Python class *4*

Big Data

이번 수업에서는 빅데이터를 다루기 위한 첫 걸음을 내딛어 보도록 하겠습니다.

Naver검색결과 얻기

그 첫걸음으로는 하나의 url을 선정하여 해당 url의 데이터들을 가져오기를 해보겠습니다. 우선 웹의 url을 요청 위해 `urllib.request`를 `import`하고, 데이터를 다루기 위해 데이터 교환 형식의 일종인 `json` (JavaScript Object Notation)을 `import`합니다. 방금 `import` 한 것들은 앞으로 자주 반복해서 사용하게 되니 기억해두면 좋습니다.

```
import urllib.request
import json
```

그럼 이제 환경은 셋팅 되었으니 도구들을 만들어 보도록 하겠습니다. 만들기 전에, 먼저 우리가 무엇을 필요로 하게 될지 생각해보도록 하는 시간을 가져 보겠습니다.

우선 가장 먼저 데이터들이 있는 url을 요청해야 합니다. 그리고 요청해서 얻은 url로 부터 정보들을 가져옵니다. 정보들을 얻었으면 필요한 부분만 뽑아서 정리하고, 결과를 출력합니다. 출력은 새 파일을 만드는 걸로 하겠습니다.

url 요청 메소드

```
def get_request_url(url):
    req = urllib.request.Request(url)
    req.add_header("X-Naver-Client-id", "bg8bHjj3T_f8BIOx9EbI")
    req.add_header("X-Naver-Client-Secret", "y4BsA_TfTD")
    try:
        response = urllib.request.urlopen(req)
        if response.getcode() == 200:
            return response.read().decode('utf-8')
    except Exception as e:
        return None
```

검색 결과 얻는 메소드

```
def getNaverSearchResult(sNode, search_text, page_start, display):
    base ="https://openapi.naver.com/v1/search"
    node ="/%s.json" % sNode
    parameters = "?query=%s&start=%s&display=%s" % (urllib.parse.quote(search_text),
page_start, display)
    url = base+node+parameters
    retData = get_request_url(url)

    if (retData == None):
        return None
    else:
        return json.loads(retData)
```

포스트 데이터 얻는 메소드

```
def getPostData(post, jsonResult):
    title = post['title']
    description = post['description']
    org_link = post['originallink']
    link = post['link']

    pDate = post['pubDate']

    jsonResult.append({'title':title, 'description':description, 'org_link':org_link,
'link':link, 'pDate':pDate})
    return
```

메인 메소드

메인 메소드에서는 위 메소드들을 활용해서 json 데이터들을 정리해서 파일로 출력을 하게 됩니다.

```
def main():
    jsonResult = []

    sNode = 'news'
    search_text = '오재일'
    display_count = 100

    jsonSearch = getNaverSearchResult(sNode, search_text, 1, display_count)

    while((jsonSearch != None) and (jsonSearch['display'] != 0)):
        for post in jsonSearch['items']:
            getPostData(post, jsonResult)

        nStart = jsonSearch['start'] + jsonSearch['display']
        jsonSearch = getNaverSearchResult(sNode, search_text, nStart, display_count)

    with open('%s_naver_%s.json' % (search_text, sNode), 'w', encoding='utf8') as
outfile:
        retJson = json.dumps(jsonResult,
                             indent=4, sort_keys=True,
                             ensure_ascii=False)
        outfile.write(retJson)

    print ('%s_naver_%s.json SAVED' % (search_text, sNode))
```

```
if __name__ == '__main__':
    main()
```

```
오재일_naver_news.json SAVED
```

`main()` 메소드를 실행했더니 성공적으로 `오재일_naver_news.json` 이라는 파일이 생성되었습니다.