Rotman

INTRO TO PYTHON

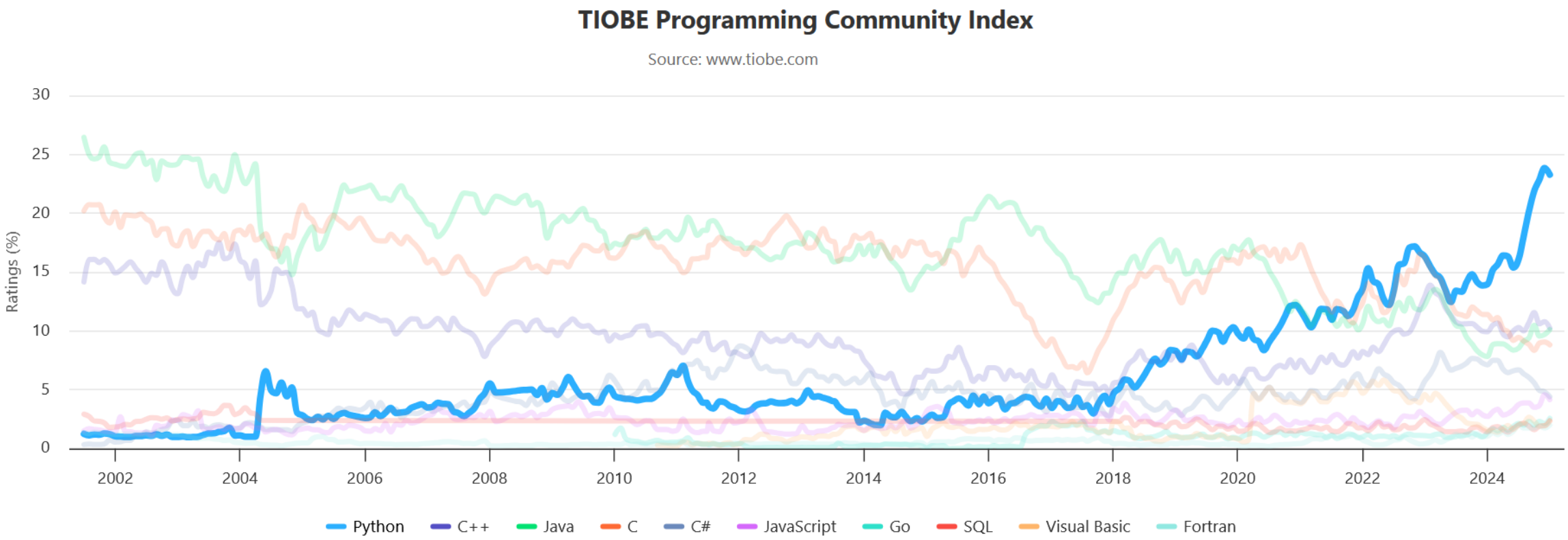
Programming and Data Analysis Basics

August 27, 2025 Prepared by Jay Cao / MDAL (Built on Niti's past slides)

Website: https://rmdal.github.io/python-intro-2025-fall/

Rotman School of Management
UNIVERSITY OF TORONTO

Popularity of Python – Number 1



Ref. https://www.tiobe.com/tiobe-index/

What Can Python Do – Just About Anything

- Statistical analysis
- Scientific computing
- Machine learning / AI
- Data visualization

- Others
  - Scripting & automation
  - Web development
  - Systems testing & prototyping
  - Desktop applications

Install and Code Python

- Notebook in the Cloud (**best choice** for beginners)
  - Google Colab (**Our Choice Today**)
  - Uoft JupyterHub (2G memory limit)

- Notebook on your laptop
  - Anaconda Python distribution
    - Installation comes with Python, Jupyter Notebook, and many data science packages
    - Convenient for beginners (with a GUI launchpad)
    - But installation comes with a very large footprint (4.4G)

- Python Official Distribution + Development Environment (e.g. VS Code)
  - Install additional packages/libraries on your own
  - Work with both pure Python code and Python notebook

Plan for Today and Next Session

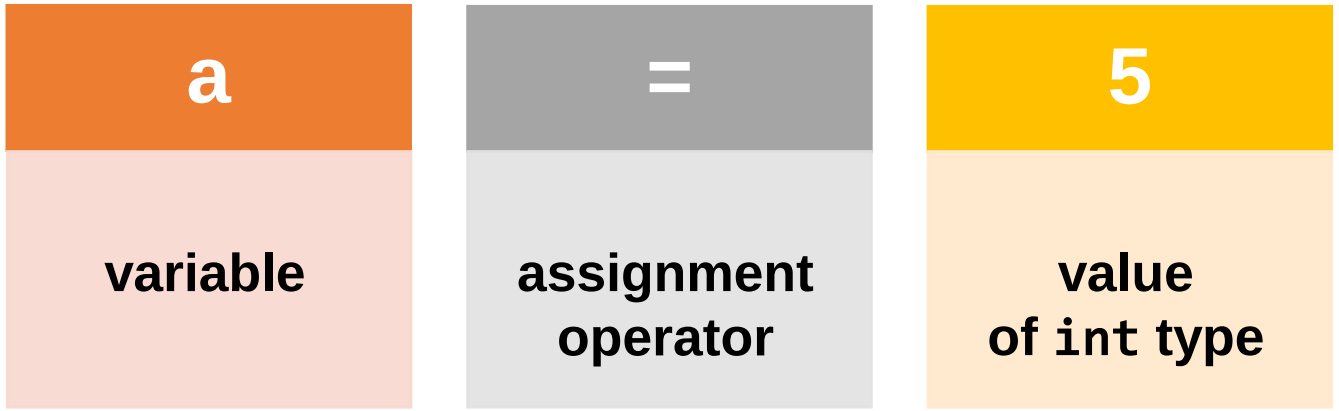
- Programming basics (companion notebook on workshop site)
  - Data structures
  - Programming structures
  - Functions

- Data analysis basics (companion notebook on workshop site)
  - Simple data processing (operations on 2D tables/dataframes)
  - Simple analysis on a stock price time series
  - Predict next-day stock price with a linear regression model

Basic Data Structure - 1

- What's data structure
  - a way of storing and organizing data/values of certain types

- Value, type, variable, and the assignment operator (=)



An assignment statement

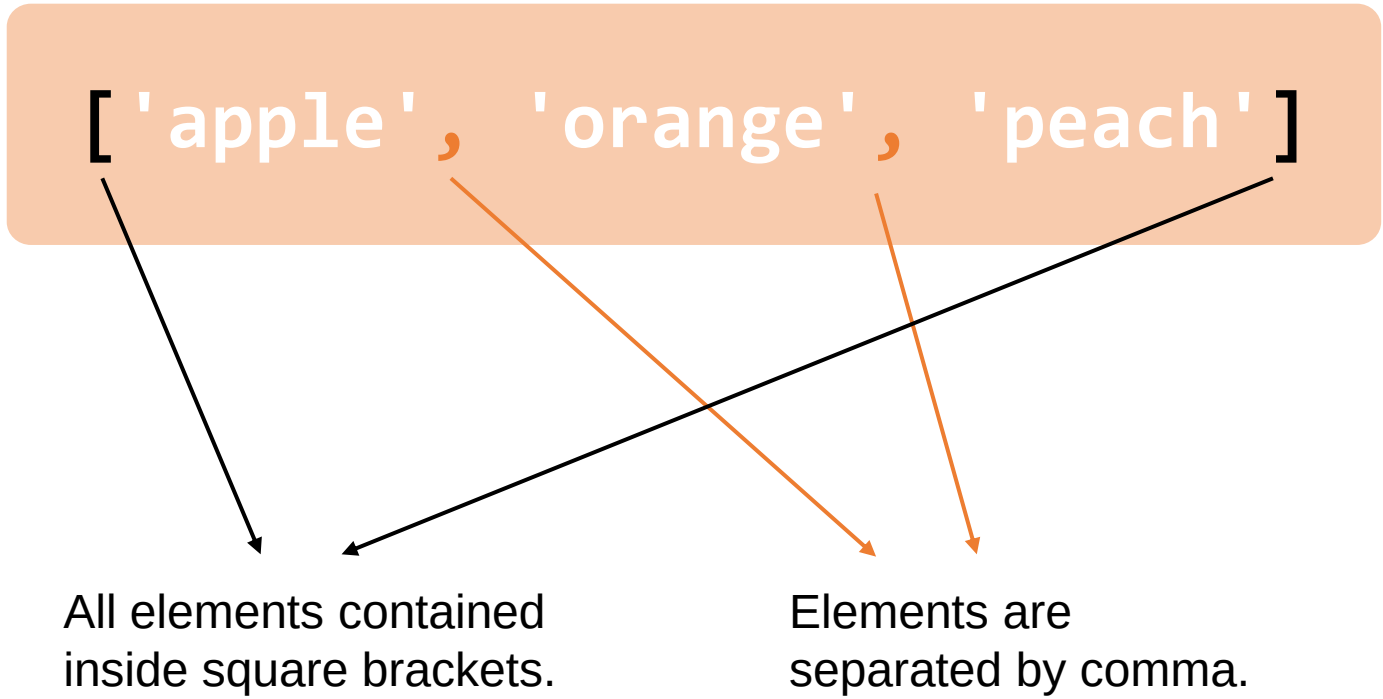
Basic Data Structure - 2

- Basic numeric types: `int`, `float`, `complex`

- String type (`str`)
  - String index
  - Methods associated with string object

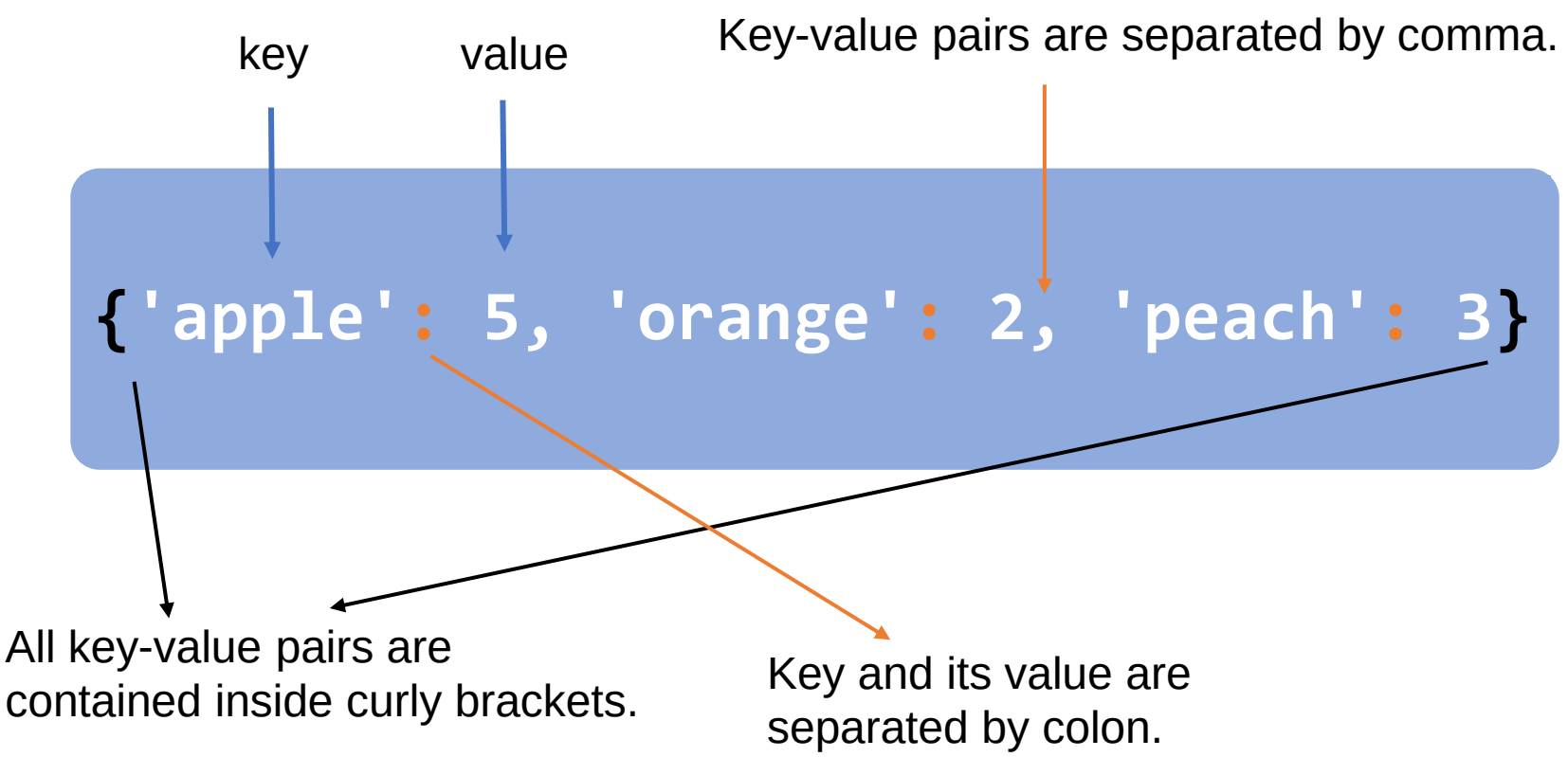
More DS Native to Python - List

- Mutable
- Ordered
- indexed

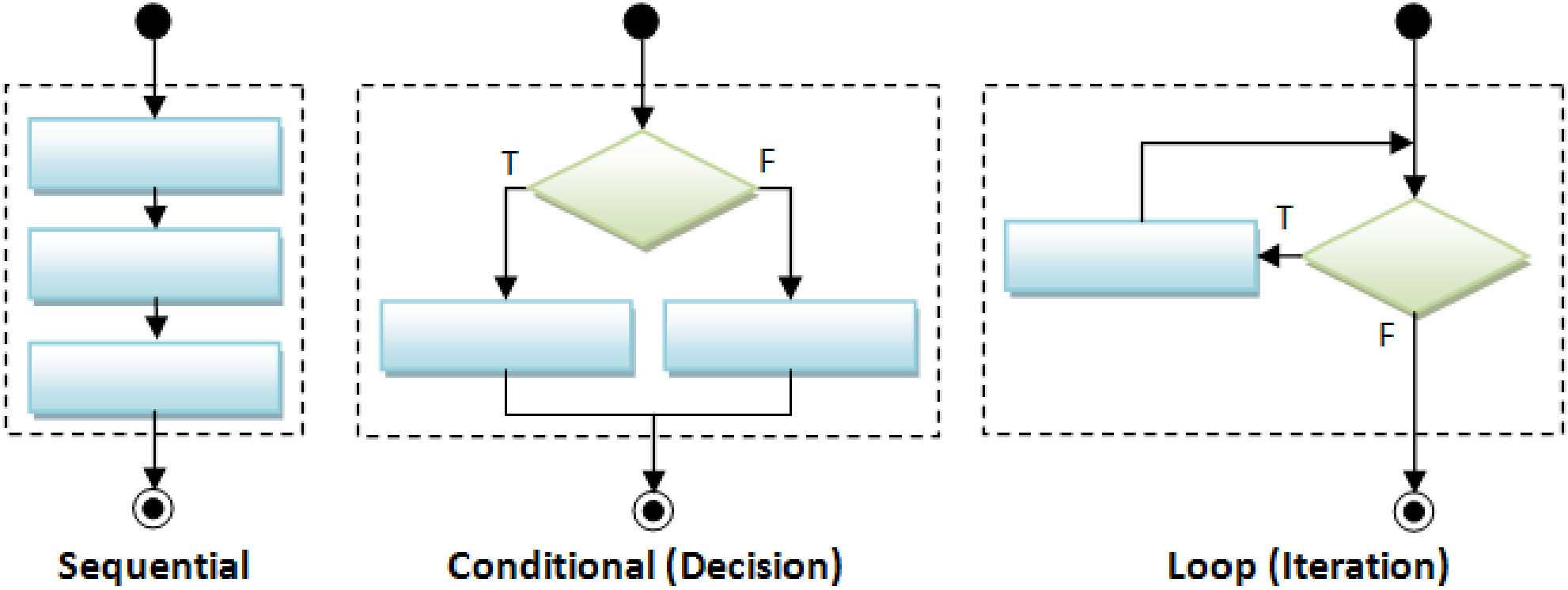


More DS Native to Python - Dictionary

- Mutable
- Unordered
- Key-value pair

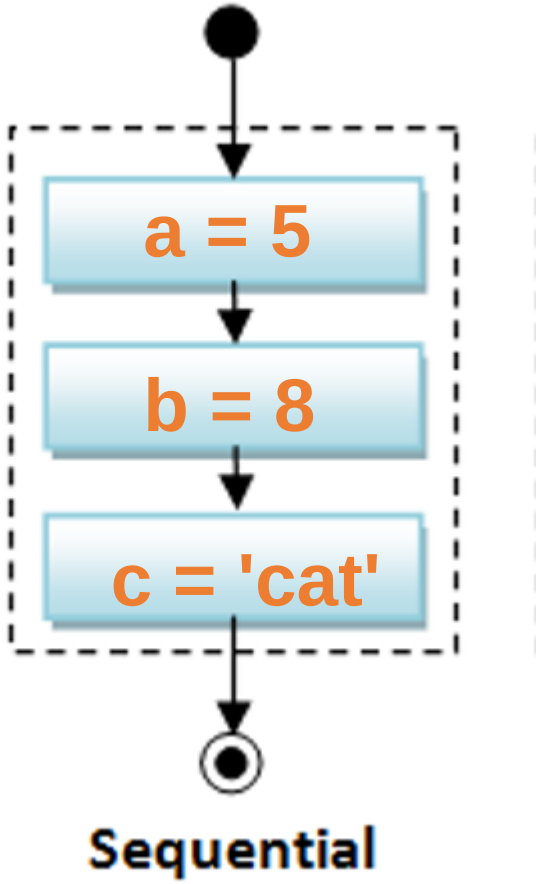


Programming Structures



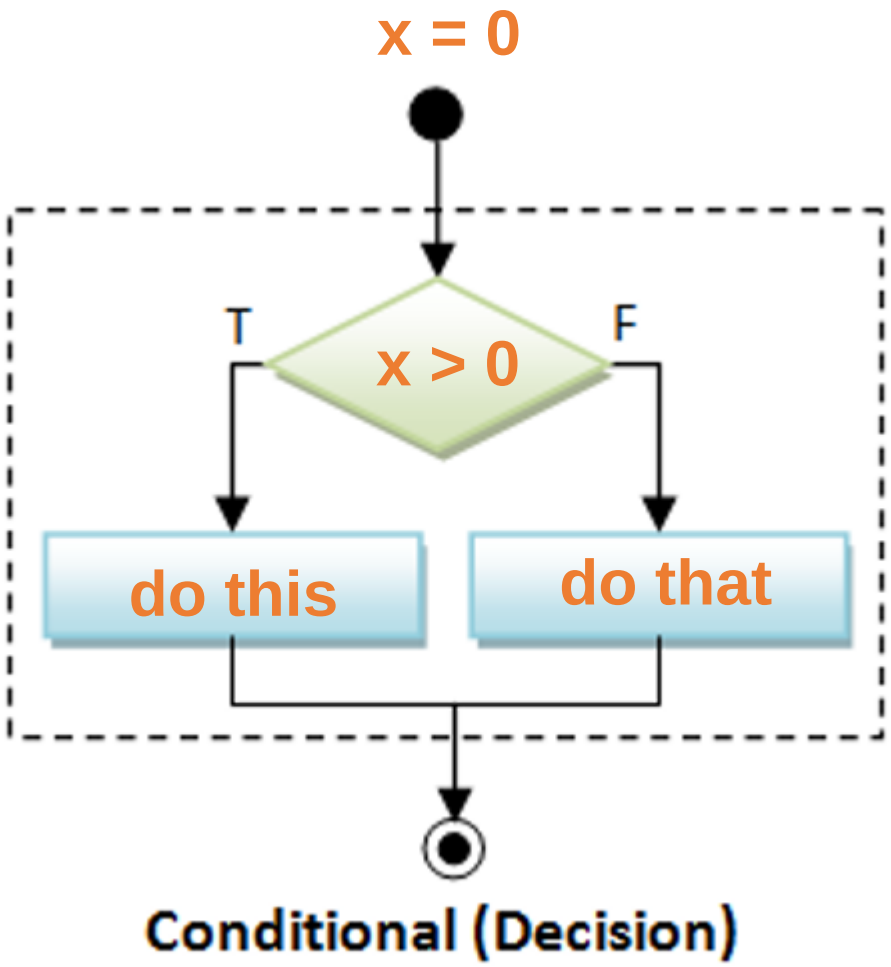
Sequential

- Code executes in sequence



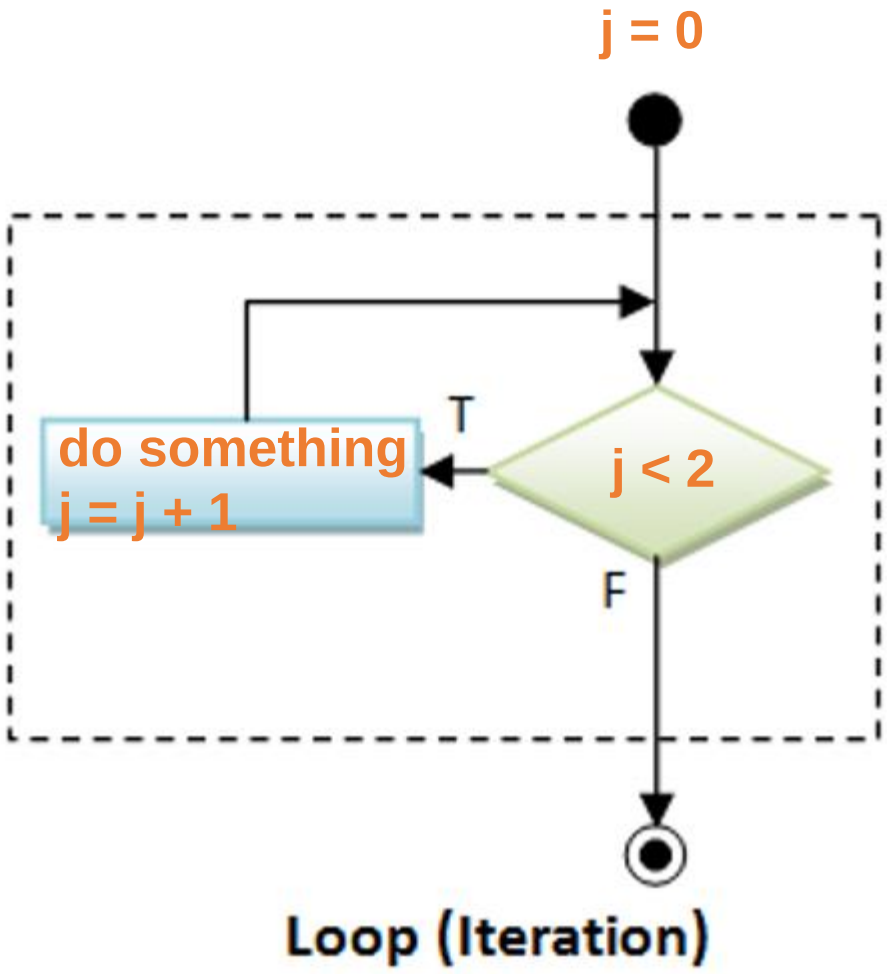
Conditional

- Whether a certain block of code is executed or not depends on whether a condition is satisfied.



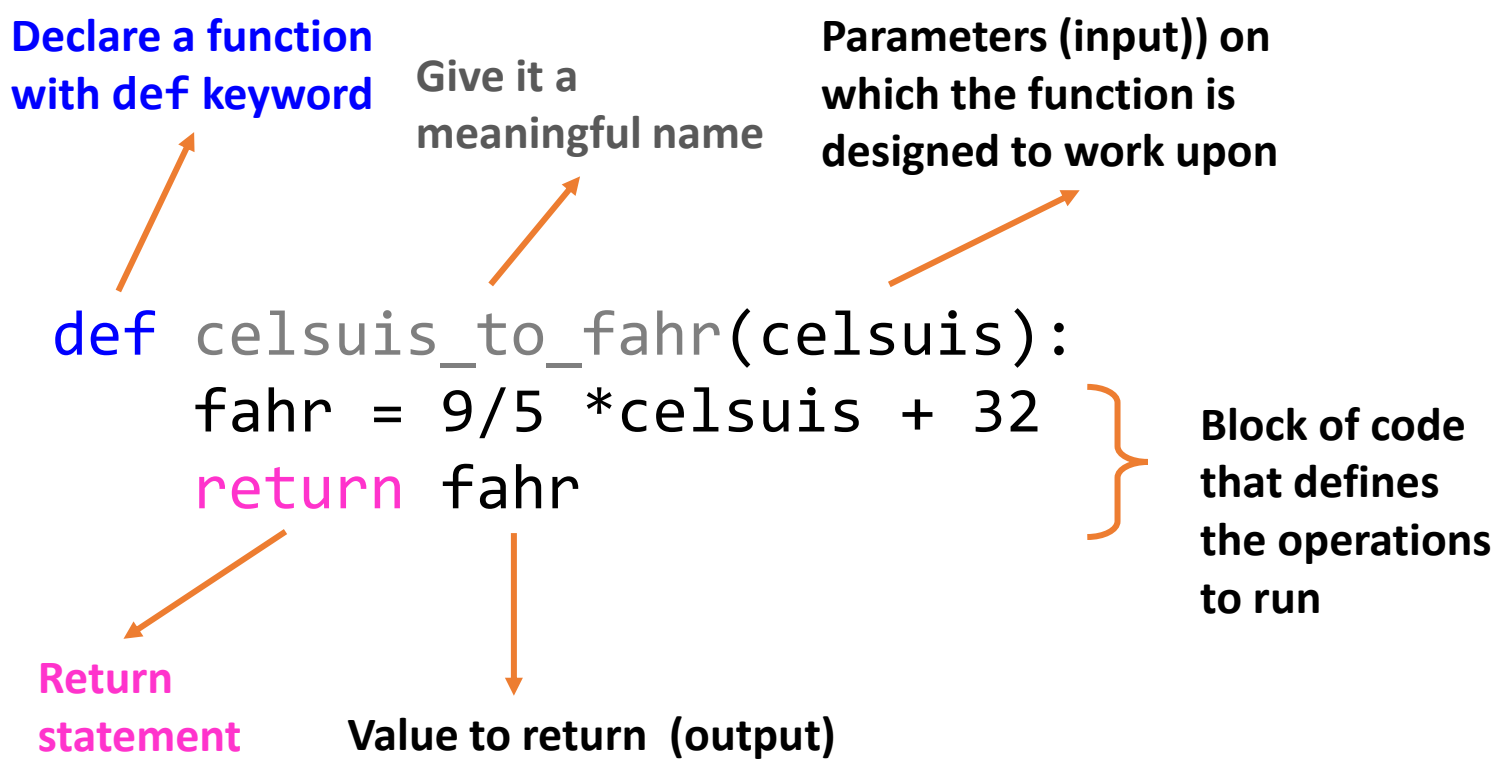
Iterative

- A block of code repeatedly executed for either identical tasks or similar tasks



Function – Custom Functions

- What's a function
  - a logical block of code
  - input -> output
- Why write functions
  - Reusability
  - Abstraction
  - Maintainability



Other Functions

- Built-in functions

- A method is
  - A functions associated with an object (an instance of a class)
  - Accessed using the dot operator (`.`)

```
# create a list
num_list = [4, 8, 10, 15]

# print() and sum() are
# built-in functions
print(sum(num_list))

# remove is a method associated
# with a list object
num_list.remove(10)
print(num_list)
```

Functions/Methods From Other Packages

- Third-party packages/libraries offer functions for various of tasks

- Useful data science packages
  - numpy: operations on vectors and matrices/arrays.
  - pandas: processing 2D tables (dataframes).
  - matplotlib: plotting.
  - scikit-learn: machine learning.

```
# import the numpy module
import numpy as np

# create a 2x3 array
# array() is a function provided by numpy
ar = np.array([[1, 2, 3],
               [4, 5, 6]])

# print the array
print(ar)

# find the largest element in the array
# max() is a method associated with the array object
print(ar.max())

# find the array's shape
# shape is an attribute of the array object;
# it's not a method or function
print(ar.shape)
```

Data Analysis Basics

- Let’s walk through the notebook together
  - Simple data processing (operations on 2D tables/dataframes)
  - Simple analysis on a stock price time series
  - Predict next-day stock price with a linear regression model