파이썬 머신러닝 판다스 데이터분석

Lecture (4)

Dr. Heesuk Kim

목차

Part 1. 판다스 입문

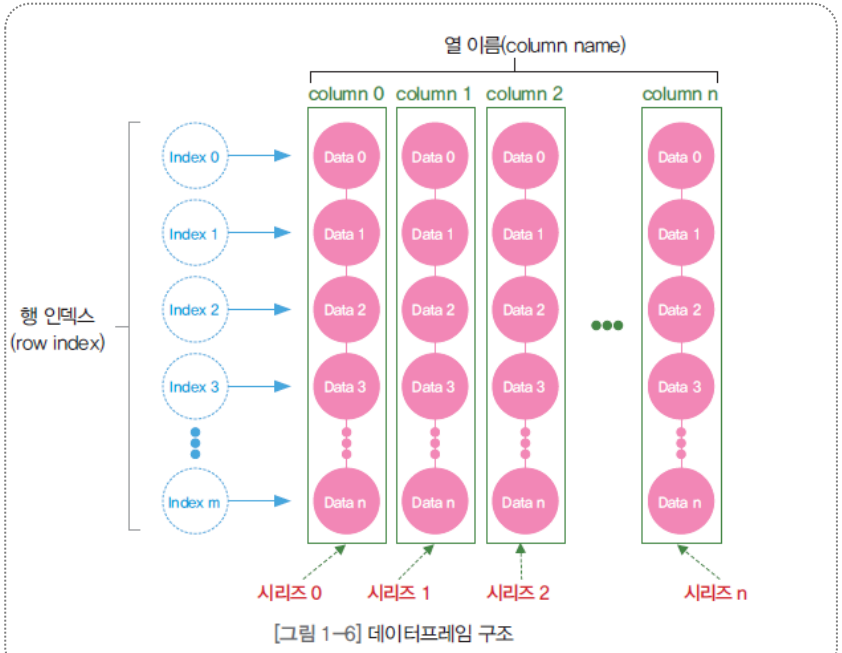
2-2. 데이터프레임

- **개요**

1) 데이터프레임은 2차원 배열. R의 데이터프레임에서 유래.

2) 데이터프레임의 열은 시리즈 객체. 시리즈를 열벡터(vector)라고 하면, 데이터프레임은 여러 개의 열벡터들이 같은 행 인덱스를 기준으로 줄지어 결합된 2차원 벡터 또는 행렬(matrix).

3) 데이터프레임은 행과 열을 나타내기 위해 두 가지 종류의 주소를 사용. 행 인덱스(row index)와 열 이름(column name 또는 column label)으로 구분.



[그림 1-6] 데이터프레임 구조

4) 데이터프레임의 각 열은 공통의 속성을 갖는 일련의 데이터를 나타냄.

5) 각 행은 개별 관측대상에 대한 다양한 속성 데이터들의 모음인 레코드(record).

[예시]

다음 주식종목 리스트에서, 각 행은 하나의 주식종목에 관한 관측값(observation)을 나타낸다.

각 열은 종목코드, 회사이름, 액면가, 총 주식수 등 공통의 속성이나 범주를 나타내는데, 보통 변수(variable)로 활용된다

열

종목 코드	회사 이름	액면가	총 주식수
005930	삼성전자	100원	5,970백만 주
017670	SK텔레콤	500원	81백만 주
005380	현대자동차	5000원	214백만 주

행

[표 1-1] 주식 종목 리스트

2-2. 데이터프레임

- **데이터프레임 만들기**

1) 같은 길이(원소의 개수가 동일한)의 배열 여러 개가 필요.

데이터프레임은 여러 개의 시리즈(열, column)를 모아 놓은 집합.

2) 판다스 DataFrame() 함수를 사용.

여러 개의 리스트를 원소로 갖는 딕셔너리를 함수에 전달하는 방식을 주로 활용.

딕셔너리 → 데이터프레임 변환: `pandas.DataFrame( 딕셔너리 객체 )`

3) 딕셔너리의 값(v)에 해당하는 각 리스트가 시리즈로 변환되어 데이터프레임의 각 열이 된다.

4) 딕셔너리의 키(k)는 각 시리즈의 이름으로 변환되어, 최종적으로 데이터프레임의 열 이름이 된다.

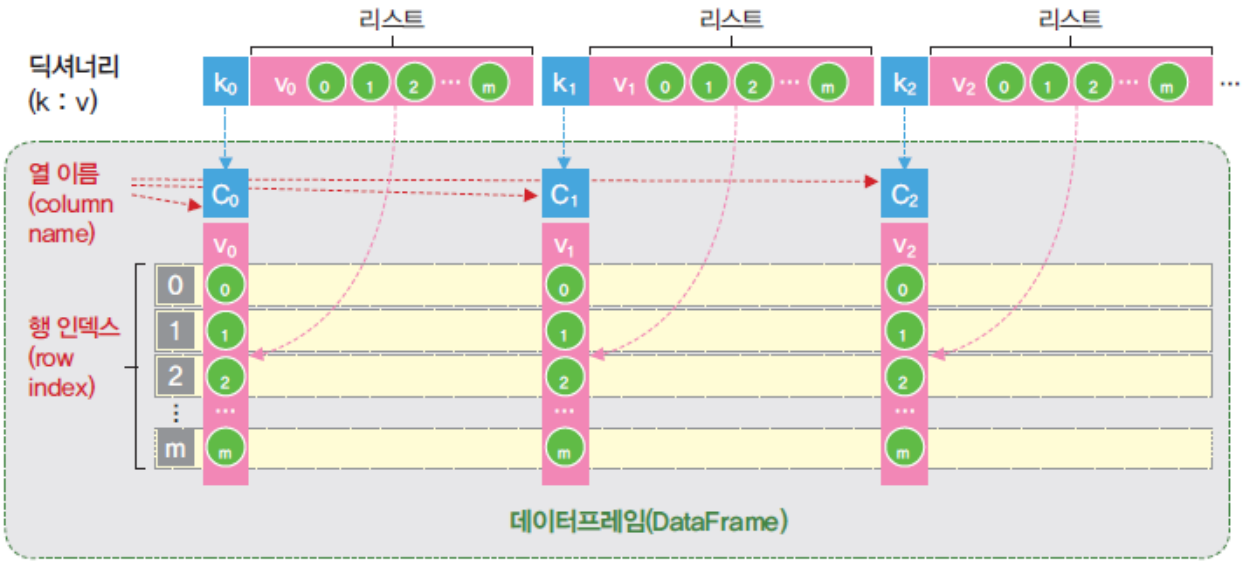
- 딕셔너리 (Dictionary) - 중괄호 { } 사용
  - ✓ key 와 value 를 1 : 1 로 대응시킨 형태이다.
  - ✓ 하나의 key 에는 하나의 value 만 대응된다.
  - ✓ key 값은 변하지 않고, value 값은 변경이 가능하다.
  - ✓ 예 : dict_data = { 'a' :1, 'b' : 2, 'c' : 3 }

2-2. 데이터프레임

- 리스트 (List) - 대괄호 [ ] 사용
  - ✓ 원소들이 연속적으로 저장되는 형태의 자료형이다.
  - ✓ 저장되는 요소들은 모두 같은 자료형일 필요는 없다.
  - ✓ 예 : list_data = ['2019-01-02', 3.14, 'ABC', 100, True]

딕셔너리 → 데이터프레임 변환: `pandas.DataFrame( 딕셔너리 객체 )`

dict_data = { 'a' :[1, 2, 3], 'b' : [10, 20, 30] , 'c' : [100, 200, 300] }



[그림 1-7] 딕셔너리 → 데이터프레임 변환

2-2. 데이터프레임

예제 1-4

원소 3개씩 담고 있는 리스트를 5개 만든다. 이들 5개의 리스트를 원소로 갖는 딕셔너리를 정의하고,
판다스 DataFrame() 함수에 전달하면 5개의 열을 갖는 데이터프레임을 만든다.

이때, 딕셔너리의 키(k)가 열 이름(c0 ~ c4)이 되고, 값(v)에 해당하는 각 리스트가 데이터프레임의 열이 된다.
행 인덱스에는 정수형 위치 인덱스(0, 1, 2)가 자동 지정된다.

〈예제 1-4〉 딕셔너리 → 데이터프레임 변환 (File: example/part1/1.4_dict_to_dataframe.py)

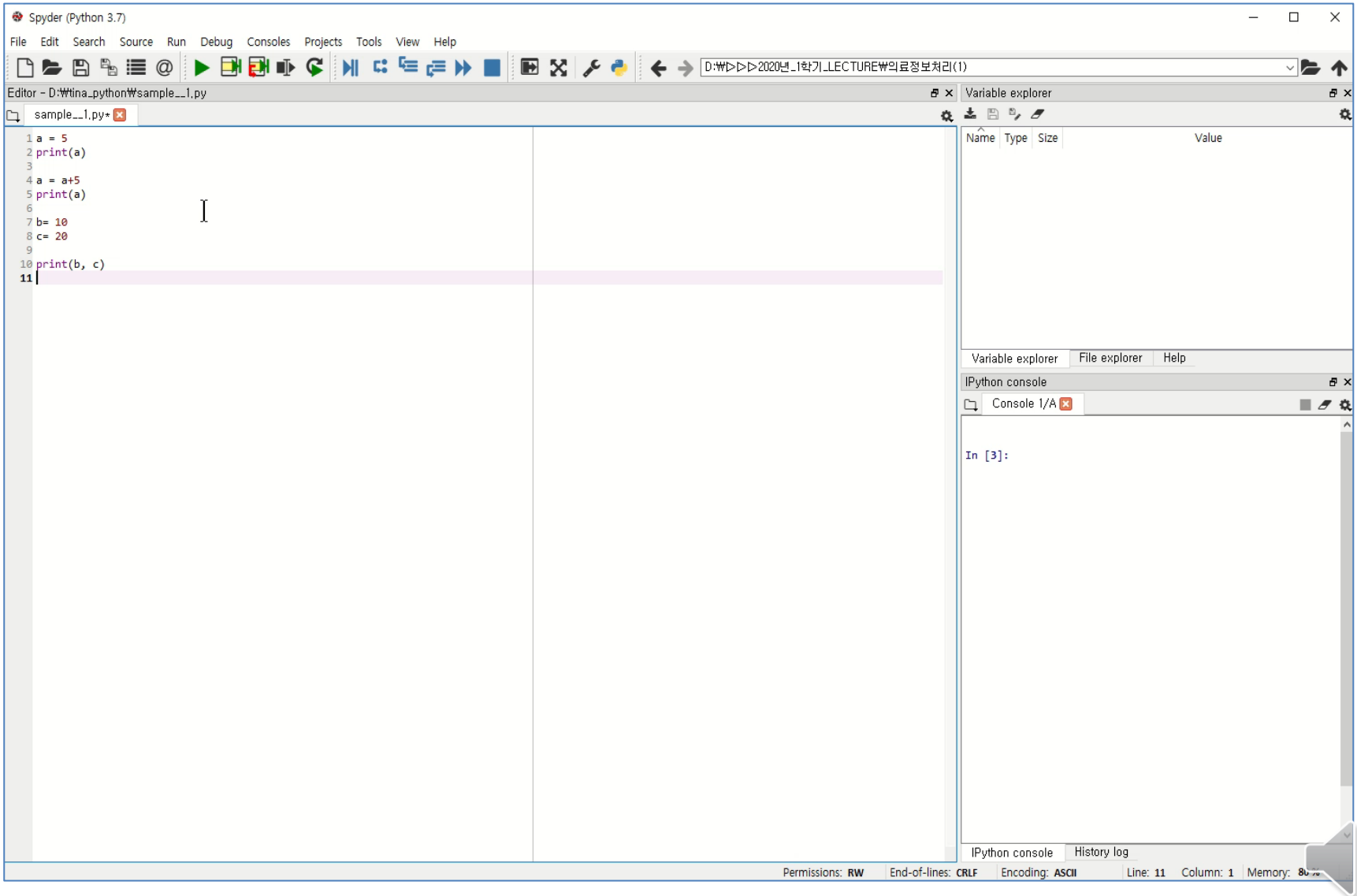
```
1  # -*- coding: utf-8 -*-
2
3  import pandas as pd
4
5  # 열이름을 key로 하고, 리스트를 value로 갖는 딕셔너리 정의(2차원 배열)
6  dict_data = {'c0':[1,2,3], 'c1':[4,5,6], 'c2':[7,8,9], 'c3':[10,11,12], 'c4':[13,14,15]}
7
8  # 판다스 DataFrame() 함수로 딕셔너리를 데이터프레임으로 변환. 변수 df에 저장
9  df = pd.DataFrame(dict_data)
10
11 # df의 자료형 출력
12 print(type(df))
13 print('\n')
14 # 변수 df에 저장되어 있는 데이터프레임 객체를 출력
15 print(df)
```

〈실행 결과〉 코드 전부 실행

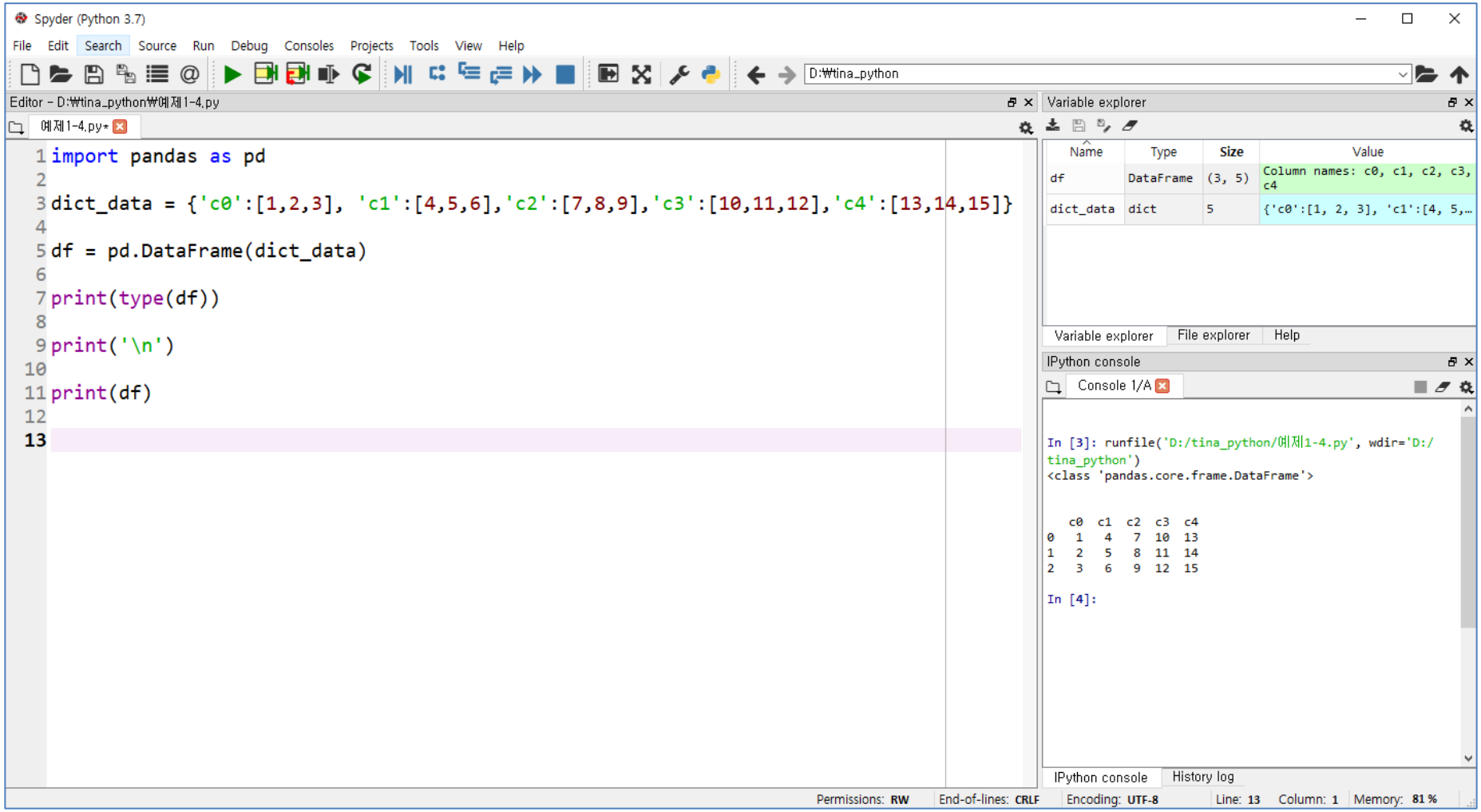
```
<class 'pandas.core.frame.DataFrame'>


   c0  c1  c2  c3  c4
0   1   4   7  10  13
1   2   5   8  11  14
2   3   6   9  12  15
```

2-2. 데이터프레임 예제 1-4



Part 0. 개발환경 준비



Any Questions?

Thank you.