**[파이썬 기초](https://dataleader.tistory.com/category/%ED%8C%8C%EC%9D%B4%EC%8D%AC%20%EA%B8%B0%EC%B4%88)**

[**[파이썬(Python) 이야기 1화] 파이썬에서 엑셀(Excel) 데이터 불러오기**](https://dataleader.tistory.com/20)

데쟁이 2021. 5. 6. 09:10

**0. 파이썬에서 엑셀을 불러오기 위해서는 어떻게 해야할까?**

데이터 분석을 하기위해서, 다양한 작업을 해야하지만, 그 중에서 자주 반복적으로 접하게 되는 부분이 엑셀을 불러오는 일입니다. 사실 이부분은 그렇게 어렵지 않기 때문에 실습을 통해서 익혀보도록하겠습니다.

**1. 넌 엑셀에 대해서 얼마나 알고 있니?**

엑셀은 크게 2가지 구분이 존재합니다. xls, xlsx 둘이 차이가 존재합니다. xls는 초기 엑셀파일 형태이고 xlsx는 요즘 우리가 쓰는 형식입니다. 문제는 엑셀을 읽는 라이브러리가 파이썬 프로그램 상에서 구분된다는 점입니다.  xlsx를 기반으로 하는 엑셀 라이브러리를 이용해야, 요즘 쓰는 엑셀을 접근해 읽을 수 있습니다.

꼭 기억하세요, xls, xlsx를 읽는 라이브러리가 치아가 존재한다고~

**2. pandas를 이용해 엑셀 읽기 - 그럼 pandas는 어떤 것 일까?**

판다스는 구조화된 데이터나 표 형식을 데이터를 빠르고 쉽고 표현적으로 다루도록 설계된 고수준의  자료구조와 함수를 제공하는 라이브러리입니다. 2010년 처음 개발되어 파이썬에서 데이터 분석을 생산적으로 수행할 수 있도록 지원합니다.

padas가 가지는 자료구조는 크게 dataframe과 1차원 배열 객체인 Series 형태를 가집니다. 이러한 pandas는 데이터 처리에 매우 유연성을 가지고 있으며, 색인기능을 제공하여 데이터 변형, 자르기, 취합 그리고 데이터의 부분집합을 선택할 수 있도록 도와줍니다.

dataframe?

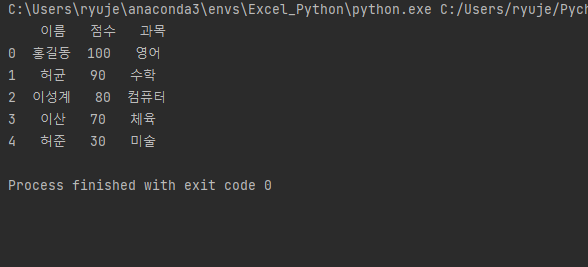
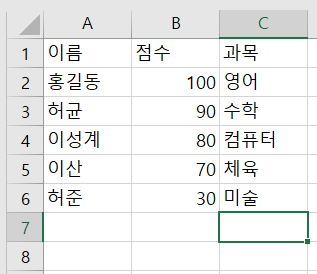
넵 맞습니다. pandas는 R 언어 에서 기본으로 제공하는 data.frame을 많은부분 준용해 python에서 유사하게 dataframe을 이용가능하도록 제공하는 라이브러리라고 생각하시면 매우 쉽습니다.

"R도 모르겠다" 하시는 분들은 일종의 엑셀 시트와 유사하느 자료구조 형태를  python에서 만들고 python 언어로, 이를  조작하는 것이 pandas라고 이해하시면 매우 쉽습니다. 데이터 공부를 하는 사람들이 pandas, pandas 하면 뭐지? 나만 모르나? 그러시지 마시고, "아~ python에서 다루는 dataframe 자료구조~ 별거 아니구먼~" 이 느낌으로 ~

자 이제 우리는 pandas에 대해서 알아보았습니다. 그럼 오늘의 목표인 엑셀을 pandas를 통해 읽어 보도록 하겠습니다.

(주의할 점) 다음 코드를 실행할 때 **openpyxl, xlrd**가 설치되지 않았거나 버전이 맞지 않는다는 애러 매시지가 발생될 수 있습니다. 이 부분에 대해서는 해당 라이브러리를 설치해 주시면 됩니다.

자 이제 우리가 읽을 엑셀파일의 형태를 보도록 하겠습니다.



자 이제 앞서 설명한 Python의 pandas로 읽어 보겠습니다. 소스코드는 아래와 같습니다.

**[소스코드]**

import pandas

dataframeXlsx = pandas.read\_excel("C:/Users/ryuje/Desktop/test.xlsx", sheet\_name="Sheet1"')  
print(dataframeXlsx)

해당 소스를 샐행하면 위의 결과와 같이 실행이 됩니다. 굉장히 쉽게 읽어지는 것을 알 수 있습니다. 물론 이를 다루는 것은 padas를 통해 좀 더 다뤄보아야 합니다.

3. 간단한 Dataframe 조작

사실 이러한 조작법은 dataframe을 사용하는데 지양해야하는 방법입니다. 왜냐하면, 데이터 프레임을 통해서, 통계적 산출을 바로 시행하기 위해서 dataframe 구조 자체를 이용하기 때문입니다. 다만, 프로그램을 작성하는 사람, 즉 프로그래머 입장에서는 선뜻 이 구조가 익숙하지 않습니다. 데이터분석 이전 시대에 프로그래머 분들은 더욱 그렇다고 생각하시면 됩니다.

하지만, 경우에 따라서는 for나 for each를 통해, 접근이 필요할 경우가 있습니다. 이럴때 어떻게 해야할 까요?

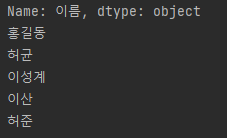
**[소스코드]**

import pandas

dataframeXlsx = pandas.read\_excel("C:/Users/ryuje/Desktop/test.xlsx", sheet\_name="Sheet1"')  
print(dataframeXlsx)

print(dataframeXlsx["이름"])  
for row in dataframeXlsx["이름"] :  
    print(row)

아래의 코드를 작성해 보면 결과는 다음과 같이 출력되는 것을 알 수 있습니다.



자 이제는 이름과 점수를 출력해보도록 할까요?

**[소스코드]**

import pandas

dataframeXlsx = pandas.read\_excel("C:/Users/ryuje\Desktop/test.xlsx", sheet\_name = "Sheet1")

print(dataframeXlsx) #프린트

print(dataframeXlsx["이름"])  
for row in dataframeXlsx["이름"] :  
    print(row)

for index, row in dataframeXlsx.iterrows() :  
    print(row["이름"], row["점수"])



위와 같이 결과가 산출되는 것을 확인하실 수 있습니다. 물론 zip를 사용해서 동일하게 구현할 수 있습니다.

for index, row in zip(dataframeXlsx["이름"], dataframeXlsx["점수"]) :  
    print(index, row)



3. 맺음말

이번 포스트에서는 pandas를 통해서, 엑셀을 읽는 법에 대해서 배워보았습니다. 또, 간단하게 padas도 실습해 보았든데요~ 사실 pandas는 좀 더 많은 연습이 필요합니다. 이후 포스트에서는 좀 더 관려 내용에 대해서 후속 포스팅을 통해 소개해보도록 하겠습니다. 그럼 오늘도 칼퇴하시고, 또 뵙겠습니다.

좋아요~ 구독~은 포스팅에 많은 도움이됩니다.

출처: <https://dataleader.tistory.com/20> [Data Miner]  
  
출처: <https://dataleader.tistory.com/20> [Data Miner]