

# Virtual pagination (VP)

Ce tutoriel vient en complément des tutoriels des add-ons.

Vous pouvez télécharger les tutoriels des add-ons [ici](#).

## Qu'est-ce que Virtual pagination ?

[Virtual Pagination](#) vous permet de transformer de longs contenus en une série de pages accessibles via des liens de pagination **1** **2** **3** **4** **5** .

Dans Hot Potatoes vous pouvez utiliser Virtual pagination :

- dans le document ressource pour toutes les applications (sauf le Masher), y compris les add-ons (sauf JMem).  
[Exemple 1](#)
- dans le texte de l'exercice pour JCloze, in the exercise text for JCloze, y compris les add-ons (sauf FIb).  
[Exemple 2](#).

## Comment utiliser Virtual pagination quand on crée un exercice ?

Pour utiliser Virtual pagination il faut :

- **placervirtualpaginate.js** and **virtualpagination.css** dans le répertoire de l'exercice. Vous les trouverez dans le répertoire resources.
- utiliser les fichiers source modifiés (répertoire source\_XXX\_\*\*VP\*\*).
- mettre chaque portion de texte entre :
  - \* `<div class="virtualpage">` et `</div>` pour le texte du **conteneur d'exercice**.
  - \* `<div class="virtualpage_rt">` et `</div>` pour le texte du **conteneur de ressource**.

Voir les fichiers de données dans [demos VP.zip](#)

**Note:** n'insérez pas un retour chariot entre les divs, sinon les portions seront décalées d'une ligne vers le bas quand vous cliquerez sur le lien de pagination.

Par exemple n'écrivez pas :

```
<div class="virtualpage">texte de la portion1</div>
<div class="virtualpage">texte de la portion2</div>
```

mais :

```
<div class="virtualpage">texte de la portion1</div><div class="virtualpage">texte de la portion2</div>
```

# Comment modifier le style des liens de pagination ?

## 1- Couleurs

Les couleurs sont définies dans **virtualpagination.css**.

Ouvrez **virtualpagination.css** avec un éditeur de texte (Notepad++, Gedit etc. N'utilisez pas MS Word ou Libreoffice Writer).

Recherchez `paginationstyle a{ /*Pagination links style*/` et `.paginationstyle .flatview`

```
.paginationstyle a{ /*Pagination links style*/
padding: 0 5px;
text-decoration: none;
border: 1px solid black;
color: navy;
background-color: white;
}
```



```
.paginationstyle.flatview a:hover, .paginationstyle .flatview
a.selected{ /*Pagination div "flatview" links style*/
color: #000;
background-color: yellow;
}
```

Vous pouvez modifier les couleurs, par exemple (je sais que le résultat est hideux mais c'est à des fins pédagogiques !):

```
.paginationstyle a{ /*Pagination links style*/
padding: 0 5px;
text-decoration: none;
border: 1px solid red;
color: blue;
background-color: pink;
}
```



```
.paginationstyle .flatview a:hover, .paginationstyle .flatview
a.selected{ /*Pagination div "flatview" links style*/
color: LawnGreen;
background-color: Violet;
}
```

Vous pouvez modifier ces valeurs (vous pouvez trouver le nom ou le code hexadécimal des couleurs sur [wikipedia](https://fr.wikipedia.org/wiki/Liste_de_couleurs) ).

**N'oubliez pas de sauvegarder le fichier modifié.**

## 2- Texte

Ouvrez **virtualpaginate.js** avec un éditeur de texte (Notepad++, Gedit etc. N'utilisez pas MS Word ou Libreoffice Writer).

Cherchez flatviewhtml+= et modifiez la ligne :

```
flatviewhtml+= '<a href="#flatview" rel="'+i+'">'+(i+1)+'</a> ' //build sequential pagination links
```

Pour obtenir **Q1** **Q2** :

```
flatviewhtml+= '<a href="#flatview" rel="'+i+'">'+ 'Q'+(i+1)+'</a> ' //build sequential pagination links
```

Pour obtenir **page1** **page2** :

```
flatviewhtml+= '<a href="#flatview" rel="'+i+'">'+ 'page'+(i+1)+'</a> ' //build sequential pagination links
```

**N'oubliez pas de sauvegarder le fichier modifié.**