

AIoT AutoCar Prime 으로 배우는 온디바이스 AI 프로그래밍

6.2 pandas

6.2.1 데이터 처리

pandas

- pandas(판다스) : 행과 열이 있는 데이터 테이블 사용
 - 색인별 데이터 조회, 조건별 데이터 조회 등 데이터 통계와 분석에 유용
 - 인공지능을 학습시키기 전, 데이터 필터링을 위해 사용
 - 대량의 데이터를 확인하기 쉬움

데이터 프레임

- Pandas는 데이터 테이블과 비슷한 개념으로 데이터를 관리
- 시리즈 : 인덱스와 값만 있는 1차원 테이블
- 데이터 프레임 : 열과 인덱스, 값이 있는 2차원 테이블

시리즈 생성

- series : 인덱스와 값만 갖는 1차원 구조
 - 파이썬의 딕셔너리와 비슷하게 인덱스를 참조해 값에 접근
 - pandas.Series메소드로부터 생성
 - 인덱스와 값 2개를 입력

```
01:         import pandas as pd
02:
03:         value = [32, 68, 220, 72]
04:         index = ["온도", "습도", "강수량", "불쾌지수"]
05:         sr = pd.Series(value, index=index)
06:
07:         print(type(sr))
08:         print(sr)
09:         print(sr["온도"])
```

시리즈 생성

- values : 시리즈가 갖고 있는 값 반환
- Index : 시리즈의 인덱스 반환

```
10:      print(sr.index)
11:      print(sr.values)
```

데이터 프레임 생성

- DataFrame : 인덱스와 값뿐만 아니라 열도 갖는 2차원 구조
 - 특정 값에 접근하려면 인덱스와 열을 동시에 참조해야 함
 - DataFrame[index][column] : 열과 인덱스를 지정하여 특정 값을 추출
 - pandas.DataFrame메소드로부터 생성
 - 파라미터로 값만 입력해 생성한 경우 인덱스와 열에 자동으로 순번을 지정

데이터 프레임 생성

```
01:         import pandas as pd
02:
03:         value = [[32, 68, 220, 72],
04:                  [28, 30, 0, 12],
05:                  [38, 81, 0, 91]]
06:         df = pd.DataFrame(value)
07:
08:         print(df)
09:         print(df[0][1])
```

데이터 프레임 생성

- 인덱스, 값, 열을 직접 지정하여 데이터 프레임을 생성 가능
 - 지정할 요소의 리스트를 파라미터로 입력

```
01: import pandas as pd
02:
03: value = [[32, 68, 220, 72],
04:          [28, 30, 0, 12],
05:          [38, 81, 0, 91]]
06: columns = ["온도", "습도", "강수량", "불쾌지수"]
07: index = ["초여름", "늦봄", "한여름"]
08: df = pd.DataFrame(value, index=index, columns=columns)
09:
10: print(df)
11: print(df["온도"]["늦봄"])
```

데이터 프레임 생성

- index : 각 행의 색인 반환
- columns : 각 열의 색인 반환
- values : 시리즈와 달리 2차원 배열로 값을 반환

```
01:         import pandas as pd
02:
03:         value = [[32, 68, 220, 72],
04:                  [28, 30, 0, 12],
05:                  [38, 81, 0, 91]]
06:         columns = ["온도", "습도", "강수량", "불쾌지수"]
07:         index = ["초여름", "늦봄", "한여름"]
08:         df = pd.DataFrame(value, index=index, columns=columns)
09:
10:         print(df.index)
11:         print(df.columns)
12:         print(df.values)
```

내용 정리

- Pandas : 데이터 통계와 분석에 유용한 패키지
 - ▣ 행과 열이 있는 데이터 테이블을 사용
 - ▣ 데이터 테이블과 비슷한 개념으로 데이터를 관리
- Series(list, index=list) : Pandas의 Series타입으로 생성
 - ▣ 각각 값과 인덱스를 담고있는 리스트 2개를 입력
 - ▣ values 속성 : 시리즈가 갖고 있는 값 반환
 - ▣ index 속성 : 인덱스를 반환합니다.

내용 정리

- DataFrame(list, index=list, columns=list) :
 - Pandas의 DataFrame타입으로 생성
 - 값, 인덱스, 열을 담고있는 리스트 3개를 입력
 - 파라미터로 값만 입력해 생성한 경우 인덱스와 열에 자동 순번 지정
 - index 속성 : 각 행의 색인 반환
 - columns 속성 : 각 열의 색인 반환
 - values : 배열로 값 반환

내용 정리

- `DataFrame.head(x)` : 데이터 프레임의 앞 부분 x개의 튜플 조회
- `DataFrame.tail(x)` : 데이터 프레임의 뒷 부분 x개의 튜플 조회
- `DataFrame[name]` : 데이터 프레임에서 특정 열을 조회

연습문제

□ 문제 19. 다음 코드를 읽고 출력을 작성해보세요.

```
01:         import pandas as pd
02:
03:         value = [[1, 2, 3, 5],
04:                  [7, 11, 13, 17],
05:                  [19, 23, 29, 31]]
06:
07:         df = pd.DataFrame(value)
08:
09:         print(df[2][2])
```

연습문제

□ 문제 20. 다음 코드를 읽고 출력을 작성해보세요.

```
01:         import pandas as pd
02:
03:         value = [1, 5, 9, 13]
04:         index = ["A", "B", "C", "D"]
05:         sr = pd.Series(value, index=index)
06:
07:         print(sr["C"])
```

연습문제

- 문제 21. 데이터 프레임을 사용해 다음과 같은 출력을 하는 코드를 작성해보세요.

[출력]

	A	B	C	D
a	0	2	4	6
b	1	3	5	7
c	2	4	6	8

연습문제

- 문제 22. 다음 코드는 Cds센서의 밝기 값을 출력하는 코드입니다.

```
01:         from pop import Cds,delay
02:
03:         cds = Cds(7)
04:
05:         arr1, arr2, arr3 = [], [], []
06:
07:         for i in range(3):
08:             arr1.append(cds.readAverage())
09:             delay(300)
10:             arr2.append(cds.readAverage())
11:             delay(300)
12:             arr3.append(cds.readAverage())
13:             delay(300)
14:
15:         arr = [arr1,arr2,arr3]
16:
17:         print(arr)
```

연습문제

- ▣ A. 코드를 실행시켜 결과를 확인해 보세요.
- ▣ B. arr를 value로 갖고, ['A' , 'B' , 'C']를 index로 갖는 시리즈를 생성해 보세요.
- ▣ C. arr를 value로 갖고, ['A' , 'B' , 'C']를 columns, ['D' , 'E' , 'F']를 index로 갖는 데이터 프레임을 생성해 보세요.