

2. 데이터 적재

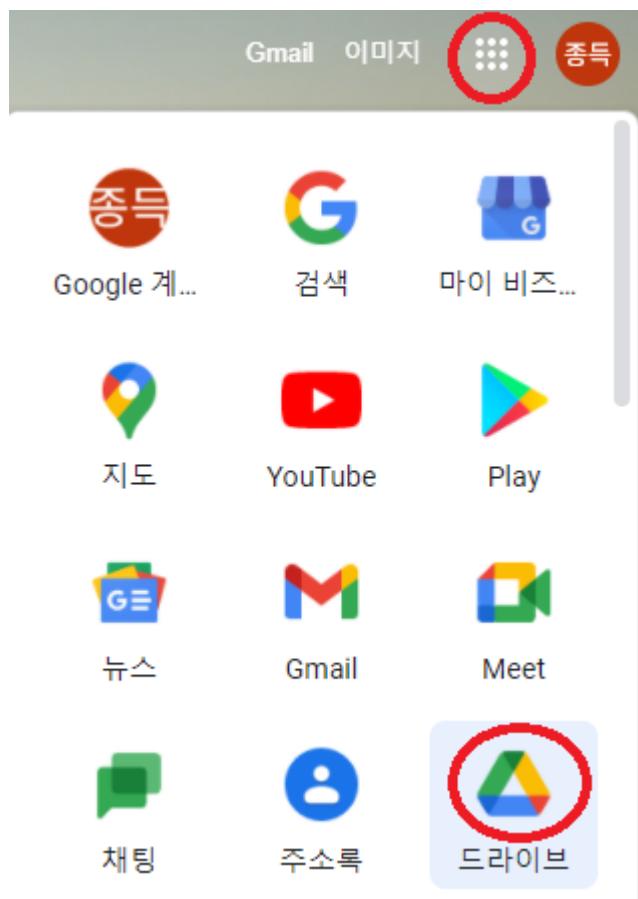
- (1) 샘플 데이터셋 적재하기
- (2) 모의 데이터셋 불러오기
- (3) CSV 파일 적재하기

엑셀 파일 중 CSV(comma-separated value) 타입을 불러오려면 판다스 라이브러리를 사용합니다. 판다스 라이브러리의 `read_csv` 함수로 로컬 혹은 원격 CSV 파일을 불러옵니다.

- 내 컴퓨터에 있는 파일을 불러올 때

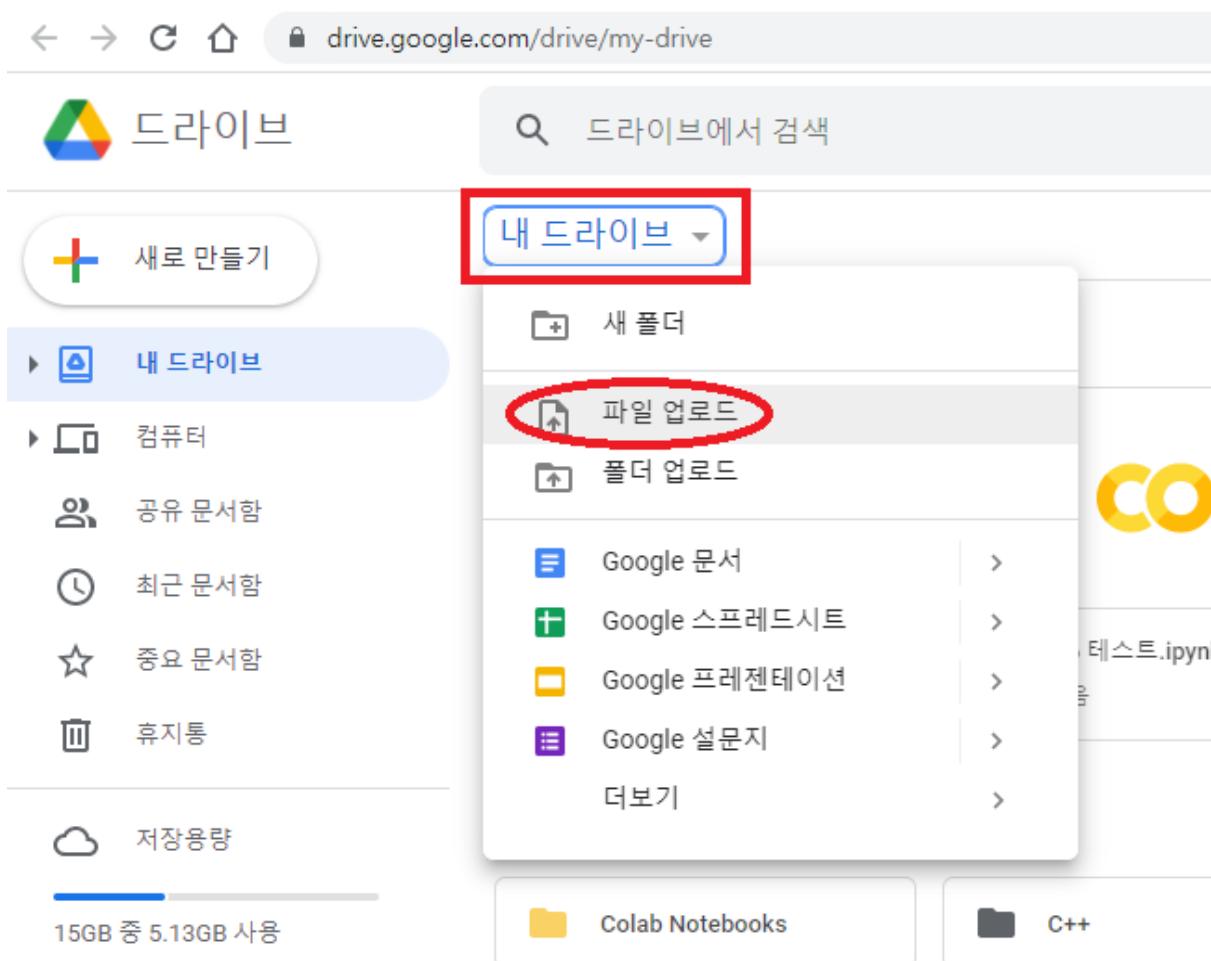
일단 구글 드라이브에 자료 파일을 업로드합니다.

구글 검색기로 가서 화면 상단 우측에 아래 그림과 같은 아이콘을 클릭하면 펼쳐지는 앱들 중에서 ‘드라이브’를 선택합니다. 또는 ‘drive.google.com’으로 들어가도 됩니다.

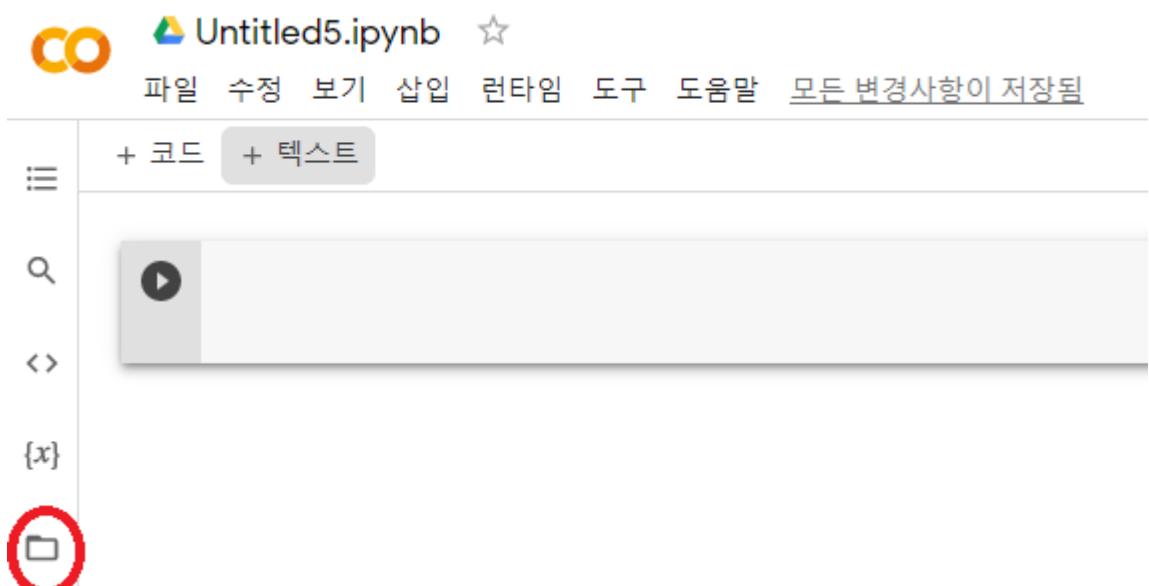


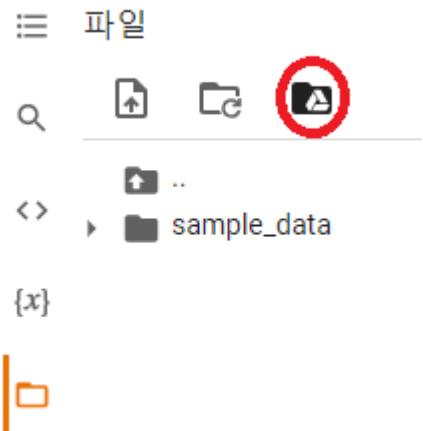
구글 드라이브 화면이 나오면 아래 그림처럼 ‘내 드라이브’를 펼쳐서 나오는 메뉴중 ‘파일 업로드’를 선택합니다.

그러면 윈도우즈 탐색기가 나오고 자신이 원하는 파일을 구글 드라이브에 업로드할 수 있습니다.



이제 구글 코랩에서 왼쪽 맨 마지막 아이콘(아래 그림에 동그라미친 부분)을 클릭합니다.





클릭한 폴더 아이콘 바로 오른쪽에 위 그림처럼 생긴 아이콘(동그라미로 가리킴)을 클릭합니다. 그러면 다음과 같은 대화창이 뜹니다. ‘Google Drive에 연결’을 클릭합니다.

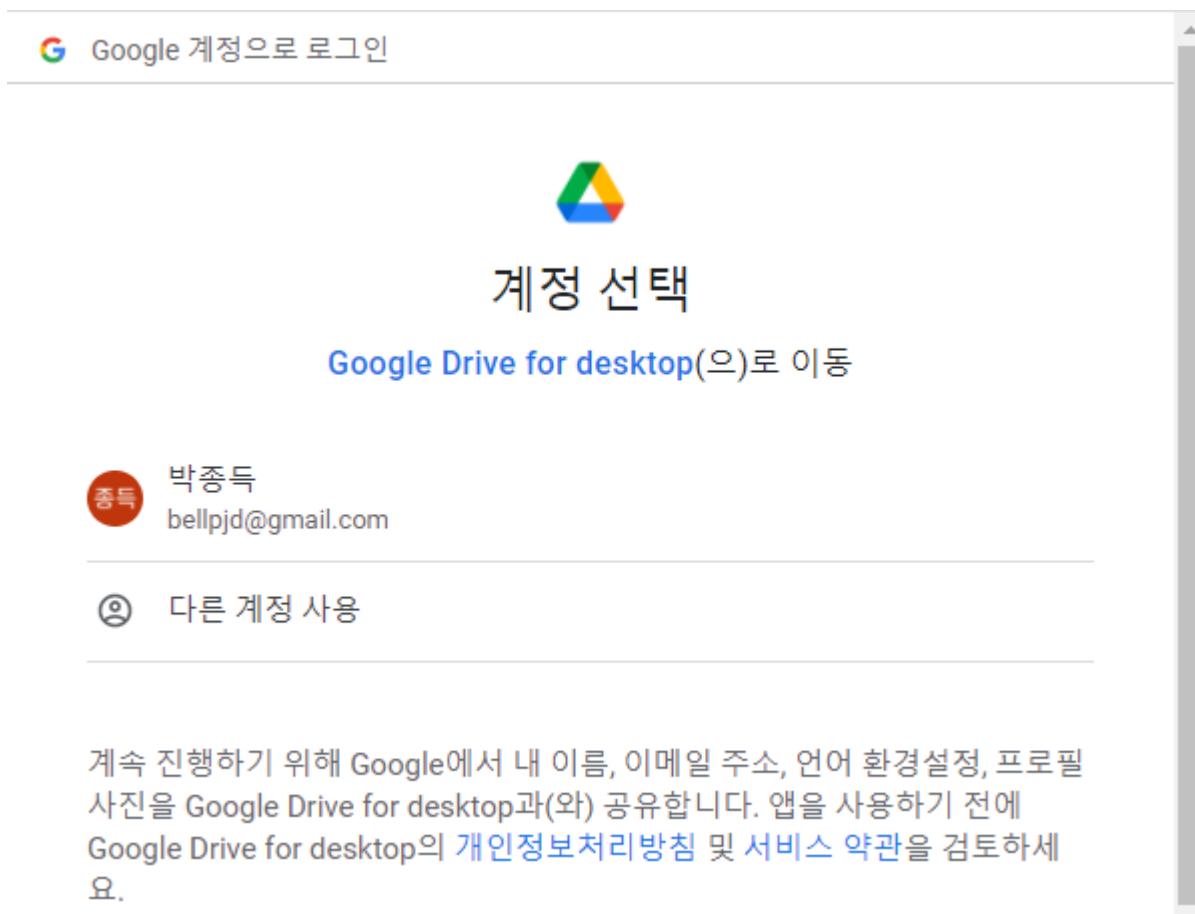
노트북이 Google Drive 파일에 액세스하도록 허용하시겠습니까?

Google Drive에 연결하면 액세스 권한이 취소될 때까지 이 노트북에서 실행된 코드가 Google Drive의 파일을 수정할 수 있습니다.

아니요

Google Drive에 연결

그러면 계정 로그인 안내화면이 아래 그림처럼 뜰 것입니다. 자신의 계정을 선택하세요.



Google 계정으로 로그인

계정 선택

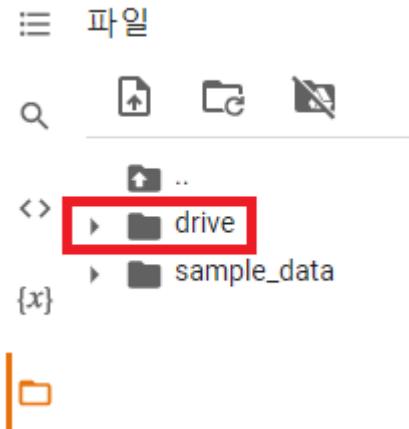
Google Drive for desktop(으)로 이동

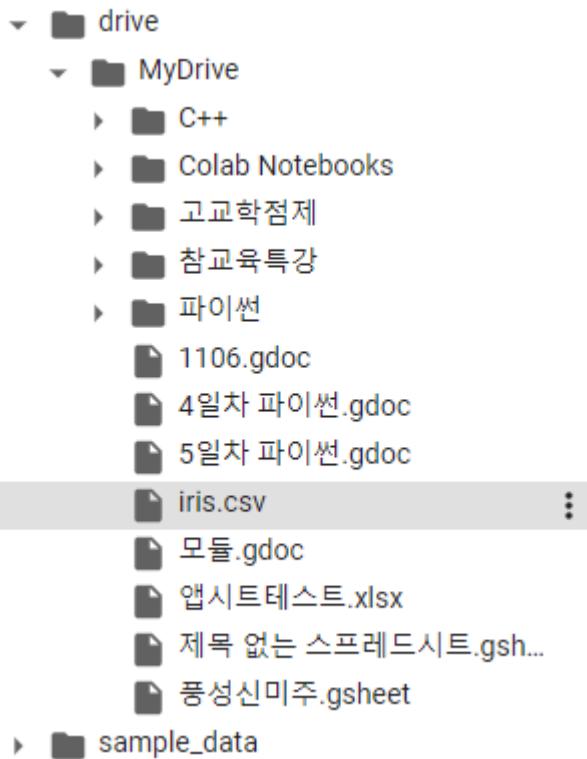
종득 박종득
bellpj@gmail.com

다른 계정 사용

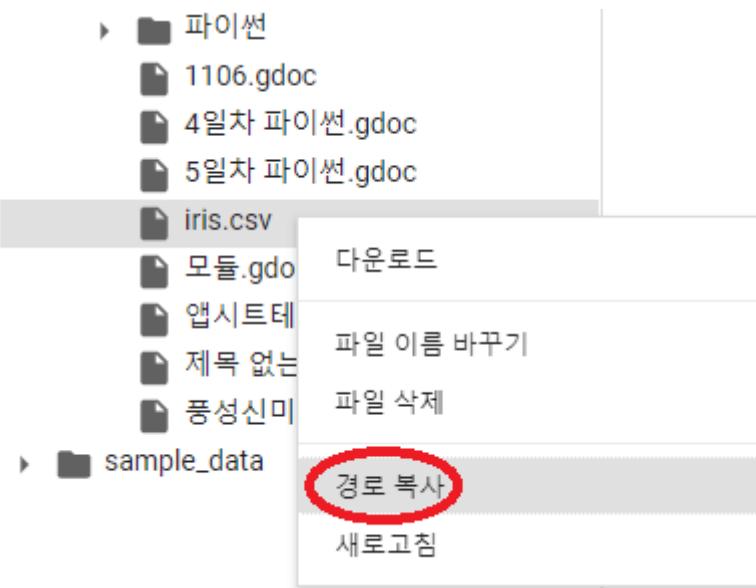
계속 진행하기 위해 Google에서 내 이름, 이메일 주소, 언어 환경설정, 프로필 사진을 Google Drive for desktop과(와) 공유합니다. 앱을 사용하기 전에 Google Drive for desktop의 [개인정보처리방침](#) 및 [서비스 약관](#)을 검토하세요.

이제 아래 그림처럼 구글 드라이브가 마운트된 모습을 볼 수 있습니다.





이제 구글 드라이브로 들어가서 원하는 데이터 파일을 선택하세요.



원하는 데이터 파일을 선택한 후 마우스 오른쪽 버튼을 누르면 나오는 팝업 화면에서 '경로 복사'를 클릭합니다.
구글 코랩 노트북의 준비해둔 셀에 붙여넣기합니다. 이 때 사전에 준비해야 할 명령어들이 있습니다. 아래 그림처럼 판다스를 임포트하고 구글 코랩에서 드라이브를 임포트합니다. 그리고 판다스의 `read_csv()` 함수를 미리 타이핑해 놓은 후 매개변수란에 복사한 경로를 붙여넣기 하면 됩니다.

```
[6] import pandas as pd
from google.colab import drive
df = pd.read_csv('/content/drive/MyDrive/iris.csv')
df.head(2)

  5.1  3.5  1.4  0.2 Iris-setosa
0   4.9   3.0   1.4   0.2   Iris-setosa
1   4.7   3.2   1.3   0.2   Iris-setosa
```

이제 위 그림처럼 읽은 아이리스 자료 파일의 첫 두 행을 인쇄해 보세요.

- 파일이 url을 가지고 있을 때

위 과정 중에서 `read_csv`의 드라이브 위치를 url 주소로 교체하면 됩니다.

3. 데이터 랭글링(Data Wrangling)

데이터 랭글링은 원본 데이터를 사용 가능한 형태로 구성하는 변환 과정입니다. 데이터 랭글링에 사용되는 가장 일반적인 구조는 데이터프레임입니다. 표형식을 생각하면 됩니다.

타이타닉 승객 데이터를 불러와 보겠습니다.

```
(8) # 타이타닉 승객 자료의 url 주소:
url = 'https://raw.githubusercontent.com/chrishalton/W
simulated_datasets/master/titanic.csv'
df = pd.read_csv(url)
# 첫 5열을 인쇄
df.head(5)
```

	Name	PClass	Age	Sex	Survived	SexCode
0	Allen, Miss Elisabeth Walton	1st	29.00	female	1	1
1	Allison, Miss Helen Loraine	1st	2.00	female	0	1
2	Allison, Mr Hudson Joshua Creighton	1st	30.00	male	0	0
3	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	1st	25.00	female	0	1
4	Allison, Master Hudson Trevor	1st	0.92	male	1	0

데이터프레임 샘플의 한 행은 하나의 샘플(여기서는 한 명의 승객)에 해당합니다. 각 열은 하나의 특성(성별, 나이 등)에 해당합니다. 예를 들면 첫 번째 승객으로 1등급 객실에 탑승한 **Elisabeth**양은 29살이고, 살아남았습니다.

각 열에 있는 자료의 타입도 다를 수 있습니다. 이름과 객실 등급, 성별은 문자열이고 나이와 생존 여부 등은 숫자로 표시되어 있습니다.

(4) 데이터 프레임 만들기

판다스로 새로운 데이터 프레임을 만드는 방법은 많이 있습니다. 그 중 한 방법을 소개합니다.

`DataFrame` 클래스를 사용하여 빈 데이터프레임을 만든 후 개별적으로 각 열을 정의합니다.

- ```
데이터프레임 구성
df = pd.DataFrame()
열 추가
df['Name'] = ['홍길동', '김갑순']
df['Age'] = [17, 17]
df['Driver'] = [False, False]
df
```

|   | Name | Age | Driver |
|---|------|-----|--------|
| 0 | 홍길동  | 17  | False  |
| 1 | 김갑순  | 17  | False  |

위 그림은 간단한 표를 만드는 과정을 보여줍니다.

- 또 다른 방법

다음 그림은 데이터프레임을 만드는 또 다른 방법입니다.

```
[17] import numpy as np
data = [['홍길동', 25, True], ['김갑순', 28, False]]
matrix = np.array(data)
df2 = pd.DataFrame(matrix, columns=
 ['Name', 'Age', 'Driver'])
df2
```

|   | Name | Age | Driver |
|---|------|-----|--------|
| 0 | 홍길동  | 25  | True   |
| 1 | 김갑순  | 28  | False  |

중간에 `matrix` 변수를 생략하고 바로 `DataFrame` 클래스에 대입할 수도 있습니다.

```
[18] df3 = pd.DataFrame(data, columns=
 ['이름', '나이', '운전면허'])
df3
```

|   | 이름  | 나이 | 운전면허  |
|---|-----|----|-------|
| 0 | 홍길동 | 25 | True  |
| 1 | 김갑순 | 28 | False |

### (5) 데이터 설명하기

데이터 적재 후 `head` 메소드로 첫 몇 행을 나타내어 보는 일로 데이터 로드를 시작합니다.

```
[19] # 타이타닉 승객 자료의 url 주소:
url = 'https://raw.githubusercontent.com/chrishalton/W
simulated_datasets/master/titanic.csv'
df = pd.read_csv(url)
첫 2열을 인쇄
df.head(2)
```

|   | Name                         | PClass | Age  | Sex    | Survived | SexCode |
|---|------------------------------|--------|------|--------|----------|---------|
| 0 | Allen, Miss Elisabeth Walton | 1st    | 29.0 | female | 1        | 1       |
| 1 | Allison, Miss Helen Loraine  | 1st    | 2.0  | female | 0        | 1       |

행과 열의 수를 확인하려면 `shape`을 사용합니다. 지난 시간에 배운 것 기억나죠?

```
[20] df.shape
```

(1313, 6)

`describe` 메소드를 사용하면 숫자열의 통계치를 얻을 수 있습니다.

다만, 아래 그림의 통계 자료에서 나이 분포는 분석할만한 자료로 쓰이겠지만, 성별같은 자료는 범주형 데이터로 생각할 이유가 없습니다. 따라서 표준편차가 큰 의미가 없을 것입니다.

[21] df.describe()

|       | Age        | Survived    | SexCode     |
|-------|------------|-------------|-------------|
| count | 756.000000 | 1313.000000 | 1313.000000 |
| mean  | 30.397989  | 0.342727    | 0.351866    |
| std   | 14.259049  | 0.474802    | 0.477734    |
| min   | 0.170000   | 0.000000    | 0.000000    |
| 25%   | 21.000000  | 0.000000    | 0.000000    |
| 50%   | 28.000000  | 0.000000    | 0.000000    |
| 75%   | 39.000000  | 1.000000    | 1.000000    |
| max   | 71.000000  | 1.000000    | 1.000000    |

#### (6) 데이터프레임 탐색

데이터프레임의 개별 데이터나 일부분을 선택하려면 loc나 iloc 메소드를 사용하여 하나 이상의 행이나 값을 선택합니다. loc는 데이터프레임의 인덱스가 레이블일 때 사용하고, iloc는 데이터프레임의 위치(색인)를 참조합니다.

[22] df.i loc[0]

```
Name Allen, Miss Elisabeth Walton
PClass 1st
Age 29
Sex female
Survived 1
SexCode 1
Name: 0, dtype: object
```

위 그림은 타니타닉 승객 자료의 첫 행에 대한 자료를 나타냅니다.

콜론으로 원하는 행의 그룹을 선택할 수 있습니다. 예를 들어 두 번째부터 네번째 행까지의 행을 선택하려면 다음과 같이 합니다.

[23] df.i loc[1:4]

|   | Name                                          | PClass | Age  | Sex    | Survived | SexCode |
|---|-----------------------------------------------|--------|------|--------|----------|---------|
| 1 | Allison, Miss Helen Loraine                   | 1st    | 2.0  | female | 0        | 1       |
| 2 | Allison, Mr Hudson Joshua Creighton           | 1st    | 30.0 | male   | 0        | 0       |
| 3 | Allison, Mrs Hudson JC (Bessie Waldo Daniels) | 1st    | 25.0 | female | 0        | 1       |

정수 색인이 아닌, 고유값을 인덱스로 사용할 경우 다음과 같이 설정합니다.

[24] df = df.set\_index(df['Name'])  
df.loc['Allen, Miss Elisabeth Walton']

```
Name Allen, Miss Elisabeth Walton
PClass 1st
Age 29
Sex female
Survived 1
SexCode 1
Name: Allen, Miss Elisabeth Walton, dtype: object
```

loc와 iloc 메소드의 슬라이싱(범위 선택)은 넘파이와는 달리 마지막 색인을 포함합니다.  
슬라이싱으로 열을 선택할 수도 있습니다.

[25] df.loc[:, 'Allison, Miss Helen Loraine', 'Age':'Sex']

|                              | Age  | Sex    |
|------------------------------|------|--------|
| Name                         |      |        |
| Allen, Miss Elisabeth Walton | 29.0 | female |
| Allison, Miss Helen Loraine  | 2.0  | female |

[27] df[['Age', 'Sex']].head(2)

|                              | Age  | Sex    |
|------------------------------|------|--------|
| Name                         |      |        |
| Allen, Miss Elisabeth Walton | 29.0 | female |
| Allison, Miss Helen Loraine  | 2.0  | female |