Programming Principles in Python (CSCI 503/490)

Data & Visualization

Dr. David Koop

Quiz

Quiz

1. Which of the following is not a Python library used for manipulating data?
   (a) numpy
   (b) penguins
   (c) pandas
   (d) polars

Quiz

2. Which of the following is not a difference between numpy arrays and python lists?
   (a) Arrays are mutable; lists are not
   (b) Array slices are views over the original array; list slices are not views
   (c) Arrays require that all elements have the same type; lists do not
   (d) Arrays are faster to access than lists

Quiz

3. Given the array `arr = np.array([[1,2],[3,4],[5,6]])`, what is `arr[1,:].shape`?
   (a) `(2,)`
   (b) `(1,2)`
   (c) `(3,1)`
   (d) `(3,2)`

Quiz

4. Evaluate `pd.Series([1,2,3],['a','b','c']) + pd.Series([1,2,3],['c','b','a'])`.
   (a) `pd.Series([2,4,6],['a','b','c'])`
   (b) `pd.Series([1,2,3,1,2,3],['a','b','c','c','b','a'])`
   (c) `pd.Series([4,4,4],['a','b','c'])`
   (d) There is an error.

Quiz

5. Given a dataframe df with a float column `pop`, what does `df['pop'] > 2` do?
   (a) Returns a boolean series with entries indicating if the `pop` column is greater than 2
   (b) Returns a new dataframe with rows where the `pop` column's value is greater than 2
   (c) Returns the `pop` series with only the entries where the value is greater than 2
   (d) Mutates the dataframe `df` so that it only includes rows where the `pop` column's value is greater than 2

pandas

- Contains high-level data structures and manipulation tools designed to make data analysis fast and easy in Python
- Originally built on top of NumPy
- Built with the following requirements:
  - Data structures with labeled axes (aligning data)
  - Support time series data
  - Do arithmetic operations that include metadata (labels)
  - Handle missing data
  - Add merge and relational operations

polars

- Contains high-level data structures and manipulation tools designed to make data analysis **"lightning"** fast and easy in Python
  - Built using Apache Arrow
  - Written from scratch using Rust but with a Python API
  - Parallelized (uses multiple cores)
  - Intuitive API

Series

- A one-dimensional data structure (with a type)
  - `s = pl.Series([1,2,3])`
  - `t = pd.Series([1,2,3])`
- May also have a name and dtype
  - `s = pl.Series('name',['a','b','c'],dtype=pl.Float)`
  - `t = pd.Series([1,2,3], name='num',dtype='float')`
- In pandas, a series has an index
  - `ti = pd.Series([1,2,3],['a','b','c']) # index ['a','b','c']`
  - `ti = pd.Series({'a': 1, 'b': 2, 'c': 3}) # same index`
- Indexing: `s[0], t[0], ti['a'], ti.iloc[0], ti.loc['a']`

Series Operations

- Like numpy: elementwise / broadcasting
  - `Series([1,2,3]) + Series([1,2,3]) # Series([2,4,6])`
  - `Series([1,2,3]) + 4 # Series([5,6,7])`
- …but for pandas, with custom indexes, the operations **align** on the index:
  - ```
    pd.Series([1,2,3],index=list('abc') +
    pd.Series([1,2,3],index=list('cba')
                        # pd.Series([4,4,4], index=['a','b','c'])
    ```
  - also have `.add`, `.subtract`, … with `fill_value` argument

[W. McKinney, Python for Data Analysis]
    ```

DataFrame

- A collection of Series (uniquely named)
  - Similar to a table in a database
  - Similar to a sheet in a spreadsheet
- `df = DataFrame({'state': ['Ohio', 'Ohio', 'Ohio', 'Nevada'],`
  `'year': [2000, 2001, 2002, 2001],`
  `'pop': [1.5, 1.7, 3.6, 2.4]})`
- In pandas:
  - Has an index shared with each series
  - Index is automatically assigned just as with a series but can be passed in as well via `index` kwarg

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

pandas DataFrame

Column Names

```
df = pd.read_csv('penguins_lter.csv')
```

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

Column Names

Index

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

Column Names

Index

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

Column: `df['Island']`

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

Column Names

Row: `df.loc[2]`

Index

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

Column: `df['Island']`

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

Column Names

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

Row: `df.loc[2]`

Index

Cell: `df.loc[341,'Species']`

Column: `df['Island']`

pandas DataFrame

```
df = pd.read_csv('penguins_lter.csv')
```

Column Names

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

Row: `df.loc[2]`

Index

Cell: `df.loc[341,'Species']`

Missing Data

Column: `df['Island']`

polars DataFrame

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	36.7
…	…	…	…	…	…	…	…	…	…
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	121	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A1"	"Yes"	"11/22/09"	46.8
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

polars DataFrame

Column Names & Types

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	36.7
…	…	…	…	…	…	…	…	…	…
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	121	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A1"	"Yes"	"11/22/09"	46.8
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

polars DataFrame

Column Names & Types

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	36.7
…	…	…	…	…	…	…	…	…	…
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	121	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A1"	"Yes"	"11/22/09"	46.8
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

Column: `df['Island']`

polars DataFrame

Column Names & Types

Row: `df[2]`

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade…	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	36.7
…	…	…	…	…	…	…	…	…	…
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	121	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A1"	"Yes"	"11/22/09"	46.8
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap…	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

Column: `df['Island']`

polars DataFrame

Column Names & Types

Row: `df[2]`

Cell: `df['Species'][341]`

Column: `df['Island']`

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	36.7
...	...	...	...	...	...	...	...	...	...
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

polars DataFrame

Column Names & Types

Row: `df[2]`

Cell: `df['Species'][341]`

Missing Data

Column: `df['Island']`

shape: (344, 10)

studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
str	i64	str	str	str	str	str	str	str	f64
"PAL0708"	1	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A1"	"Yes"	"11/11/07"	39.1
"PAL0708"	2	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N1A2"	"Yes"	"11/11/07"	39.5
"PAL0708"	3	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A1"	"Yes"	"11/16/07"	40.3
"PAL0708"	4	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N2A2"	"Yes"	"11/16/07"	null
"PAL0708"	5	"Adelie Penguin (Pygoscelis ade...	"Anvers"	"Torgersen"	"Adult, 1 Egg Stage"	"N3A1"	"Yes"	"11/16/07"	
...	...	...	...	...	...	...	...	...	...
"PAL0910"	120	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N38A2"	"No"	"12/1/09"	null
"PAL0910"	121	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A1"	"Yes"	"11/22/09"	46.8
"PAL0910"	122	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N39A2"	"Yes"	"11/22/09"	50.4
"PAL0910"	123	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A1"	"Yes"	"11/22/09"	45.2
"PAL0910"	124	"Gentoo penguin (Pygoscelis pap...	"Anvers"	"Biscoe"	"Adult, 1 Egg Stage"	"N43A2"	"Yes"	"11/22/09"	49.9

Filtering

- polars: `df.filter(pl.col('Culmen Length (mm)') > 40)`
- pandas: `dfa[dfa['Culmen Length (mm)'] > 40]`

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Date Egg	Culmen Length (mm)
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/07	39.1
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/07	39.5
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/07	40.3
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/07	NaN
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/07	36.7
...	...	...	...	...	...	...	...	...	...	...
339	PAL0910	120	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N38A2	No	12/1/09	NaN
340	PAL0910	121	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A1	Yes	11/22/09	46.8
341	PAL0910	122	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N39A2	Yes	11/22/09	50.4
342	PAL0910	123	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A1	Yes	11/22/09	45.2
343	PAL0910	124	Gentoo penguin (Pygoscelis papua)	Anvers	Biscoe	Adult, 1 Egg Stage	N43A2	Yes	11/22/09	49.9

344 rows × 17 columns

Other Operations

- Sorting
  - `df.sort('pop')`
  - `dfa.sort_values('pop')`
- Unique Values / Drop Duplicates
  - `df['state'].unique(), df.unique()`
  - `dfa['state'].unique(), dfa.drop_duplicates()`
- Value Counts
  - `df['state'].value_counts()`
- Statistics
  - `df['pop'].min(), df['pop'].mean(), df['pop'].describe()`
  - `dfa['pop'].min(), dfa['pop'].mean(), dfa['pop'].describe()`

Reading and Writing Data

- CSV:
  - Pandas: `pd.read_csv`, `dfa.to_csv`
  - Polars: `pl.read_csv`, `df.write_csv`
  - Many different options available (delimiter, skiprows, index, header, …)
- Other formats:
  - JSON: `*_json`
  - Parquet: `*_parquet`
  - Excel: `*_excel`

Assignment 7

- Illinois Unemployment Data
- Downloading and uncompressing files
- Finding files using OS libraries
- Use a match statement to process data
- Store per-county dataframes, each in a csv file

Missing Data

- polars: shows `null`
- pandas: shows `NaN` (or `NA` or `None` depending on dtype)
- Checking if missing:
  - polars: `pl.col('pop').is_null(), .is_not_null()`
  - pandas: `dfa['pop'].isnull(), .notnull()`
- Drop missing data:
  - polars: `pl.col('pop').drop_nulls()`, pandas: `dfa['pop'].dropna()`
- Filling in missing data:
  - polars: `pl.col('pop').fill_null()`,(`forward, backward, max,…`)
  - pandas: `dfa['pop'].fillna()`, now `ffill(), bfill()`

Derived Data

- Create new columns from existing columns
- pandas
  - `dfa["CulmenRatio"] = dfa['CLength'] / dfa['CDepth'] # Mut!`
  - `dfa = dfa.assign(CulmenRatio= dfa['CLength'] / dfa['CDepth'])`
- polars
  - ```
    df.with_columns(
            (df['CLength'] / df['CDepth']).alias('CulmenRatio'))
    ```
- Note that operations are computed in a vectorized manner
- Similarities to functional paradigm (map/filter):
  - specify the operation once, on entire column/frame
  - no loops
    ```

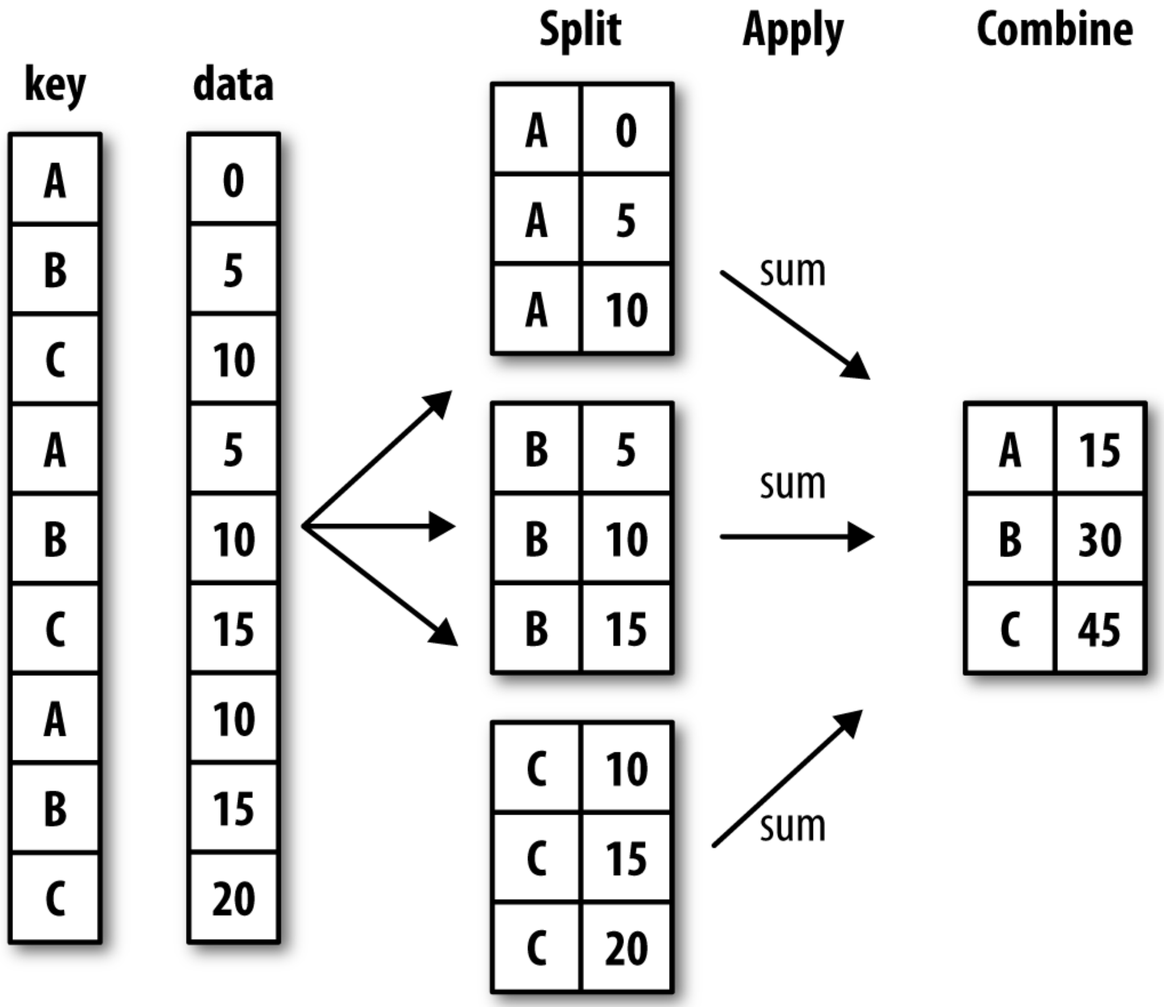
pandas inplace

- Generally, when we modify a data frame, we reassign:
  - `rdf = dfa.reset_index()`
  - This is usually very **efficient**
  - Allows for method chaining
- There are versions where you can do this "inplace" (**try to avoid this**)
  - `dfa.reset_index(inplace=True)`
  - This means **no reassignment**, but it isn't usually any faster nor better
  - Sometimes still creates a copy
  - Will likely be deprecated

Aggregation

- Descriptive statistics
  - `df['Culmen Length (mm)'].mean()`
  - `.median()`
  - `.describe()`
  - `.count()`
  - `.min(), .max()`
- Also general methods
  - `.sum()`
  - `.product()`

Split-Apply-Combine



[W. McKinney, Python for Data Analysis]

Split-Apply-Combine

- Similar to Map (split+apply) Reduce (combine) paradigm
- The Pattern:
  1. **Split** the data by some grouping variable
  2. **Apply** some function to each group independently
  3. **Combine** the data into some output dataset
- The apply step is usually one of:
  - Aggregate
  - Transform
  - Filter

[T. Brandt]

Group By

- Polars: `group_by`, Pandas: `groupby`
- `group_by` method creates a `GroupBy` object
- `group_by` **does not compute** anything until there is an aggregate step
- Sizes of groups:
  - `df.group_by('Island').agg(pl.len()) # DataFrame`
  - `dfa.groupby('Island').size() # Series`
- Can iterate through the groups (names and dataframes):
  -
```
for name, gdf in df.group_by('Island'):
    display(name, gdf)
```

Aggregation

- Single Column:
  - `df.group_by('Island').agg(pl.col('Length (mm)').mean())`
  - `dfa.groupby('Island')['Length (mm)'].mean()`
- pandas returns a Series, polars returns a DataFrame
- List of Values:
  - `df.group_by('Island').agg(pl.col('Length (mm)'))`
  - `dfa.groupby('Island')['Length (mm)'].apply(list)`

Aggregation (Multiple Columns)

- Multiple columns in an aggregation
  - `df.group_by('Island').agg(pl.col('Length','Depth').mean())`
  - `dfa.groupby('Island')[['Length','Depth']].mean()`
- Multiple aggregations for a column
  - ```
    df.group_by('Island').agg(pl.col('Length').min().alias('LMin'),
                              pl.col('Length').max().alias('LMax'))
    ```
  - `dfa.groupby('Island').agg({'Length': ['min','max']})`
  - ```
    dfa.groupby('Island').agg(LMin=('Length','min')
                              LMax=('Length','max'))
    ```

Different Data Layouts

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

Initial Data

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Transpose

name	trt	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Tidy Data

[H. Wickham, 2014]