Python: list pagination

In this article we will see how to paginate a list in Python.

It involves using the comprehension to generate a list whose elements will be the pages that make up the main list, i.e. the sublists created through the slicing technique.

```
def paginate(items, per_page):
    pages = [items[i:i+per_page] for i in range(0,
len(items), per_page)]
    return {
        'total': len(items),
        'pages_no': len(pages),
        'pages': pages
    }
```

Our function will return a dictionary containing the total number of entries in the main list, the number of pages it was split into, and the list of pages created.

The `range()` function is also used here with the third parameter which indicates the number of steps per iteration. For example, supposing we have a list `items` of 100 elements and a parameter `per_page` equal to 10, at the first iteration we will have:

```
pages = [items[0:0+10]... range(0, 100, 10)]
```

And so on.