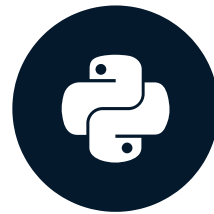


# Introducing the dataset

INTRODUCTION TO PYTHON FOR FINANCE



Adina Howe  
Instructor

# Overall Review

- Python shell and scripts
- Variables and data types
- Lists
- Arrays
- Methods and functions
- Indexing and subse ing
- Matplotlib

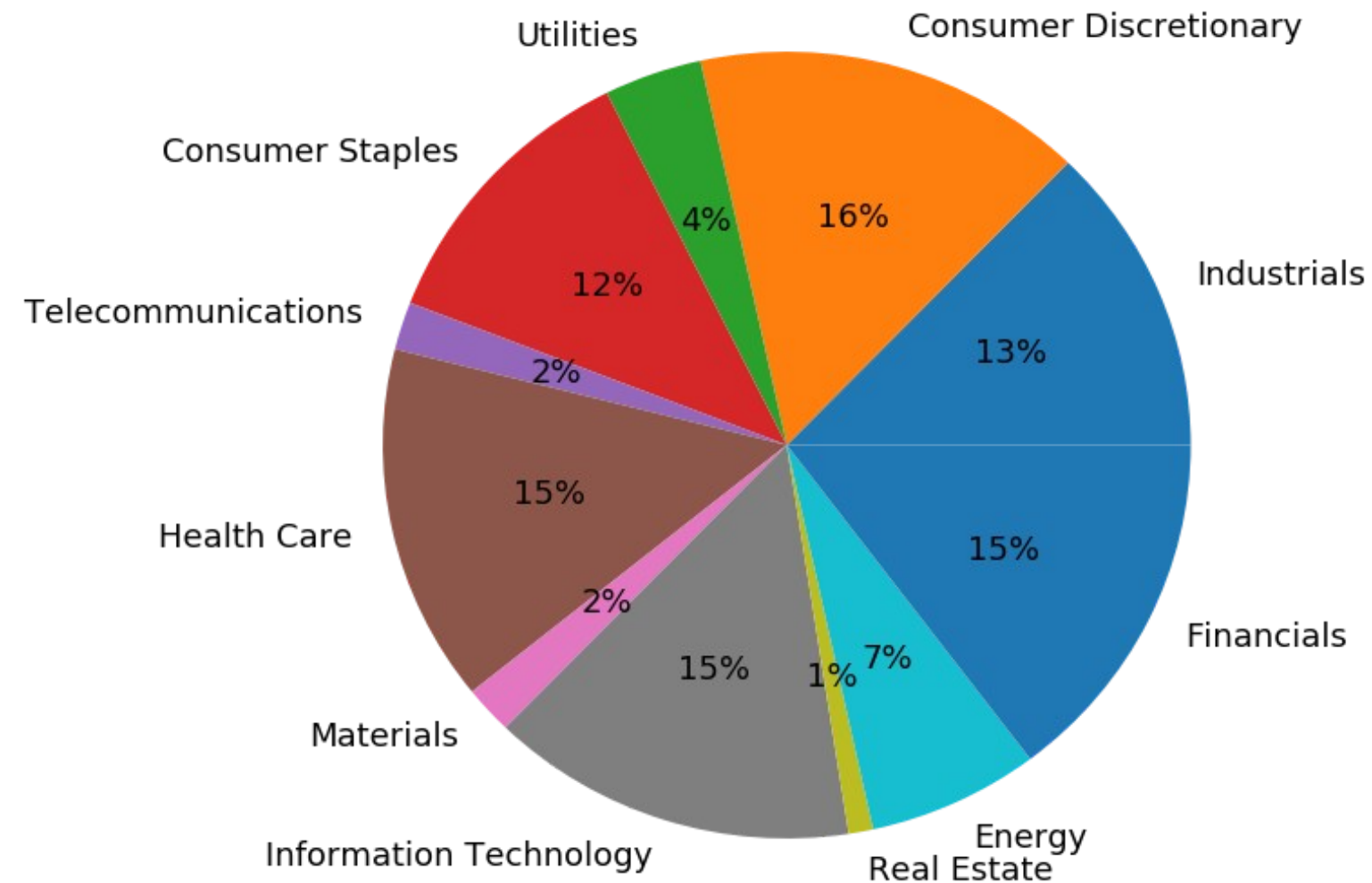
# S&P 100 Companies

Standard and Poor's S&P 100:

- made up of major companies that span multiple industry groups
- used to measure stock performance of large companies

# S&P 100 Case Study

Sectors of Companies within the S&P 100 in 2017



# The data

<u>Name</u>	<u>Sector</u>	<u>Price</u>	<u>EPS</u>
Apple Inc	Information Technology	170.12	9.2
Abbvie Inc	Health Care	93.29	5.31
Abbott Laboratories	Health Care	55.28	2.41
Accenture Plc	Information Technology	145.3	5.91
Allergan Plc	Health Care	171.81	15.42
American International Group	Financials	59.5	2.51
Allstate Corp	Financials	100.5	6.79
Amgen	Health Care	168.93	12.58
...	...	...	...
Visa Inc	Information Technology	110.27	3.48
Verizon Communications Inc	Telecommunications	45.85	3.75
Walgreens Boots Alliance	Consumer Staples	70.25	5.1
Wells Fargo & Company	Financials	54.02	4.14
Wal-Mart Stores	Consumer Staples	96.08	4.36
Exxon Mobil Corp	Energy	80.31	3.56

# Price to Earnings Ratio

$$\text{Price to earning ratio} = \frac{\text{Market price}}{\text{Earnings per share}}$$

- The ratio for valuing a company that measures its current share price relative to its per-share earnings
- In general, higher P/E ratio indicates higher growth expectations

# Your mission

## GIVEN

Lists of data describing the S&P 100: names, prices, earnings, sectors

## OBJECTIVE PART I

Explore and analyze the S&P 100 data, specifically the P/E ratios of S&P 100 companies

# Step 1: examine the lists

```
In [1]: my_list = [1, 2, 3, 4, 5]
```

```
# first element
```

```
In [2]: print(my_list[0])
```

```
1
```

```
# last element
```

```
In [3]: print(my_list[-1])
```

```
5
```

```
# range of elements
```

```
In [4]: print(my_list[0:3])
```

```
[1, 2, 3]
```



## Step 2: Convert lists to arrays

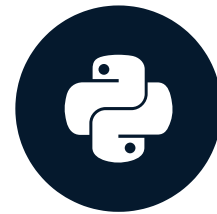
```
# Convert lists to arrays
import numpy as np
my_array = np.array(my_list)
```

# Step 3: Elementwise array operations

```
# Elementwise array operations  
array_ratio = array1 / array2
```

# A closer look at the sectors

INTRODUCTION TO PYTHON FOR FINANCE



Adina Howe  
Instructor

# Your mission

## GIVEN

Numpy arrays of data describing the S&P 100: names, prices, earnings, sectors

## OBJECTIVE PART II

Explore and analyze sector-specific P/E ratios within companies of the S&P 100

# Step 1: Create a boolean filtering array

```
stock_prices = np.array([100, 200, 300])  
filter_array = (stock_prices >= 150)  
print(filter_array)
```

```
[False True  True]
```

## Step 2: Apply filtering array to subset another array

```
stock_prices = np.array([100, 200, 300])  
filter_array = (stock_prices >= 150)  
print(stock_prices[filter_array])
```

```
[200 300]
```

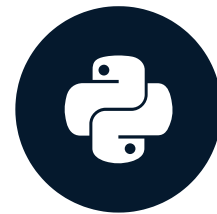
# Step 3: Summarize P/E ratios

Calculate the average and standard deviation of these sector-specific P/E ratios

```
import numpy as np
average_value = np.mean(my_array)
std_value = np.std(my_array)
```

# Visualizing trends

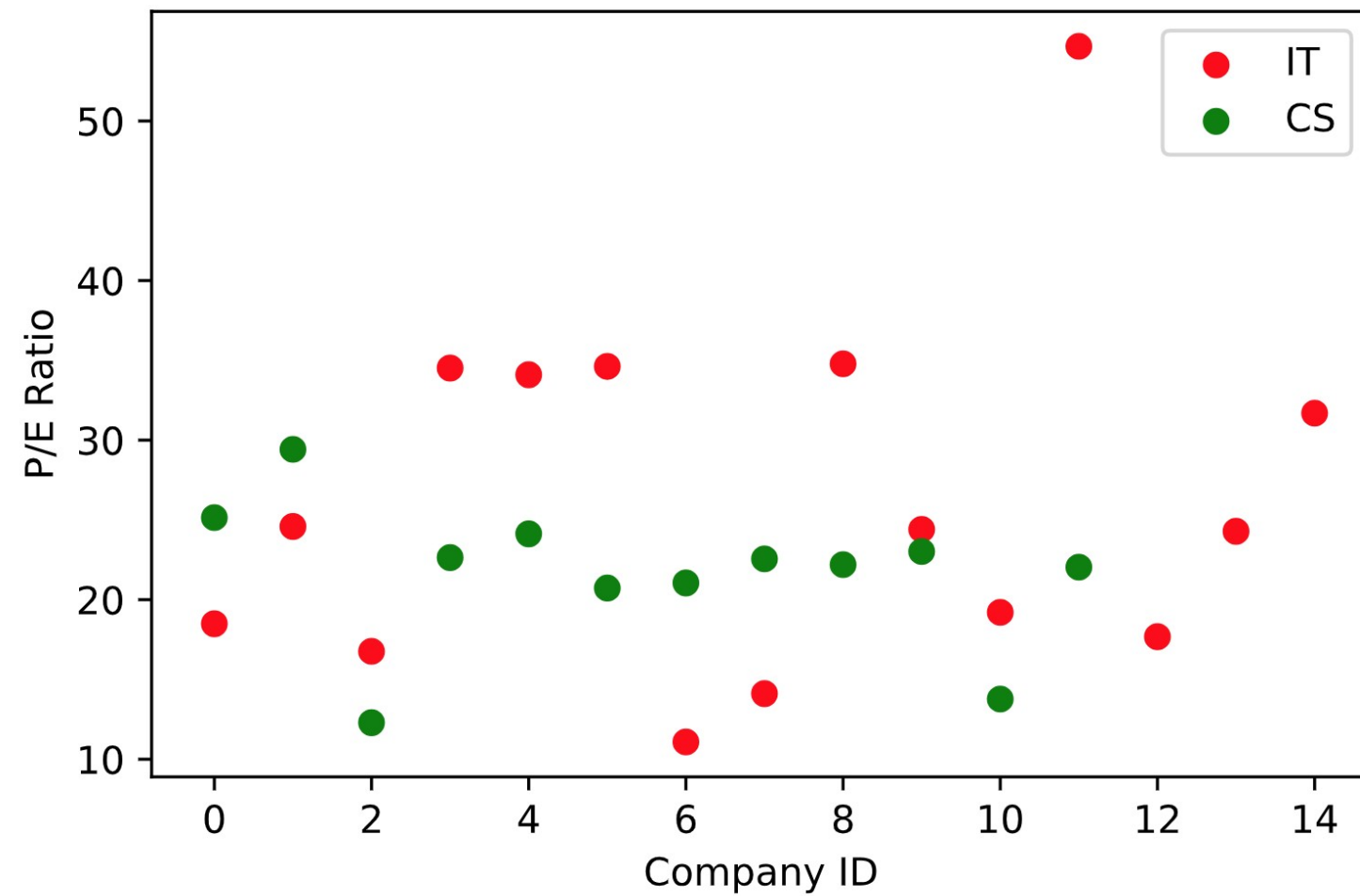
INTRODUCTION TO PYTHON FOR FINANCE



Adina Howe  
Instructor



# Your mission - outlier?



# Step 1: Make a histogram

```
import matplotlib.pyplot as plt
plt.hist(hist_data, bins = 8)
plt.show()
```

# Step 2: Identify the Outlier

- Identify the outlier P/E ratio
- Create a boolean array      Iter to subset this company
- Filter out this company information from the provided datasets