

Pagination

Si vous avez beaucoup de données à afficher sur une seule page, il est préférable d'utiliser une pagination.

Usage

::demo Configurez `layout` avec les différent éléments de pagination que vous souhaitez, séparés par des virgules. Les éléments possibles sont: `prev` (pour aller vers la page précédente), `next` (pour aller vers la page suivante), `pager` (liste des pages), `jumper` (un champ pour aller directement à une page précise), `total` (Le nombre total d'entrées), `size` (un sélecteur qui détermine la taille de la page) et `->` (chaque élément après celui-ci sera poussé vers la droite).

```
<div class="block">
  <span class="demonstration">Quand vous avez quelques pages</span>
  <el-pagination
    layout="prev, pager, next"
    :total="50">
  </el-pagination>
</div>
<div class="block">
  <span class="demonstration">Quand vous avez plus que 7 pages</span>
  <el-pagination
    layout="prev, pager, next"
    :total="1000">
  </el-pagination>
</div>
```

...

Nombre de pages

::demo Par défaut, Pagination réduit les boutons lorsqu'il y a plus de 7 pages. Ce nombre est configurable avec l'attribut `pager-count`.

```
<el-pagination
  :page-size="20"
  :pager-count="11"
  layout="prev, pager, next"
  :total="1000">
</el-pagination>
```

...

Boutons avec couleur de fond

::demo Réglez l'attribut `background` pour change la couleur de fond des boutons.

```
<el-pagination
  background
  layout="prev, pager, next">
```

```
:total="1000">
</el-pagination>
```

...

Petite pagination

Utilisez une pagination de taille réduite si vous manquez d'espace.

:::demo Ajoutez simplement l'attribut `small` et la pagination sera de taille réduite.

```
<el-pagination
  small
  layout="prev, pager, next"
  :total="50">
</el-pagination>
```

...

Plus d'éléments

Vous pouvez ajouter plus de modules suivant vos besoins.

:::demo Cet exemple contient un use-case complet. Il utilise les événements `size-change` et `current-change` pour gérer la taille de la page et son changement. `page-sizes` accepte un tableau d'entiers, chacun représentant une taille dans les options de `sizes`, e.g. `[100, 200, 300, 400]` indique que le sélecteur aura quatre options: 100, 200, 300 ou 400 entrées par page.

```
<template>
  <div class="block">
    <span class="demonstration">Total d'entrées</span>
    <el-pagination
      @size-change="handleSizeChange"
      @current-change="handleCurrentChange"
      :current-page.sync="currentPage1"
      :page-size="100"
      layout="total, prev, pager, next"
      :total="1000">
    </el-pagination>
  </div>
  <div class="block">
    <span class="demonstration">Changement de taille de la page</span>
    <el-pagination
      @size-change="handleSizeChange"
      @current-change="handleCurrentChange"
      :current-page.sync="currentPage2"
      :page-sizes="[100, 200, 300, 400]"
      :page-size="100"
      layout="sizes, prev, pager, next"
      :total="1000">
    </el-pagination>
  </div>
</template>
```

```

<div class="block">
  <span class="demonstration">Aller à</span>
  <el-pagination
    @size-change="handleSizeChange"
    @current-change="handleCurrentChange"
    :current-page.sync="currentPage3"
    :page-size="100"
    layout="prev, pager, next, jumper"
    :total="1000">
  </el-pagination>
</div>
<div class="block">
  <span class="demonstration">Tous en même temps</span>
  <el-pagination
    @size-change="handleSizeChange"
    @current-change="handleCurrentChange"
    :current-page.sync="currentPage4"
    :page-sizes="[100, 200, 300, 400]"
    :page-size="100"
    layout="total, sizes, prev, pager, next, jumper"
    :total="400">
  </el-pagination>
</div>
</template>
<script>
  export default {
    methods: {
      handleSizeChange(val) {
        console.log(`${val} items per page`);
      },
      handleCurrentChange(val) {
        console.log(`current page: ${val}`);
      }
    },
    data() {
      return {
        currentPage1: 5,
        currentPage2: 5,
        currentPage3: 5,
        currentPage4: 4
      };
    }
  }
</script>

```

...

Cacher la pagination s'il n'y a qu'une seule page

Lorsqu'il n'y a qu'une seule page, il est possible de cacher la pagination avec l'attribut `hide-on-single-page`.

...demo

```

<div>
  <el-switch v-model="value">
  </el-switch>
  <el-pagination
    :hide-on-single-page="value"
    :total="5"
    layout="prev, pager, next">
  </el-pagination>
</div>

<script>
  export default {
    data() {
      return {
        value: false
      }
    }
  }
</script>

```

...

Attributs

Attribut	Description	Type	Valeurs acceptées	Défaut
small	Si la pagination doit être petite.	boolean	—	false
background	Si les boutons doivent avoir une couleur de fond.	boolean	—	false
page-size	Nombre d'entrées sur chaque page, supporte le modificateur .sync.	number	—	10
total	Nombre total d'entrées.	number	—	—
page-count	Nombre de pages. Réglez total ou page-count et le nombre de pages sera affiché; si vous avez besoin de page-sizes, total est requis.	number	—	—
pager-count	Nombre de sélecteurs de pages. Pagination se réduit lorsque le nombre de pages dépasse cette valeur.	number	Nombre impair entre 5 et 21.	7
current-page	Le numéro de page courant, supporte le modificateur .sync.	number	—	1
layout	L'ensemble des éléments de la pagination, séparés par des virgules.	string	sizes, prev, pager, next, jumper, ->, total, slot	'prev, pager, next, jumper,

				->, total'
page-sizes	Options pour la taille des pages.	number[]	—	[10, 20, 30, 40, 50, 100]
popper-class	Classe du menu de sélection de la taille des pages.	string	—	—
prev-text	Texte du bouton prev.	string	—	—
next-text	Texte du bouton next.	string	—	—
disabled	Si la pagination est désactivée.	boolean	—	false
hide-on-single-page	Si la pagination doit être cachée quand il n'y a qu'une seule page.	boolean	—	-

Évènements

Nom	Description	Paramètres
size-change	Se déclenche quand <code>page-size</code> change.	La nouvelle taille.
current-change	Se déclenche quand <code>current-page</code> change.	La nouvelle page courante
prev-click	Se déclenche quand le bouton prev est cliqué et que la page courante change.	La nouvelle page courante.
next-click	Se déclenche quand le bouton next est cliqué et que la page courante change.	La nouvelle page courante.

Slot

Name	Description
—	Contenu personnalisé. Pour l'utiliser vous devez déclarer <code>slot</code> dans <code>layout</code> .