Pandas 라이브러리

천양하

목차

Pandas의 기초

- Pandas 개요
- Pandas 설치
- Pandas 활용

Pandas 라이브러리 개요

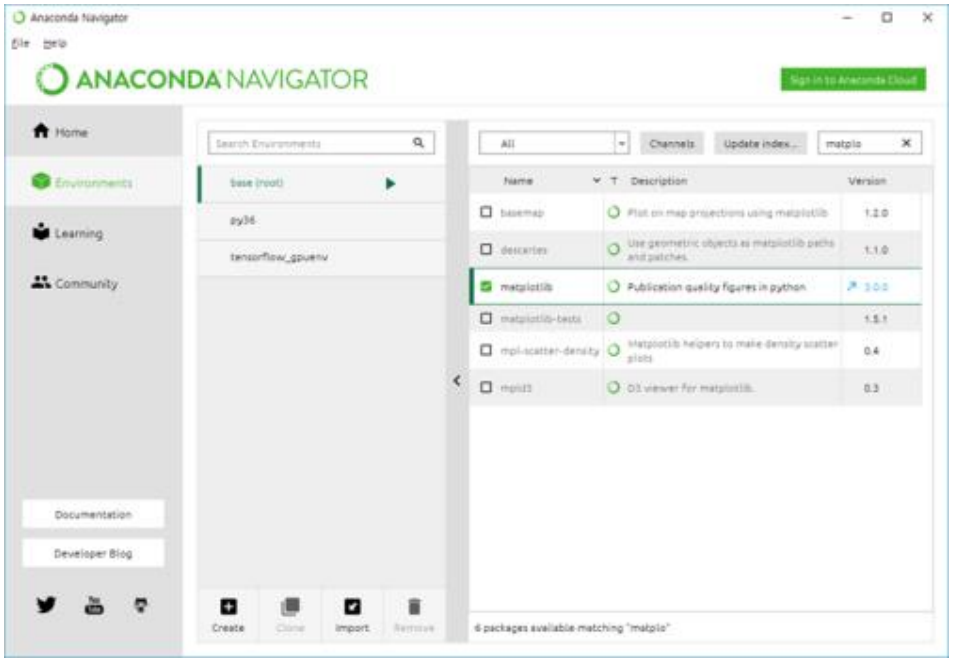
데이타 분석(Data Analysis)을 위해 널리 사용되는 파이썬 라이브러리 패키지

- pandas는 데이터를 불러오고, 전처리에 사용된다.
- pandas는 과학용 파이썬 배포판인 아나콘다(Anaconda)에 기본적으로 제공
- pandas는 CSV 파일, 텍스트 파일, 엑셀 파일, SQL 데이타베이스, HDF5 포맷 등 다양한 외부 리소스에 데이타를 읽고 쓸 수 있는 기능을 제공

Pandas 설치

아나콘다 내비게이터

- Environments 탭 선택
- All 선택하고 pandas 검색
- pandas 체크하고 설치



Colab 창

- 명령문 실행

명령문	pip list	# 설치된 라이브러리들을 보여줌
명령문	pip install pandas import pandas as pd	#라이브러리가 없다면 설치실행 #판다스 라이브러리 불러오기

pandas의 활용

- Pandas는 크게 세 가지의 자료구조를 지원한다.
  - 1차원 자료구조인 Series
  - 2차원 자료구조인 DataFrame
  - 3차원 자료구조인 Panel

- Pandas의 데이터형
  - Objects : 문자 또는 문자열 형
  - Int64 : 정수형
  - Float64 : 실수형

pandas의 사용 : Series

- 복수의 행(row)으로 이루어진 하나의 열(column) 구조
- 색인(index)을 가지고 원하는 데이터에 접근할 수 있음
- 자동으로 색인을 만들어 줌
- 이름을 pandas로 사용하기 너무 길어 pd로 대체

명령문	`import pandas as pd` `pd.Series([7, 3, 5, 8])`

결과	`0  7`
	`1  3`
	`2  5`
	`3  8`
	`dtype: int64`

pandas의 사용 : Series

- 자동으로 색인을 만들지 않고 index 키워드를 사용해 원하는 색인의 이름을 입력

명령문	x = pd.Series([7, 3, 5, 8], index=['서울', '대구', '부산', '광주']) print(x)

```
서울    7
대구    3
부산    5
광주    8
dtype: int64
```

pandas의 사용 : Series

색인을 나열하여 원하는 값 출력

명령문	x[['서울', '대구']]
결과	서울 7 대구 3 dtype: int64

pandas의 사용 : Series

Index

- 만들어진 시리즈에서 인덱스만을 출력

명령문	x.index
결과	Index(['서울', '대구', '부산', '광주'], dtype='object')

Values

- 만들어진 시리즈 데이터에서 값들만을 출력

명령문	x.values
결과	array([7, 3, 5, 8])

pandas의 사용 : Series

Sorted( ) 함수

- 인덱스나 값들로 정렬

명령문	print(sorted(x.index))
결과	['광주', '대구', '부산', '서울']

명령문	print(sorted(x.values))
결과	[3, 5, 7, 8]

명령문	x.reindex(sorted(x.index)) x
결과	서울 7 대구 3 부산 5 광주 8 dtype: int64

pandas의 사용 : Series

Series의 합

- 인덱스별로 저장된 값들의 합을 구함
- x와 y에 공통된 인덱스가 존재해야 더할 수 있으므로 광주와 대전은 NaN으로 표시

명령문	x=pd.Series([3, 8, 5, 9], index=['서울', '대구', '부산', '광주']) y=pd.Series([2, 4, 5, 1], index=['대구', '부산', '서울', '대전']) x+y

```
광주     NaN
대구    10.0
대전     NaN
부산     9.0
서울     8.0
dtype: float64
```

pandas의 사용 : Series

Unique( )

- 시리즈로부터 유일한 값들만을 반환

명령문	medal = [1, 3, 2, 4, 2, 3] x = pd.Series(medal) pd.unique(x)

```
array([1, 3, 2, 4])
```

명령문	medal = ['민준', '현우', '서연', '동현', '서연', '현우'] x = pd.Series(medal) pd.unique(x)

```
array(['민준', '현우', '서연', '동현'], dtype=object)
```

pandas의 사용 : DataFrame

- 2차원 배열
- 행(row)과 열(column)로 구성
- 열(column)에 대한 각각의 이름을 부여

명령문	import pandas as pd data= { 'age' : [23, 43, 12, 45], 'name' : ['민준', '현우', '서연', '동현'], 'height' : [175.3, 180.3, 165.8, 172.7] } x = pd.DataFrame(data, columns = ['name', 'age', 'height']) x

	name	age	height
0	민준	23	175.3
1	현우	43	180.3
2	서연	12	165.8
3	동현	45	172.7

pandas의 사용 : DataFrame

- **name 컬럼의 내용만 출력**

명령문	x.name

```
0    민준
1    현우
2    서연
3    동현
Name: name, dtype: object
```

pandas의 사용 : Panel

- 3차원 자료구조인 Panel은 Axis 0 (items), Axis 1 (major_axis), Axis 2 (minor_axis) 등 3개의 축을 가짐
- Axis 0은 그 한 요소가 2차원의 DataFrame 에 해당
- Axis 1은 DataFrame의 행(row)에 해당
- Axis 2는 DataFrame의 열(column)에 해당

pandas의 사용 : Panel

- **다음은 numpy를 사용하여 3차원 난수를 발생시킨 후, 이를 pandas.Panel() 에 적용한 예**
  - 2 (items) x 3 (major_axis) x 4 (minor_axis) 크기의 Panel 객체가 생성되었음을 알 수 있다.

```
import pandas as pd
import numpy as np

data = np.random.rand(2,3,4)
p = pd.Panel(data)
print(p)
```

```
<class 'pandas.core.panel.Panel'>
Dimensions: 2 (items) x 3 (major_axis) x 4 (minor_axis)
Items axis: 0 to 1
Major_axis axis: 0 to 2
Minor_axis axis: 0 to 3
```

pandas의 사용 : Panel

- Panel 객체 p로부터 p[0]을 조회하면, Axis 0 의 첫번째 요소인 DataFrame이 출력됨을 볼 수 있다.

```
p[0]
```

	0	1	2	3
0	0.825041	0.605136	0.084374	0.220112
1	0.635797	0.693157	0.120320	0.992910
2	0.532627	0.767363	0.208099	0.668378

pandas의 사용 : 외부데이터 가져오기

- 엑셀 파일로부터 데이타를 읽어 오는 기능

	A	B	C	D	E
1	ID	국어	영어	수학	
2	1	80	85	75	
3	2	90	100	95	
4	3	75	70	65	
5					
6					
7					

Test.xlsx

명령문	import pandas as pd from google.colab import files upload = files.upload() df = pd.read_excel('Test.xlsx')

명령문	df

	ID	국어	영어	수학
0	1	80	85	75
1	2	90	100	95
2	3	75	70	65

pandas의 사용 : 외부데이터 가져오기

엑셀에서 csv 또는 txt 파일을 가져와서 저장

- .csv 와 .txt 파일은 텍스트 파일임.
- 엑셀 실행 후 새 통합문서 열기
- A1셀 클릭 – 데이터 [탭] – 외부 데이터 가져오기 [그룹] – 텍스트 선택
- 텍스트 마법사 창에서 – 1단계 : 구분기호로 분리 됨 [원본파일 : 949 한국어]
- - 2단계 : 구분기호 : 쉼표 선택
- - 3단계 : 열 선택 : 열 데이터 서식 – 일반 – [마침]

pandas의 사용 : 외부데이터 가져오기

- **csv 또는 txt 파일을 가져와서 Pandas 형식으로 저장**

명령문	from google.colab import files upload = files.upload()

명령문	food = pd.read_csv('food.csv') food.head()

	Series_reference	Period	Data_value	STATUS	UNITS	Subject	Group	Series_title_1
0	CPIM.SE9S01	1999.06	645	REVISED	Index	Consumers Price Index - CPI	Food Price Index for New Zealand, Seasonally a...	Seasonally adjusted
1	CPIM.SE9S01	1999.07	647	REVISED	Index	Consumers Price Index - CPI	Food Price Index for New Zealand, Seasonally a...	Seasonally adjusted
2	CPIM.SE9S01	1999.08	645	REVISED	Index	Consumers Price Index - CPI	Food Price Index for New Zealand, Seasonally a...	Seasonally adjusted
3	CPIM.SE9S01	1999.09	644	REVISED	Index	Consumers Price Index - CPI	Food Price Index for New Zealand, Seasonally a...	Seasonally adjusted
4	CPIM.SE9S01	1999.10	641	REVISED	Index	Consumers Price Index - CPI	Food Price Index for New Zealand, Seasonally a...	Seasonally adjusted

pandas의 사용 : 외부데이터 가져오기

공공데이터

- www.data.go.kr 에서 교통사고에 대한 파일을 다운로드하여 acci.csv의 이름으로 파일을 저장

명령문	upload = files.upload()

명령문	accident = pd.read_csv('acci.csv') accident.head()
결과	에러발생!

명령문	accident = pd.read_csv('acci.csv', engine = 'python') accident.head()
결과	일부 한글이 깨져보임

pandas의 사용 : 외부데이터 가져오기

공공데이터

- 파일 인코딩 방법

명령문	accident = pd.read_csv('acci.csv', encoding = '949') accident.head()

명령문	accident = pd.read_csv('acci.csv', encoding = 'euc-kr') accident.head()

	법규위반	주야	발생건수	사망자수	부상자수	중상	경상	부상신고
0	과속	주	159	34	334	140	178	16
1	과속	야	218	73	348	200	139	9
2	교차로 통행방법 위반	주	8817	82	14031	3915	9530	586
3	교차로 통행방법 위반	야	5904	29	9728	2401	6884	443
4	기타	주	9388	141	14070	4271	9217	582

pandas의 사용 : 외부데이터 가져오기

- 공공데이터 www.data.go.kr
  - 전국 신규 민간 아파트 분양가격 동향

pandas의 사용 : 외부데이터 가져오기

공공데이터 www.data.go.kr

- 전국 신규 민간 아파트 분양가격 동향
- pc에 다운로드 한 파일의 이름을 영문이나 알아보기 쉬운 짧은 문자로 수정한다. (예 : 20200331.csv)
- 다음과 같이 코랩으로 업로드 한다

명령문	from google.colab import files upload = files.upload()

명령문	apt = pd.read_csv('20200331.csv', engine = 'python') apt.head()
결과	일부 한글이 깨져보임

- 파일을 인코딩하여 읽어들이기 쉽게 한다.('949' 또는 'euc-kr')

명령문	apt = pd.read_csv('20200331.csv', encoding = '949') apt.head(10)

THANK YOU FOR YOUR ATTENTION