vos projets, vous pouvez lier chacune de vos pages à sa feuille de style CSS minimisée, sur un serveur CDN :

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/gridlex/2.2.3/gridlex.min.css">
```

Ou

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/gridlex/2.2.3/gridlex.min.css">
```

L'autre solution consiste à télécharger l'archive du projet. Sur la page d'accueil du site, cliquez sur le bouton **Download (zip)**. Vous téléchargez une archive nommée **gridlex-master.zip**.

Dans l'archive décompressée, vous trouverez notamment :

- un dossier nommé **docs** contenant des exemples et les feuilles de style CSS **gridlex.css** et **gridlex.min.css**, cette dernière correspondant au fichier optimisé et minimisé pour la production.
- un dossier nommé **src** dans lequel vous trouverez les styles CSS aux formats **sass** et **less**.

2. Les points de rupture

La grille **Gridlex** définit plusieurs points de rupture pour avoir une grille responsive.

- `@media screen and (max-width: 80em)` : pour les écrans ayant une largeur maximale de 80 em. Le suffixe à utiliser dans les règles CSS est `_lg`, pour *large*.
- `@media screen and (max-width: 64em)` : pour les écrans ayant une largeur maximale de 64 em. Le suffixe à utiliser dans les règles CSS est `_md`, pour *medium*.
- `@media screen and (max-width: 48em)` : pour les écrans ayant une largeur maximale de 48 em. Le suffixe à utiliser dans les règles CSS est `_sm`, pour *small*.
- `@media screen and (max-width: 35.5em)` : pour les écrans ayant une largeur maximale de 35.5 em. Le suffixe à utiliser dans les règles CSS est `_xs`, pour *extra-small*.

3. Placer trois blocs dans la grille

La grille de **Gridlex** est basée sur douze colonnes. Les lignes doivent utiliser la classe `.grid-*`. Le suffixe `*` indique le nombre de blocs souhaité. Si nous souhaitons avoir trois blocs, la classe à indiquer est donc `grid-3`. Les trois blocs seront chacun placés sur quatre colonnes : $12 \text{ (colonnes de la grille)} / 3 \text{ (blocs voulus)} = 4 \text{ (colonnes pour chaque bloc)}$.

Chaque bloc de la ligne doit utiliser la classe `col`.

Voici un exemple très simple d'une ligne comportant trois blocs :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <title>Gridlex</title>
  <meta charset="UTF-8" />
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/
ajax/libs/gridlex/2.2.3/gridlex.min.css">
</head>
<body>
<div class="grid-3">
  <div class="col">Donec sed odio dui...</div>
  <div class="col">Maecenas sed diam eget...</div>
  <div class="col">Integer posuere erat...</div>
</div>
</body>
</html>
```

Voici l'affichage obtenu :

Donec sed odio dui. Donec ullamcorper nulla non metus auctor fringilla. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec sed odio dui. Vestibulum id ligula porta felis euismod semper.	Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Nullam quis risus eget urna mollis ornare vel eu leo. Maecenas sed diam eget risus varius blandit sit amet non magna.	Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Nulla vitae elit libero, a pharetra augue. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nulla vitae elit libero, a pharetra augue.
---	--	---

Sur un écran de largeur moindre, la largeur des blocs est réduite :

Donec sed odio dui. Donec ullamcorper nulla non metus auctor fringilla. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec sed odio dui. Vestibulum id ligula porta felis euismod semper.	Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Nullam quis risus eget urna mollis ornare vel eu leo. Maecenas sed diam eget risus varius blandit sit amet non magna.	Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Nulla vitae elit libero, a pharetra augue. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nulla vitae elit libero, a pharetra augue.
---	--	---

Notez que pour cet exemple, la grille n'est pas responsive.

Le fichier de cet exemple à télécharger est **Chapitre-09/C09-B3.html**.

Si vous créez plus de blocs dans la ligne qu'il n'y a de blocs déclarés dans la classe de la ligne, les blocs supplémentaires iront à la ligne.

Dans cet exemple, nous déclarons trois blocs, mais il y en a quatre de créés dans la ligne :

```
<div class="grid-3">
  <div class="col">Donec sed odio dui...</div>
  <div class="col">Maecenas sed diam eget...</div>
  <div class="col">Integer posuere erat...</div>
  <div class="col">Aenean eu leo quam
</div>
```

Voici l'affichage obtenu :

Donec sed odio dui. Donec ullamcorper nulla non metus auctor fringilla. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec sed odio dui. Vestibulum id ligula porta felis euismod semper.	Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Nullam quis risus eget urna mollis ornare vel eu leo. Maecenas sed diam eget risus varius blandit sit amet non magna.	Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Nulla vitae elit libero, a pharetra augue. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nulla vitae elit libero, a pharetra augue.
Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Nullam id dolor id nibh ultricies vehicula ut id elit. Vestibulum id ligula porta felis euismod semper. Maecenas sed diam eget risus varius blandit sit amet non magna.		

Le quatrième bloc, celui en trop, passe automatiquement à la ligne suivante.

4. Créer une grille responsive avec les blocs

Nous reprenons ici l'exemple précédent, mais nous souhaitons avoir une grille responsive. Pour les écrans larges, nous avons trois blocs dans la ligne ; pour les écrans les plus petits, nous n'avons qu'un seul bloc dans la ligne. Les trois blocs sont alors les uns sous les autres.

Pour ce faire, il suffit d'ajouter `xs-1` à la classe précédente. Comme nous l'avons vu, `xs` indique le point de rupture pour les écrans `extra-small` et `1` indique que nous souhaitons avoir une colonne dans la ligne.

Les deux composantes du nom de la classe sont à séparer par le caractère underscore `_`. Voilà le nom de la classe à utiliser : `class="grid-3_xs-1"`.

Voici la syntaxe complète :

```
<div class="grid-3_xs-1">
  <div class="col">Donec sed odio dui...</div>
  <div class="col">Maecenas sed diam eget...</div>
  <div class="col">Integer posuere erat...</div>
</div>
```

L'affichage sur un grand écran est identique à l'exemple précédent. Voici l'affichage sur un écran de largeur très petite, sous le point de rupture :

Donec sed odio dui. Donec ullamcorper nulla non metus auctor fringilla. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec sed odio dui. Vestibulum id ligula porta felis euismod semper.

Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Nullam quis risus eget urna mollis ornare vel eu leo. Maecenas sed diam eget risus varius blandit sit amet non magna.

Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Nulla vitae elit libero, a pharetra augue. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nulla vitae elit libero, a pharetra augue.

Le fichier de cet exemple à télécharger est **Chapitre-09/C09-B4.html**.