# Pagination documentation

## Installation

You are given a jquery.simplePagination.js file in the exam archive.

* **Add to project**
  + Copy the file from the archive to your project
* **Link in html**
  + Link the pagination script file in the index.html of your app

## Usage

* **Fix your 'list all notes' query** – when you are making a get request in order to get all notes, you should include in the query the amount of notes you want to show, the amount of notes you want to skip and the count of all notes. *Hint: use the "limit", "skip" , "count" URL parameters to achieve the effect.*
* **Create a pagination object** – after your request has return the resulting notes and after you rendered the "Office notes" or "My notes" page you should create a pagination object by adding the following code:

|  |
| --- |
| $('#pagination').pagination({  items: *totalNumberOfNotes*,  itemsOnPage: *ammountOfNotesPerPage*,  cssStyle: 'light-theme',  hrefTextPrefix: *the prefix of the current page. E.g. '#/office/'*  }).pagination('selectPage', *the current page*); |

Where *totalNumberOfNotes* is the total amount of notes that were to be received. *AmmountOfNotesPerPage* is the number of shown notes per page (in the exam case it should be 10). *hrefTextPrefix* is the URL prefix on which the pagination will work. E.g. '#/my-notes/' if you are on 'My notes' page or '#/office/' if you are on 'Office notes' page. *After* that the library automatically creates the pagination on the bottom of the page.

* **Changing pages** – when clicking on different pages from the pagination tool, your URL changes in the following way: If you click on page 1, your URL becomes the href prefix + 1. E.g. '#/office/1'. If you click on page 2, your URL becomes for example '#/office/2'. Each time you change page you need to render the template, afterwards initialize the pagination object and do not forget to call the 'selectPage' function so that the pagination is correctly showing the selected page. As shown in the example above, after creating the object you should chain a .pagination() function with 2 arguments: a string – 'selectPage' and a number – the current page.
* **Fix routing** – after you have done all the steps above, you should fix your routing listener (most likely Sammy.js) to listen not only for the route '#/office/' but also for the page number afterwards. Also when trying to acces the '#/office/' root without a page number, you should be redirected to the first page ('#/office/1'). The same applies for the 'My notes' page as well.