

Grillade by KNACSS

[À propos](#)

[Gabarits avec Schnaps.it](#) ↗

[es]

Une grille CSS simple et fluide

"Grillade" est un système de grille simple et élaboré avec CSS3 Flexbox. Il est intégré par défaut dans le micro-framework KNACSS d'Alsacrations mais peut être utilisé de manière autonome sans nécessiter KNACSS ou d'autre dépendance.

L'objectif avoué de Grillade est de demeurer le plus léger possible (contrairement à des grilles complexes de frameworks que l'on ne nommera pas)... tout en proposant l'ensemble des fonctionnalités attendues d'une grille : responsive webdesign, possibilité de préciser le nombre de colonnes, taille de gouttière, appliquer des offsets (pull, push), réordonner les éléments, dimensionner individuellement chaque enfant, et réaliser ses propres grilles à l'aide de mixins Sass.

 **KNACSS et Grillade sont distribués librement [sur Github](#) sous [licence WTFPL](#), toutes contributions sont bienvenues.**



Grillade CSS
CSS minifié (10ko)



Grillade SCSS
version Sass (3ko)

[Lire la documentation](#)



Testez Grillade directement en ligne [sur CodePen](#) !



Extensible

Simplissime par défaut (une simple classe `.grid` sur un conteneur pour l'activer), Grillade offre cependant de nombreuses possibilités et d'options de configuration pour chacun de vos projets.



Mobile First

Par défaut, Grillade n'est activée que si la taille d'écran est supérieure à celle d'un smartphone. Ainsi les éléments demeurent dans leur état naturel (sur une colonne en général) sur un petit device.



Compatible Sass

Pensé pour être couplé avec des préprocesseurs tels que Sass, Grillade prévoit un fichier de configuration avec toutes vos variables de projet.

En action !

Voici à présent la documentation illustrée de Grillade, moultement agrémentée de codes source.

1. [Préambule](#)
2. [Une grille simple, monoligne](#)
3. [La même avec gouttière](#)
4. [Dimensionner les enfants](#)
5. [Une grille multi-lignes](#)
6. [Les offsets \(pull et push\)](#)
7. [Réordonner les éléments](#)
8. [Inverser la grille](#)
9. [Taille d'écran intermédiaire](#)
10. [Votre grille avec des mixins Sass](#)

Préambule

Voici quelques dernières informations avant de commencer.

- Grillade est basé sur CSS3 Flexbox, ce qui le rend incompatible avec les anciennes versions d'Internet Explorer (IE9 inclus) ainsi que certains anciens navigateurs (Android <4.4 par exemple)
- Les préfixes CSS3 (`-webkit-` , `-moz-` , `-ms-` , ...) sont pris en charge au sein de la version CSS de Grillade, mais pas dans sa version Sass. Il vous sera donc nécessaire de les ajouter, de préférence automatiquement grâce à l'excellent outil [autoprefixer](#).
- Les valeurs par défaut des points de rupture responsive (Breakpoints) de Grillade sont : `$tiny = 544px` et `$small = 768px` , vous pouvez modifier ou ajouter des valeurs dans la version Sass.

Une grille simple, mono-ligne

Appliquez simplement la classe `.grid` sur un conteneur pour obtenir une grille mono-ligne.

Les enfants, quels qu'ils soient et quel que soit leur nombre, s'afficheront de manière équitable les uns à côté des autres.

```
<div class="grid"> <!-- parent (div ou autre) -->
  <div>...</div>
  <div>...</div>
  <aside>...</aside>
</div>
```

Le résultat :

je suis un <div>, sans class

hey moi aussi

moi je suis un <aside>

La même avec gouttière

L'ajout de la classe `.has-gutter` crée un espace entre les enfants

Cet espace est configurable dans la version `.scss` de Grillade. La valeur par défaut de la gouttière est `1rem` mais il est parfaitement possible de la modifier en optant pour la valeur et l'unité qui vous siéra (pixel, em, rem, %).

```
<div class="grid has-gutter">
  <div>...</div>
  <div>...</div>
```

```
<div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack

Lorem Elsass ipsum Choucroute knack

Lorem Elsass

La taille de la gouttière est configurable à l'aide de classes additionnelles : `.has-gutter-l` (gouttière x2) ou `.has-gutter-xl` (gouttière x4).

```
<div class="grid has-gutter-xl">
  <div>...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute
knack

Lorem Elsass ipsum Choucroute
knack

Lorem Elsass

Dimensionner les enfants

Il est possible, en option, de dimensionner chaque enfant par rapport à l'espace contenu au sein du parent, grâce aux classes `.one-half`, `.one-third`, `.one-quarter`, `.one-fifth`, `.two-thirds`, `.three-quarters`, `.one-sixth`, `.five-sixths` ou `.full`

```
<div class="grid">
  <div class="one-fifth">...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

`.one-fifth`

Lorem Elsass

Cela fonctionne également avec l'indication de gouttière `.has-gutter`

```
<div class="grid has-gutter">
  <div class="one-fifth">...</div>
  <div>...</div>
  <div class="one-fifth">...</div>
</div>
```

Le résultat :

.one-fifth

Lorem Elsass

.one-fifth

Une grille multi-lignes

À partir de l'objet `.grid`, il suffit d'ajouter un suffixe `-N` (par exemple `-3`) pour passer en **mode multi-lignes**. Les valeurs du suffixe, et donc le nombre de colonnes peuvent aller de 2 à 12.

```
<div class="grid-3">
  <div>...</div>
  <div>...</div>
  <div>...</div>
  <div>...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack

Lorem Elsass ipsum Choucroute knack

Lorem Elsass

knack

Lorem Elsass

Cela fonctionne également avec l'indication de gouttière `.has-gutter`

```
<div class="grid-3 has-gutter">
  <div>...</div>
  <div>...</div>
```

```

    <div>...</div>
    <div>...</div>
    <div>...</div>
</div>

```

Le résultat :

Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack	Lorem Elsass
knack	Lorem Elsass	

Cela fonctionne également avec des enfants dimensionnés explicitement.

```

<div class="grid-3">
  <div class="one-half">...</div>
  <div class="one-half">...</div>
  <div class="one-third">...</div>
  <div class="two-thirds">...</div>
  <div class="one-quarter">...</div>
</div>

```

Le résultat :

Lorem Elsass	Lorem Elsass ipsum Choucroute knack	Choucroute knack
.one-half		.one-half
.one-third	.two-thirds	
.full		
.one-quarter	.one-quarter	.one-half
.one-sixth	.one-sixth	.two-thirds

Les offsets (pull et push)

Ajoutez les classes `.pull` ou `.push` pour pousser un enfant vers sa droite ou sa gauche.

```
<div class="grid-3">
  <div>...</div>
  <div>...</div>
  <div class="push">...</div>
  <div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack	Lorem Elsass
	.push	Lorem Elsass

Réordonner les éléments

Les classes `.grid-item-first` et `.grid-item-last` servent à réordonner visuellement les éléments.

```
<div class="grid-3">
  <div>...</div>
  <div>...</div>
  <div class="grid-item-first">...</div>
</div>
```

Le résultat :

.grid-item-first	Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack
------------------	-------------------------------------	-------------------------------------

Inverser la grille

Il est possible d'inverser totalement le sens de lecture de la grille grâce au suffixe `--reverse`

```
<div class="grid-3--reverse">
  <div>...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

troiz	deuz	preums
-------	------	--------

Taille d'écran intermédiaire

Vous pouvez indiquer le nombre de colonnes souhaitées lorsque la taille d'écran est intermédiaire (entre *tiny* et *small*, soit entre 545px et 768px par défaut) à l'aide d'un suffixe `-small-N` (avec N entre 1 et 4).

Par exemple, un conteneur `.grid-3-small-2` disposera de 1 colonne en mobile (cas par défaut), puis 2 colonnes en écran intermédiaire, puis 3 colonnes sur grand écran.

```
<div class="grid-3-small-2">
  <div>...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack	Lorem Elsass
Lorem	Lorem Elsass	

Cela fonctionne avec des gouttières (classe `.has-gutter`) ainsi qu'avec des dimensions explicites sur les enfants

```
<div class="grid-3-small-2 has-gutter">
  <div>...</div>
```

```
<div>...</div>
<div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack	Lorem Elsass
.one-quarter	.three-quarters	

Votre grille avec des mixins Sass

Le préprocesseur Sass (version `.scss`) permet d'étendre les possibilités de Grillade, voire de concevoir votre propre grille. Pour commencer, il est très facilement possible de modifier les variables de taille de gouttière (`$grid-gutters`).

Vous pouvez également concevoir vos propres colonnes. Le mixin suivant, appliqué à votre classe `.grid-perso` construira une grille de 4 colonnes sans gouttière : `.grid-perso { @include grid(4, 0); }`

```
<div class="grid-perso">
  <div>...</div>
  <div>...</div>
  <div>...</div>
</div>
```

Le résultat :

Lorem Elsass ipsum Choucroute knack	Lorem Elsass ipsum Choucroute knack	Lorem Elsass	Lorem
Lorem Elsass			

Avec gouttière personnalisée, cela pourrait donner : `.grid-perso { @include grid(4, 1rem); }`

```
<div class="grid-perso">
  <div>...</div>
```

```
<div>...</div>  
<div>...</div>  
</div>
```

Le résultat :

Lorem Elsass ipsum
Choucroute knack

Lorem Elsass ipsum
Choucroute knack

Lorem Elsass

Lorem

Lorem Elsass

Et maintenant, à vous de jouer !

À propos de Grillade

Grillade est un outil construit avec amour et du CSS par [Raphaël Goetter](#) et [Alsacrérations](#).

Alsacrérations est une innovante agence web alsacienne créée en 2006 qui répond à tous types de projets (développement, intégration, accessibilité, mobilité).

Un grand merci à [Matthieu Bousendorfer](#) pour le design du site web, à [Stéphanie Walter](#) pour la conception du logo, ainsi qu'à [Hugo Giraudel](#), [Philippe Vayssière](#), [Xavier Zawala](#), [Nicolas Hoffmann](#) pour leurs conseils et astuces pour faire évoluer l'outil.