

Introducing DataFrames

What's the point of pandas?



Course outline

- Chapter 1: DataFrames
 - Sorting and subsetting
 - Creating new columns

Course outline

- **Chapter 1: DataFrames**
 - Sorting and subsetting
 - Creating new columns
- **Chapter 2: Aggregating Data**
 - Summary statistics
 - Counting
 - Grouped summary statistics

Course outline

- **Chapter 1: DataFrames**
 - Sorting and subsetting
 - Creating new columns
- **Chapter 2: Aggregating Data**
 - Summary statistics
 - Counting
 - Grouped summary statistics
- **Chapter 3: Slicing and Indexing Data**
 - Subsetting using slicing
 - Indexes and subsetting using indexes

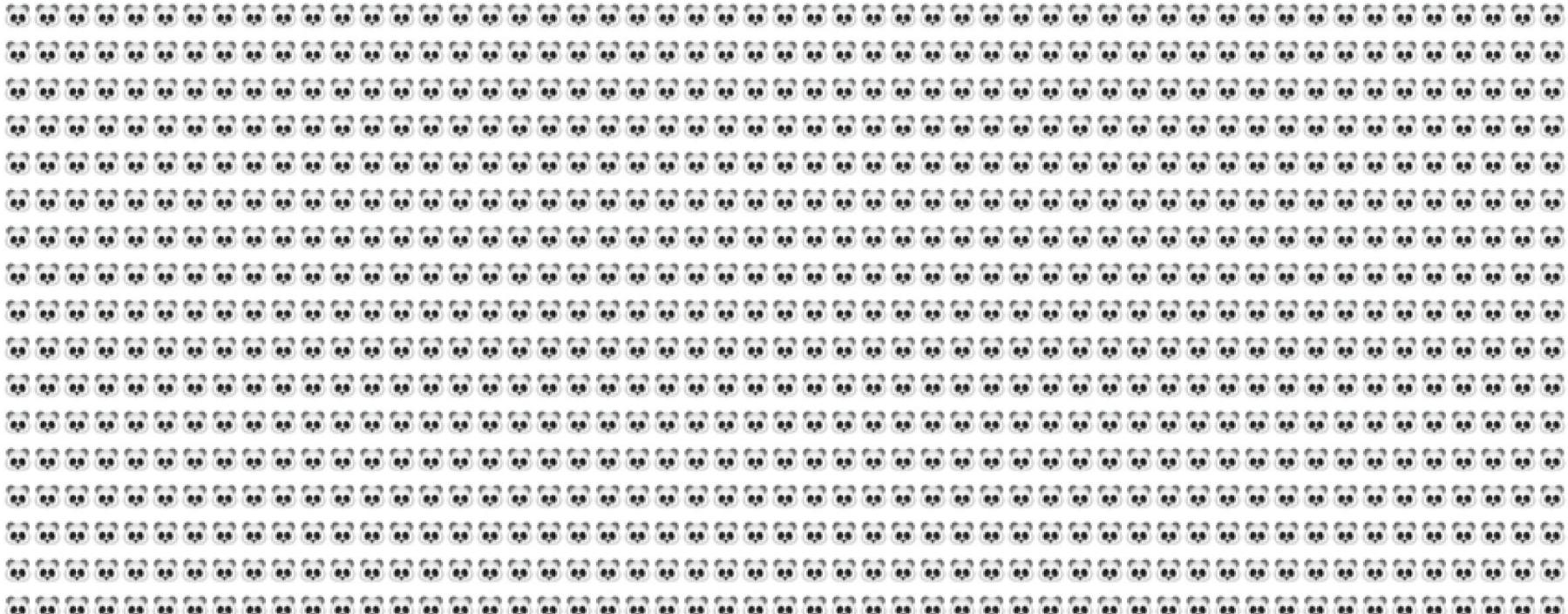
Course outline

- **Chapter 1: DataFrames**
 - Sorting and subsetting
 - Creating new columns
- **Chapter 2: Aggregating Data**
 - Summary statistics
 - Counting
 - Grouped summary statistics
- **Chapter 3: Slicing and Indexing Data**
 - Subsetting using slicing
 - Indexes and subsetting using indexes
- **Chapter 4: Creating and Visualizing Data**
 - Plotting
 - Handling missing data
 - Reading data into a DataFrame

pandas is built on NumPy and Matplotlib



pandas is popular



¹ <https://pypistats.org/packages/pandas>

Rectangular data

Name	Breed	Color	Height (cm)	Weight (kg)	Date of Birth
Bella	Labrador	Brown	56	25	2013-07-01
Charlie	Poodle	Black	43	23	2016-09-16
Lucy	Chow Chow	Brown	46	22	2014-08-25
Cooper	Schnauzer	Gray	49	17	2011-12-11
Max	Labrador	Black	59	29	2017-01-20
Stella	Chihuahua	Tan	18	2	2015-04-20
Bernie	St. Bernard	White	77	74	2018-02-27

pandas DataFrames

```
print(dogs)
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01
1	Charlie	Poodle	Black	43	24	2016-09-16
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
3	Cooper	Schnauzer	Gray	49	17	2011-12-11
4	Max	Labrador	Black	59	29	2017-01-20
5	Stella	Chihuahua	Tan	18	2	2015-04-20
6	Bernie	St. Bernard	White	77	74	2018-02-27

Exploring a DataFrame: .head()

```
dogs.head()
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01
1	Charlie	Poodle	Black	43	24	2016-09-16
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
3	Cooper	Schnauzer	Gray	49	17	2011-12-11
4	Max	Labrador	Black	59	29	2017-01-20

Exploring a DataFrame: .info()

```
dogs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7 entries, 0 to 6
Data columns (total 6 columns):
name            7 non-null object
breed           7 non-null object
color           7 non-null object
height_cm       7 non-null int64
weight_kg       7 non-null int64
date_of_birth   7 non-null object
dtypes: int64(2), object(4)
```

Exploring a DataFrame: .shape

```
dogs.shape
```

```
(7, 6)
```

Exploring a DataFrame: .describe()

```
dogs.describe()
```

	height_cm	weight_kg
count	7.000000	7.000000
mean	49.714286	27.428571
std	17.960274	22.292429
min	18.000000	2.000000
25%	44.500000	19.500000
50%	49.000000	23.000000
75%	57.500000	27.000000
max	77.000000	74.000000

Components of a DataFrame: .values

```
dogs.values
```

```
array([['Bella', 'Labrador', 'Brown', 56, 24, '2013-07-01'],
       ['Charlie', 'Poodle', 'Black', 43, 24, '2016-09-16'],
       ['Lucy', 'Chow Chow', 'Brown', 46, 24, '2014-08-25'],
       ['Cooper', 'Schnauzer', 'Gray', 49, 17, '2011-12-11'],
       ['Max', 'Labrador', 'Black', 59, 29, '2017-01-20'],
       ['Stella', 'Chihuahua', 'Tan', 18, 2, '2015-04-20'],
       ['Bernie', 'St. Bernard', 'White', 77, 74, '2018-02-27']],
      dtype=object)
```

Components of a DataFrame: .columns and .index

```
dogs.columns
```

```
Index(['name', 'breed', 'color', 'height_cm', 'weight_kg', 'date_of_birth'],  
      dtype='object')
```

Components of a DataFrame: .columns and .index

```
dogs.columns
```

```
Index(['name', 'breed', 'color', 'height_cm', 'weight_kg', 'date_of_birth'],  
      dtype='object')
```

```
dogs.index
```

```
RangeIndex(start=0, stop=7, step=1)
```

pandas Philosophy

| There should be one -- and preferably only one -- obvious way to do it.

- *The Zen of Python* by Tim Peters, Item 13

¹ <https://www.python.org/dev/peps/pep-0020/>

pandas Philosophy

| There should be one -- and preferably only one -- obvious way to do it.

- *The Zen of Python* by Tim Peters, Item 13



¹ <https://www.python.org/dev/peps/pep-0020/>

Let's practice!

Sorting and subsetting

Sorting

```
dogs.sort_values("weight_kg")
```

	name	breed	color	height_cm	weight_kg	date_of_birth
5	Stella	Chihuahua	Tan	18	2	2015-04-20
3	Cooper	Schnauzer	Gray	49	17	2011-12-11
0	Bella	Labrador	Brown	56	24	2013-07-01
1	Charlie	Poodle	Black	43	24	2016-09-16
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
4	Max	Labrador	Black	59	29	2017-01-20
6	Bernie	St. Bernard	White	77	74	2018-02-27

Sorting in descending order

```
dogs.sort_values("weight_kg", ascending=False)
```

	name	breed	color	height_cm	weight_kg	date_of_birth
6	Bernie	St. Bernard	White	77	74	2018-02-27
4	Max	Labrador	Black	59	29	2017-01-20
0	Bella	Labrador	Brown	56	24	2013-07-01
1	Charlie	Poodle	Black	43	24	2016-09-16
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
3	Cooper	Schnauzer	Gray	49	17	2011-12-11
5	Stella	Chihuahua	Tan	18	2	2015-04-20

Sorting by multiple variables

```
dogs.sort_values(["weight_kg", "height_cm"])
```

		name	breed	color	height_cm	weight_kg	date_of_birth
5	Stella		Chihuahua	Tan	18	2	2015-04-20
3	Cooper		Schnauzer	Gray	49	17	2011-12-11
1	Charlie		Poodle	Black	43	24	2016-09-16
2	Lucy		Chow Chow	Brown	46	24	2014-08-25
0	Bella		Labrador	Brown	56	24	2013-07-01
4	Max		Labrador	Black	59	29	2017-01-20
6	Bernie	St. Bernard		White	77	74	2018-02-27

Sorting by multiple variables

```
dogs.sort_values(["weight_kg", "height_cm"], ascending=[True, False])
```

	name	breed	color	height_cm	weight_kg	date_of_birth
5	Stella	Chihuahua	Tan	18	2	2015-04-20
3	Cooper	Schnauzer	Gray	49	17	2011-12-11
0	Bella	Labrador	Brown	56	24	2013-07-01
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
1	Charlie	Poodle	Black	43	24	2016-09-16
4	Max	Labrador	Black	59	29	2017-01-20
6	Bernie	St. Bernard	White	77	74	2018-02-27

Subsetting columns

```
dogs[ "name" ]
```

```
0      Bella
1    Charlie
2      Lucy
3    Cooper
4      Max
5    Stella
6    Bernie
Name: name, dtype: object
```

Subsetting multiple columns

```
dogs[["breed", "height_cm"]]
```

	breed	height_cm
0	Labrador	56
1	Poodle	43
2	Chow Chow	46
3	Schnauzer	49
4	Labrador	59
5	Chihuahua	18
6	St. Bernard	77

Subsetting multiple columns

```
dogs[["breed", "height_cm"]]
```

	breed	height_cm
0	Labrador	56
1	Poodle	43
2	Chow Chow	46
3	Schnauzer	49
4	Labrador	59
5	Chihuahua	18
6	St. Bernard	77

```
cols_to_subset = ["breed", "height_cm"]  
dogs[cols_to_subset]
```

	breed	height_cm
0	Labrador	56
1	Poodle	43
2	Chow Chow	46
3	Schnauzer	49
4	Labrador	59
5	Chihuahua	18
6	St. Bernard	77

Subsetting rows

```
dogs[ "height_cm" ] > 50
```

```
0    True
1   False
2   False
3   False
4    True
5   False
6    True
Name: height_cm, dtype: bool
```

Subsetting rows

```
dogs[dogs["height_cm"] > 50]
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01
4	Max	Labrador	Black	59	29	2017-01-20
6	Bernie	St. Bernard	White	77	74	2018-02-27

Subsetting based on text data

```
dogs[dogs["breed"] == "Labrador"]
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01
4	Max	Labrador	Black	59	29	2017-01-20

Subsetting based on dates

```
dogs[dogs["date_of_birth"] > "2015-01-01"]
```

	name	breed	color	height_cm	weight_kg	date_of_birth
1	Charlie	Poodle	Black	43	24	2016-09-16
4	Max	Labrador	Black	59	29	2017-01-20
5	Stella	Chihuahua	Tan	18	2	2015-04-20
6	Bernie	St. Bernard	White	77	74	2018-02-27

Subsetting based on multiple conditions

```
is_lab = dogs["breed"] == "Labrador"  
is_brown = dogs["color"] == "Brown"  
dogs[is_lab & is_brown]
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01

Subsetting based on multiple conditions

```
is_lab = dogs["breed"] == "Labrador"  
is_brown = dogs["color"] == "Brown"  
dogs[is_lab & is_brown]
```

```
      name      breed  color  height_cm  weight_kg date_of_birth  
0    Bella    Labrador   Brown        56         24    2013-07-01
```

```
dogs[ (dogs["breed"] == "Labrador") & (dogs["color"] == "Brown") ]
```

Subsetting using .isin()

```
is_black_or_brown = dogs["color"].isin(["Black", "Brown"])
dogs[is_black_or_brown]
```

	name	breed	color	height_cm	weight_kg	date_of_birth
0	Bella	Labrador	Brown	56	24	2013-07-01
1	Charlie	Poodle	Black	43	24	2016-09-16
2	Lucy	Chow Chow	Brown	46	24	2014-08-25
4	Max	Labrador	Black	59	29	2017-01-20

Let's practice!

New columns

Adding a new column

```
dogs["height_m"] = dogs["height_cm"] / 100  
print(dogs)
```

	name	breed	color	height_cm	weight_kg	date_of_birth	height_m
0	Bella	Labrador	Brown	56	24	2013-07-01	0.56
1	Charlie	Poodle	Black	43	24	2016-09-16	0.43
2	Lucy	Chow Chow	Brown	46	24	2014-08-25	0.46
3	Cooper	Schnauzer	Gray	49	17	2011-12-11	0.49
4	Max	Labrador	Black	59	29	2017-01-20	0.59
5	Stella	Chihuahua	Tan	18	2	2015-04-20	0.18
6	Bernie	St. Bernard	White	77	74	2018-02-27	0.77

Doggy mass index

BMI = weight in kg / (height in m)²

Doggy mass index

$$\text{BMI} = \text{weight in kg} / (\text{height in m})^2$$

```
dogs["bmi"] = dogs["weight_kg"] / dogs["height_m"] ** 2  
print(dogs.head())
```

	name	breed	color	height_cm	weight_kg	date_of_birth	height_m	bmi
0	Bella	Labrador	Brown	56	24	2013-07-01	0.56	76.530612
1	Charlie	Poodle	Black	43	24	2016-09-16	0.43	129.799892
2	Lucy	Chow Chow	Brown	46	24	2014-08-25	0.46	113.421550
3	Cooper	Schnauzer	Gray	49	17	2011-12-11	0.49	70.803832
4	Max	Labrador	Black	59	29	2017-01-20	0.59	83.309394

Multiple manipulations

Multiple manipulations

```
bmi_lt_100 = dogs[dogs["bmi"] < 100]
```

Multiple manipulations

```
bmi_lt_100 = dogs[dogs["bmi"] < 100]  
bmi_lt_100_height = bmi_lt_100.sort_values("height_cm", ascending=False)
```

Multiple manipulations

```
bmi_lt_100 = dogs[dogs["bmi"] < 100]  
bmi_lt_100_height = bmi_lt_100.sort_values("height_cm", ascending=False)  
bmi_lt_100_height[["name", "height_cm", "bmi"]]
```

	name	height_cm	bmi
4	Max	59	83.309394
0	Bella	56	76.530612
3	Cooper	49	70.803832
5	Stella	18	61.728395

Let's practice!