+ Code + Text

Pandas Selection and Indexing

alt text

```
import pandas as pd
import random

# read the data from the downloaded CSV file.
data = pd.read_csv('https://s3-eu-west-1.amazonaws.com/shanebucket/downloads/uk-500.csv
```

```
data.head(5)
```

	first_name	last_name	company_name	address	city	county	postal	phone1	phone2	
0	Aleshia	Tomkiewicz	Alan D Rosenburg Cpa Pc	14 Taylor St	St. Stephens Ward	Kent	CT2 7PP	01835-703597	01944-369967	atomkiewicz@hotr
1	Evan	Zigomalas	Cap Gemini America	5 Binney St	Abbey Ward	Buckinghamshire	HP11 2AX	01937-864715	01714-737668	evan.zigomalas@gr
2	France	Andrade	Elliott, John W Esq	8 Moor Place	East Southbourne and Tuckton W	Bournemouth	BH6 3BE	01347-368222	01935-821636	france.andrade@hotr
3	Ulysses	Mcwalters	Mcmahan,	505 Exeter	Hawerby	Lincolnshire	DN36	01912-	01302-	ulysses@hotr

1. Selecting pandas data using “iloc”

```
# Single selections using iloc and DataFrame
# Rows:
data.iloc[0] # first row of data frame (Aleshia Tomkiewicz) - Note a Series data type o
```

```
first_name                                   Aleshia
last_name                                 Tomkiewicz
company_name                 Alan D Rosenburg Cpa Pc
address                                 14 Taylor St
city                               St. Stephens Ward
county                                          Kent
postal                                       CT2 7PP
phone1                                  01835-703597
phone2                                  01944-369967
email                        atomkiewicz@hotmail.com
web             http://www.alandrosenburgcpapc.co.uk
Name: 0, dtype: object
```

```
data.iloc[1] # second row of data frame (Evan Zigomalas)
```

```
data.iloc[-1] # last row of data frame (Mi Richan)
```

```
# Columns:
data.iloc[:,0] # first column of data frame (first_name)
```

```
data.iloc[:,1] # second column of data frame (last_name)
data.iloc[:,-1] # last column of data frame (id)
```

```
# Multiple row and column selections using iloc and DataFrame
data.iloc[0:5] # first five rows of dataframe
data.iloc[:, 0:2] # first two columns of data frame with all rows
```

```
data.iloc[[0,3,6,24], [0,5,6]] # 1st, 4th, 7th, 25th row + 1st 6th 7th columns.
data.iloc[0:5, 5:8] # first 5 rows and 5th, 6th, 7th columns of data frame (county -> p
```

2. Selecting pandas data using “loc”

The Pandas loc indexer can be used with DataFrames for two different use cases:

a.) Selecting rows by label/index

b.) Selecting rows with a boolean / conditional lookup

The loc indexer is used with the same syntax as iloc: data.loc[,] .

2a. Label-based / Index-based indexing using .loc

```
data.set_index("last_name", inplace=True)
data.head()
data.iloc[0]
```

```
# Select rows with index values 'Andrade' and 'Veness', with all columns between 'city'
data.loc[['Andrade', 'Veness'], 'city':'email']
```

```
# Select same rows, with just 'first_name', 'address' and 'city' columns
data.loc['Andrade':'Veness', ['first_name', 'address', 'city']]
```

2b. Boolean / Logical indexing using .loc

```
# Select rows with first name Antonio, # and all columns between 'city' and 'email'
data.loc[data['first_name'] == 'Antonio', 'city':'email']

data.loc[((data['county']=='Kent') & (data['first_name']=='Aleshia')), 'city':'email']
```

```
# Select rows where the email column ends with 'hotmail.com', include all columns
data.loc[data['email'].str.endswith("hotmail.com")]

#startswith, contains, find

len()
```

```
# Select rows with last_name equal to some values, all columns
data.loc[data['first_name'].isin(['France', 'Tyisha', 'Eric'])]
# equivalent
#data.loc[(data['first_name']=='France') |(data['first_name']=='Tyisha')|(data['first_n
#data.loc[(data['first_name']== 'France') |(data['first_name']== 'Tyisha')|(data['first_n
```

```
# Select rows with first name Antonio AND hotmail email addresses
data.loc[data['email'].str.endswith("gmail.com") & (data['first_name'] == 'Antonio')]
```

```
# # A lambda function that yields True/False values can also be used.
# Select rows where the company name has 4 words in it.
data.loc[data['company_name'].apply(lambda x: len(x.split(' ')) == 4)]
data.head()
data.reset_index(inplace=True)
```

```
data.head()
```

```
data[(data['county']=='Kent') & (data['first_name']=='Aleshia')]
```

Iloc for changing the data

Double-click (or enter) to edit

```
data.loc[(data['county']=='Kent'), ['first_name']] = 'Dhanya'
```

```
data.loc[(data['county']=='Kent') & (data['first_name']=='Dhanya')]
data.loc[(data.first_name== 'Dhanya'), ['company_name', 'city']] = ['Amrita','Coimbator
```

```
data.loc[data.first_name=='Dhanya']
```

```
# similarly or, and other logical operators
```

```
# Slicing through list of values
#print df.iloc[[1, 3, 5], [1, 3]]
#print df.iloc[1:3, :]
#print df.iloc[:,1:3]
```

```
df=pd.read_csv('https://raw.githubusercontent.com/fivethirtyeight/data/master/cabinet-t
```

```
df.head()
```

```
df.tail()
```

```
len(df.loc[df['position'].str.startswith('Secretary')])
```

```
df['end'].value_counts
```

```
df['president'].unique()
```

```
df['president'].nunique()
```

```
df.dtypes
```

```
df.loc[(df['position']=='OMB Director')& (df['president']=='Clinton')]['appointee']
```

https://github.com/fivethirtyeight/data/blob/master/cabinet-turnover/cabinet-turnover.csv

https://raw.githubusercontent.com/pplonski/datasets-for-start/master/credit/data.csv# QUIZ

link text

https://forms.office.com/Pages/ResponsePage.aspx?id=o835AF4H5USqC6ujrdZTn63NXi4eP7xHpYuejR_M4qdUNk9QUFlQOThQUUdKM080RFhTR1YzTlYwUC4u