UNISTUDY

파이썬

32강. 특수 파일 처리

1. 특수 파일 처리

- 엑셀(excel)은 행과 열의 교차되는 셀(cell)단위로 자료를 저장하는 응용 프로그램이다. 따라서 엑셀 문 서는 행 단위 또는 열 단위로 자료를 처리할 수 있다. 특히 열(칼럼) 단위로 처리할 경우 유용하다. 이절 에서는 열 단위로 자료를 처리할 수 있는 특수 파일들에 대해서 알아본다.

2. CSV, Excel 파일

- 열 단위로 자료를 처리할 수 있는 대표적인 파일은 CSV와 Excel 파일 등이 있다. 특히 CSV(Comma Separated Value) 파일은 각 항목을 콤마(,)로 구분하여 자료가 기록된 파일 형식으로 excel 응용프 로그램에 의해서 만들 수 있다.

3. CSV 파일 읽기

- 다음 예문은 pandas 패키지를 이용하여 CSV 파일을 가져와서 열 단위로 자료를 처리하는 과정이다.

3. CSV 파일 읽기

chapter08.lecture.step06_csv_excel_file.py

```
# (1) pandas 패키지 import
import pandas as pd
import os
# 현재 작업 디렉터리 확인
print(os.getcwd()) # D:\Pywork\workspace


# (2) csv 파일 읽기
score = pd.read_csv("chapter08/data/score.csv")
print(score.info()) # 파일 정보
print(score.head()) # 칼럼명 포함 앞부분 5개 행


# (3) 칼럼 추출
kor = score.kor # 객체.칼럼명
eng = score['eng'] # 객체['칼럼명']
mat = score['mat'] # 객체['칼럼명']
dept = score['dept'] # 객체['칼럼명']
```

Python Console

```
D:\Pywork\workspace

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15 entries, 0 to 14
Data columns (total 5 columns):
no      15 non-null int64
kor     15 non-null int64
eng     15 non-null int64
mat     15 non-null int64
dept    15 non-null int64
```

3. CSV 파일 읽기

```
# (4) 과목별 최고 점수
print('max kor = ', max(kor))
print('max eng = ', max(eng))
print('max mat = ', max(mat))
```

```
   no  kor  eng  mat  dept
0  101   88   85   86   101
1  102   70   90   92   101
2  103   90   45   55   101
3  104   55   52   74   102
4  105   65   85   96   102
```

```
# (5) 과목별 최하 점수
print('min kor = ', min(kor))
print('min eng = ', min(eng))
print('min mat = ', min(mat))
```

```
max kor =  95
max eng =  90
max mat =  96

min kor =  40
min eng =  45
min mat =  41
```

```
# (6) 과목별 평균 점수
from statistics import mean
print('국어 점수 평균 : ',
round(mean(kor),3))

print('영어 점수 평균 : ', round(mean(eng),3))
print('수학 점수 평균 : ', round(mean(mat),3))
```

```
국어 점수 평균 :  71.4
영어 점수 평균 :  67.933
수학 점수 평균 :  71.333
```

```
# (7) dept 빈도수
dept_count = {} # 빈 set

for key in dept :

    dept_count[key] = dept_count.get(key,
0) + 1

print(dept_count) # dict
```

```
{101: 5, 102: 5, 103: 5}
```

3. CSV 파일 읽기

csv 파일 처리 예
(1) pandas 패키지 import
pandas 패키지를 import하고 pd를 별칭으로 지정한다.
(2) csv 파일 읽기
pd.read_csv()함수를 이용하여 해당 경로의 score.csv 파일을 읽어서 score 객체를 생성한다. score 객체에서 호출할 수 있는 info()함수를 이용하여 파일의 정보를 확인한다. 출력된 파일의 정보에서 <class 'pandas.core.frame.DataFrame'> 는 pandas의 데이터프레임 객체라는 의미이 고, RangeIndex: 15 entries, 0 to 14 는 15개의 행 수를 의미한다. 그리고 Data columns (total 5 columns) 는 5개의 열 수를 의미한다. 5개의 열 이름은 no, kor, eng, mat, dept 이고 자료형은 모두 int64이다. 결국 score.csv 파일은 데이터프레임이라는 행렬구조이고 15행5열 (15x5)의 2차원 모양(shape)을 갖는 자료라는 의미이다.

3. CSV 파일 읽기

(3) 칼럼 추출
pandas의 데이터프레임은 열 단위로 자료를 처리하는데 유용하다. 데이터프레임에서 칼럼(열)을 추출하기 위해서 다음과 같은 2가지 형식을 이용한다. 국어(kor), 영어(eng), 수학(mat), 학과 (dept) 칼럼을 추출하고 있다.
데이터프레임.칼럼명 데이터프레임['칼럼명']
(4) 과목별 최고 점수
칼럼 단위로 추출한 국어, 영어, 수학 점수를 대상으로 max()함수를 이용하여 최고점수를 계산한다.
(5) 과목별 최하 점수
칼럼 단위로 추출한 국어, 영어, 수학 점수를 대상으로 min()함수를 이용하여 최하점수를 계산한다.

3. CSV 파일 읽기

(6) 과목별 평균 점수
칼럼 단위로 추출한 국어, 영어, 수학 점수를 대상으로 mean()함수와 round()함수를 이용하여 과목 별 평균점수를 계산한다.
(7) dept 빈도수
학과(dept) 칼럼은 101, 102, 103의 3개 학과를 갖는 범주형 칼럼이다. 각 범주별로 출현빈도수를 구하고 있다. dept_count의 빈 set이 for문에 의해서 키와 값 형식인 dict 객체가 생성된다. dict 객체에서 키는 학과의 범주이고 값은 학과의 출현빈도수가 된다.

4. Excel 파일 읽기

- 다음 예문은 pandas 패키지를 이용하여 Excel 파일을 읽어서 열 단위로 자료를 처리하는 과정이다.

4. Excel 파일 읽기

chapter08.lecture.step06_csv_excel_file.py

```
# (1) excel 파일 읽기
sam = pd.ExcelFile("chapter08/data/sam_
kospi.xlsx")

# (2) excel 파싱
kospi = sam.parse("sam_kospi")
print(kospi.info())

# (3) 칼럼 추출
high = kospi['High']
low = kospi['Low']

# (4) 평균 계산
from statistics import mean
print('high mean=', mean(high))
print('low mean=', mean(low))

# (5) 평균 계산
print('High mean :', high.mean())
print('Low mean :', low.mean())
```

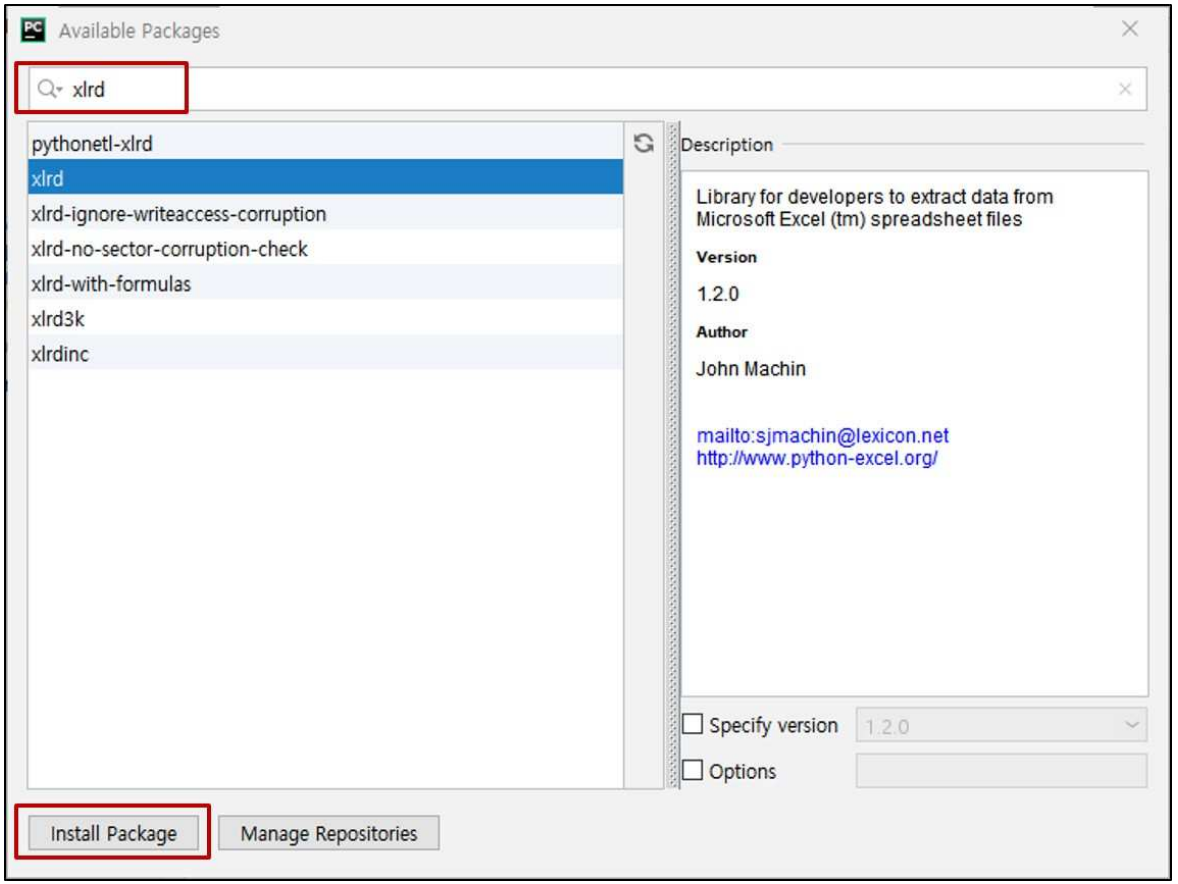
Python Console

```
ImportError: Missing optional d
ependency 'xlrd'. Install xlrd
>= 1.0.0 for Excel support Use
pip or conda to install xlrd.

<class
'pandas.core.frame.DataFrame
'>
RangeIndex: 247 entries, 0 to 2
46
Data columns (total 6 columns):
Date      247 non-null
datetime64[ns]
Open      247 non-null int64
High      247 non-null int64
Low       247 non-null int64
Close     247 non-null int64
Volume    247 non-null int64
dtypes: datetime64[ns](1), int
64(5)
memory usage: 11.6 KB
None

high mean= 1307947.3684210526
low mean= 1280919.028340081

High mean : 1307947.3684210526
Low mean : 1280919.028340081
```

4. Excel 파일 읽기

- excel 파일 처리 예
- # (1) excel 파일 읽기
- pd.ExcelFile() 함수를 이용하여 해당 경로의 엑셀 파일을 읽어오는 과정에서 'ImportError:' 라는 오류가 발생한다. 원인은 xlrd 패키지가 설치되지 않아서 발생된 오류이다. 이 오류를 해결하기 위해서 다음과 같이 xlrd 패키지를 설치하면 된다.

4. Excel 파일 읽기


Available Packages
xlrd
pythonetl-xlrd
xlrd
xlrd-ignore-writeaccess-corruption
xlrd-no-sector-corruption-check
xlrd-with-formulas
xlrd3k
xlrdinc
Description
Library for developers to extract data from Microsoft Excel (tm) spreadsheet files
Version
1.2.0
Author
John Machin
mailto:sjmachin@lexicon.net
http://www.python-excel.org/
Specify version
1.2.0
Options
Install Package
Manage Repositories

4. Excel 파일 읽기

- # (2) excel 파싱
- 엑셀 파일의 객체를 이용하여 엑셀의 시트(sheet)이름으로 파싱하면 pandas의 데이터프레임 객체가 생성된다. 그리고 데이터프레임 객체의 정보를 출력한다.
- # (3) 칼럼 추출
- 데이터프레임 객체에서 High, Low 칼럼을 추출한다.
- # (4) 평균 계산
- mean() 함수를 이용하여 High와 Low 칼럼의 평균을 계산한다.
- # (5) 평균 계산
- pandas 객체에서 제공하는 mean() 메서드를 호출하여 High와 Low 칼럼의 평균을 계산할 수 있다.
- ※ pandas 패키지는 2차원의 자료구조를 대상으로 자료를 처리할 수 있는 다양한 함수들을 제공한다. 본서에서는 pandas를 이용하여 칼럼단위로 자료를 제공하는 csv와 excel 파일을 대상으로 자료를 읽어오는 부분에 대해서만 제한한다.

UNISTUDY

THANK YOU