

レシピ29 整数とラベルの両方でデータ選択(P80)

- .iloc 、 .loc インデクサは整数とラベルの両方を同時には使用できない
- このレシピでは、カラムの整数位置をどのようにして見つけるかを示す
- 番号とラベル名の取り方

取得対象	行	列
番号	df.index.get_loc('行番')	df.columns.get_loc('列名')
ラベル名	df.index[行番]	df.columns[列番]

(1) 大学データセットを読み込み、校名（INSTNM）をインデックスにする

In [3]:

```
import pandas as pd
college = pd.read_csv('college.csv', index_col='INSTNM')
college.head(2)
```

Out[3]:

	CITY	STABBR	HBCU	MENONLY	WOMENONLY	RELAFFIL	SATVRMID	SATMTMID	DISTANCEONL
INSTNM									
Alabama A & M University	Normal	AL	1.0	0.0	0.0	0	424.0	420.0	0.
University of Alabama at Birmingham	Birmingham	AL	0.0	0.0	0.0	0	570.0	565.0	0.

2 rows × 26 columns

(2) Indexメソッド get_loc を使ってカラムの整数位置を見つける

df.get_loc(列名)：列番号を取得

In [8]:

```
# カラムからラベル名「UGDS_WHITE」の列番を求める
col_start = college.columns.get_loc('UGDS_WHITE')
col_end = college.columns.get_loc('UGDS_UNKN') + 1 # スライシングすると末尾の要素が含まれないため+1
col_start, col_end
```

Out[8]: (10, 19)

(3) (2) を元に、.ilocで列番を使ってDF取得

- 5行目まで、列はすべてを取得

In [7]:

```
college.iloc[:5, col_start:col_end]
```

Out[7]:

	UGDS_WHITE	UGDS_BLACK	UGDS_HISP	UGDS_ASIAN	UGDS_AIAN	UGDS_NHPI	UGDS_2MOR	UGDS
INSTNM								

	UGDS_WHITE	UGDS_BLACK	UGDS_HISP	UGDS_ASIAN	UGDS_AIAN	UGDS_NHPI	UGDS_2MOR	UGDS
INSTNM								
Alabama A & M University	0.0333	0.9353	0.0055	0.0019	0.0024	0.0019	0.0000	0
University of Alabama at Birmingham	0.5922	0.2600	0.0283	0.0518	0.0022	0.0007	0.0368	(
Amridge University	0.2990	0.4192	0.0069	0.0034	0.0000	0.0000	0.0000	0
University of Alabama in Huntsville	0.6988	0.1255	0.0382	0.0376	0.0143	0.0002	0.0172	0
Alabama State University	0.0158	0.9208	0.0121	0.0019	0.0010	0.0006	0.0098	0

(補足)

- 逆に行名を取得し、df.loc[行名、列名]の形でもOK

df.index[行番]：行名を取得

In [12]:

```
row_start = college.index[0]
row_end = college.index[4]
college.loc[row_start:row_end, 'UGDS_WHITE': 'UGDS_UNKN']
```

Out[12]:

	UGDS_WHITE	UGDS_BLACK	UGDS_HISP	UGDS_ASIAN	UGDS_AIAN	UGDS_NHPI	UGDS_2MOR	UGDS
INSTNM								
Alabama A & M University	0.0333	0.9353	0.0055	0.0019	0.0024	0.0019	0.0000	0
University of Alabama at Birmingham	0.5922	0.2600	0.0283	0.0518	0.0022	0.0007	0.0368	(
Amridge University	0.2990	0.4192	0.0069	0.0034	0.0000	0.0000	0.0000	0
University of Alabama in Huntsville	0.6988	0.1255	0.0382	0.0376	0.0143	0.0002	0.0172	0
Alabama State University	0.0158	0.9208	0.0121	0.0019	0.0010	0.0006	0.0098	0

レシピ30 スカラー選択の高速化（P81）

- スカラーインデクサ `.iat`、`.at` でより高速にスカラー要素を取得できる
- `iloc`や`loc`よりどれだけ高速化を比較する

(1) 校名をインデックスにしてcollegeデータセットを読み込む。 `.loc` に校名とカラム名を渡してスカラー値を取得する

```
In [14]: college = pd.read_csv('college.csv', index_col='INSTNM')
cn = 'Texas A & M University-College Station'
college.loc[cn, 'UGDS_WHITE']
```

Out[14]: 0.6609999999999999

(2) 同じ結果を `.at` で取得する

```
In [15]: college.at[cn, 'UGDS_WHITE']
```

Out[15]: 0.6609999999999999

(3) 両者の速度を比較する (`.loc` vs `.at`)

```
In [18]: %timeit college.loc[cn, 'UGDS_WHITE']
```

5.98 μ s $\pm$ 97.2 ns per loop (mean $\pm$ std. dev. of 7 runs, 100000 loops each)

```
In [19]: %timeit college.at[cn, 'UGDS_WHITE']
```

3.29 μ s $\pm$ 47.8 ns per loop (mean $\pm$ std. dev. of 7 runs, 100000 loops each)

- 結果、 `.at` が約45%高速

(4) 今度は `.iolo` と `.iat` の速度を比較する

```
In [28]: row_num = college.index.get_loc(cn)
col_num = college.columns.get_loc('UGDS_WHITE')
row_num, col_num
```

Out[28]: (3765, 10)

```
In [29]: %timeit college.iloc[row_num, col_num]
```

19.5 μ s $\pm$ 346 ns per loop (mean $\pm$ std. dev. of 7 runs, 10000 loops each)

```
In [30]: %timeit college.iat[row_num, col_num]
```

17.5 μ s $\pm$ 790 ns per loop (mean $\pm$ std. dev. of 7 runs, 10000 loops each)

- 結果、 `.iat` が約10%高速
- スカラーインデクサが速い。大して速くないと思うかもしれないがプログラムでスカラー選択が繰り返される場合、大きな違いになる

(補足) `.iat` 、 `.at` はSeriesにも使える。スカラー値を渡せば、スカラーが返される

```
In [32]: state = college['STABBR']
state.iat[1000]
```

Out[32]: 'IL'

```
In [33]: state.at['Stanford University']
```

```
Out[33]: 'CA'
```

レシピ31 行の手軽なスライシング（P83）

割愛

レシピ32 文字順にスライシング（P85）

- .loc インデクサは通常インデックスの文字列ラベルでデータを選ぶ
- その場合、インデックスがソートしておく必要がある

(1) 読み込み、校名をインデックスにセット

```
In [40]: college = pd.read_csv('college.csv', index_col='INSTNM')
college.head(2)
```

	CITY	STABBR	HBCU	MENONLY	WOMENONLY	RELAFFIL	SATVRMID	SATMTMID	DISTANCEONL
INSTNM									
Alabama A & M University	Normal	AL	1.0	0.0	0.0	0	424.0	420.0	0.
University of Alabama at Birmingham	Birmingham	AL	0.0	0.0	0.0	0	570.0	565.0	0.

2 rows × 26 columns

(2) 校名が'Sp'と'Su'の間の全大学を選ぶ（エラーになる）

```
In [37]: # college.loc['Sp':'Su']
```

KeyError: 'Sp'

```
In [43]: # cf
college.index.is_monotonic
```

```
Out[43]: False
```

(3) (2) はインデックスがソートされていないのでエラーとなった。ソートする

```
In [44]: college = college.sort_index()
```

```
In [46]: # cf
college.index.is_monotonic
```

Out[46]: True

(4) (2) と同じコマンドを実行する

```
In [39]: college.loc['Sp':'Su']
```

Out[39]:

	CITY	STABBR	HBCU	MENONLY	WOMENONLY	RELAFFIL	SATVRMID	SATMTMID	DISTANCEONLY
INSTNM									
Spa Tech Institute-Ipswich	Ipswich	MA	0.0	0.0	0.0	0	NaN	NaN	0.0
Spa Tech Institute-Plymouth	Plymouth	MA	0.0	0.0	0.0	0	NaN	NaN	0.0
Spa Tech Institute-Westboro	Westboro	MA	0.0	0.0	0.0	0	NaN	NaN	0.0
Spa Tech Institute-Westbrook	Westbrook	ME	0.0	0.0	0.0	0	NaN	NaN	0.0
Spalding University	Louisville	KY	0.0	0.0	0.0	1	490.0	440.0	0.0
...	...	...	...	...	...	...	...	...	...
Studio Academy of Beauty	Chandler	AZ	0.0	0.0	0.0	0	NaN	NaN	0.0
Studio Jewelers	New York	NY	0.0	0.0	0.0	0	NaN	NaN	0.0
Stylemaster College of Hair Design	Longview	WA	0.0	0.0	0.0	0	NaN	NaN	0.0
Styles and Profiles Beauty College	Selmer	TN	0.0	0.0	0.0	0	NaN	NaN	0.0
Styletrends Barber and Hairstyling Academy	Rock Hill	SC	0.0	0.0	0.0	0	NaN	NaN	0.0

201 rows × 26 columns



(補足)

- このレシピでアルファベットの2文字で大学を選ぶのは簡単
- 文字DからSで始まる大学をすべて選ぶなら `college.loc['D':'T']` となる
- このようなスライシングは**末尾を含む**ので、Tという大学も選ばれる
- ちなみにPythonのソートでは、大文字が小文字より先に、数字が大文字より先になる

```
In [ ]:
```