Introduction to Pandas: Takeaways

Syntax

PANDAS DATAFRAME BASICS

- Read a file into a DataFrame:

```
f500 = pd.read_csv('f500.csv', index_col=0)
```

- Return a DataFrame's data types:

```
col_types = f500.dtypes
```

- Return the dimensions of a DataFrame:

```
dims = f500.shape
```

SELECTING VALUES FROM A DATAFRAME

- Selecting a single column:

```
f500["rank"]
```

- Selecting multiple columns:

```
f500[["country", "rank"]]
```

- Selecting the first n rows:

```
first_five = f500.head(5)
```

- Selecting rows and columns from a DataFrame by label:

```
big_movers = f500.loc[["Aviva", "HP", "JD.com", "BHP Billiton"], ["rank", "previous_rank"]]
bottom_companies = f500.loc["National Grid":"AutoNation", ["rank", "sector", "country"]]
revenue_giants = f500.loc[["Apple", "Industrial & Commercial Bank of China", "China
Construction Bank", "Agricultural Bank of China"], "revenues":"profit_change"]
```

Concepts

- NumPy provides fundamental structures and tools that make working with data easier, but there are several things that limit its usefulness as a single tool when working with data:
    - The lack of support for column names forces us to frame questions as multi-dimensional array operations.
    - Support for only one data type per ndarray makes it difficult to work with data that contains numeric and string data.
    - There many low level methods — however, there are many common analysis patterns that don't have pre-built methods.
- The **pandas** library provides solutions to all of these pain points and more.
- Pandas is not a replacement for NumPy, but an *extension* of NumPy.
- The underlying code for pandas uses the NumPy library extensively.

- The main objects in pandas are **Series** and **DataFrames**. Series is equivalent to a 1D ndarray while a DataFrame is equivalent to a 2D ndarray.
- Different label selection methods:

Select by Label	Explicit Syntax	Common Shorthand
Single column from DataFrame	`df.loc[:, "col1"]`	`df["col1"]`
List of columns from DataFrame	`df.loc[:, ["col1", "col7"]]`	`df[["col1", "col7"]]`
Slice of columns from DataFrame	`df.loc[:, "col1":"col4"]`	None
Single row from DataFrame	`df.loc["row1"]`	None
List of rows from DataFrame	`df.loc[["row1", "row5"]]`	None
Slice of rows from DataFrame	`df.loc["row1":"row5"]`	`df["row1":"row5"]`
Single item from Series	`s.loc["item8"]`	`s["item8"]`
List of items from Series	`s.loc[["item1", "item7"]]`	`s[["item1", "item7"]]`
Slice of items from Series	`s.loc["item2":"item4"]`	`s["item2":"item4"]`

Resources

- Dataframe.loc[]
- Indexing and Selecting Data