INTRODUCTION TO

PYTHON

TABLE OF CONTENT

- Why Python
- Applications of Python
- Python and its features
- Packages overview
  - NumPy
  - Pandas
  - Scikit Learn
  - Matplotlib
  - Seaborn
  - TensorFlow
- Installation steps
- Jupyter Notebook for Python and shortcuts

Python’s benevolent dictator for life

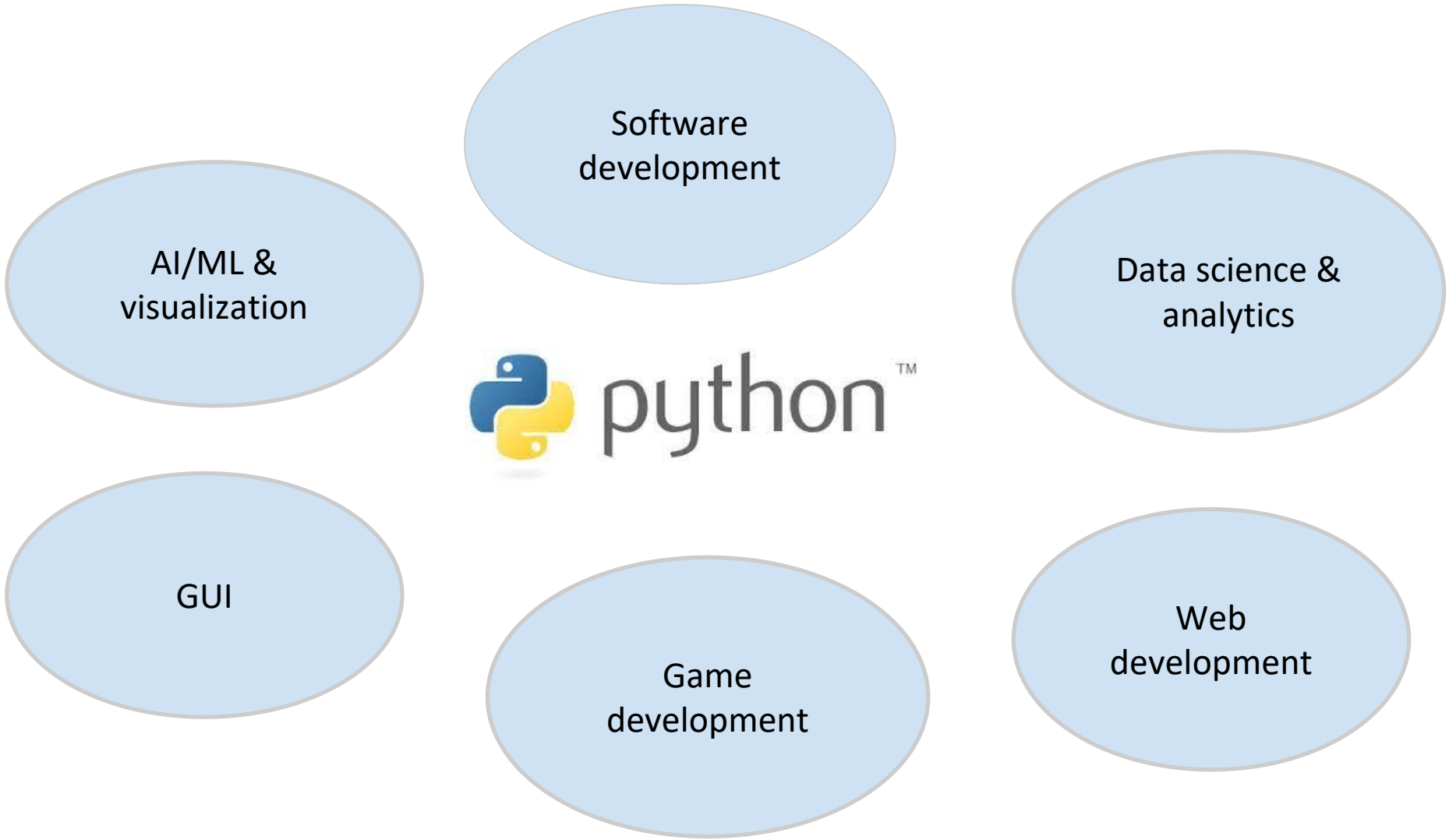
“Python is an experiment in how much freedom programmers need. Too much freedom and nobody can read another's code; too little and expressiveness is endangered.”

- Guido van Rossum

WHY PYTHON

- Easy to understand
- Free and open source
- Huge community support
- Robust standard libraries
- Wide range of applications
- Fast edit-test-debug cycle
- Portable
- Scalable

APPLICATION OF PYTHON

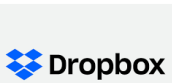


APPLICATION OF PYTHON

Large section of YouTube video sharing system is written in Python.

NASA uses Python to perform many specific tasks.

Web search system uses Python.



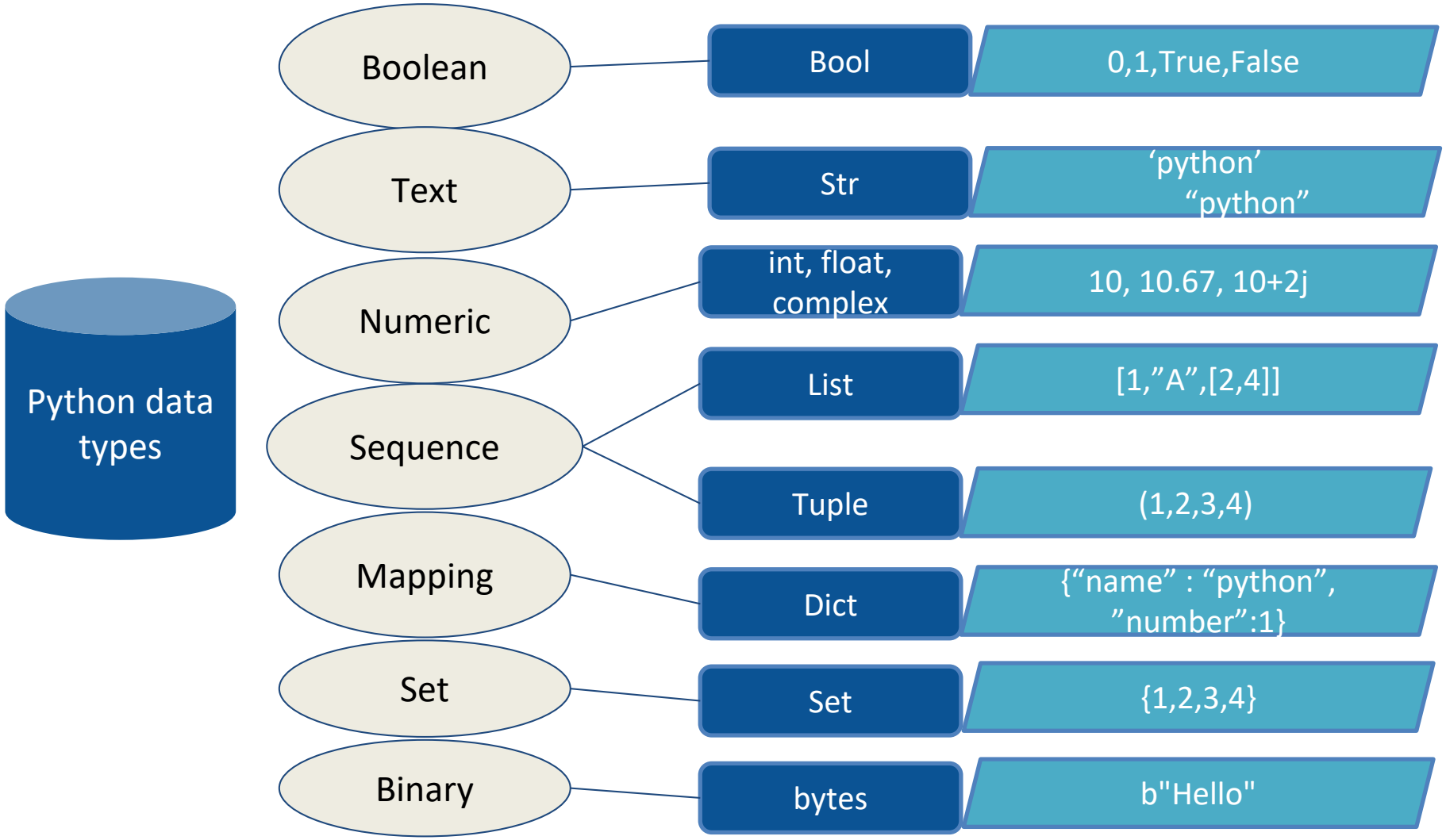
Server & Client software of Dropbox storage use Python.

Peer to peer file sharing system started with Python.

PYTHON & ITS FEATURES

- Object oriented programming
- Interpreted language
- High-level programming
- Dynamic semantics
- High-level built in data structures
- Automatic garbage collection

PYTHON BUILT-IN DATA TYPES



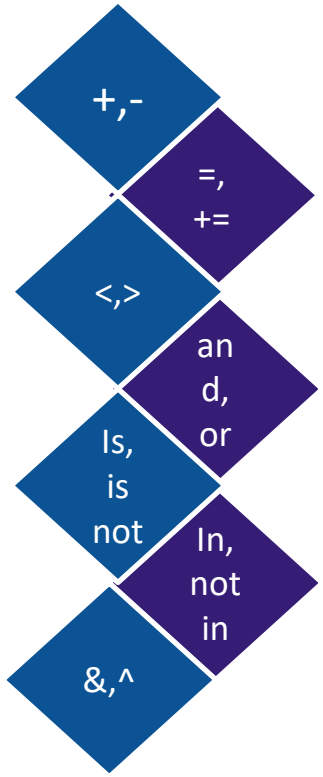
PYTHON OPERATORS

Arithmetic operators

Relational operators

Identity operators

Bitwise operators



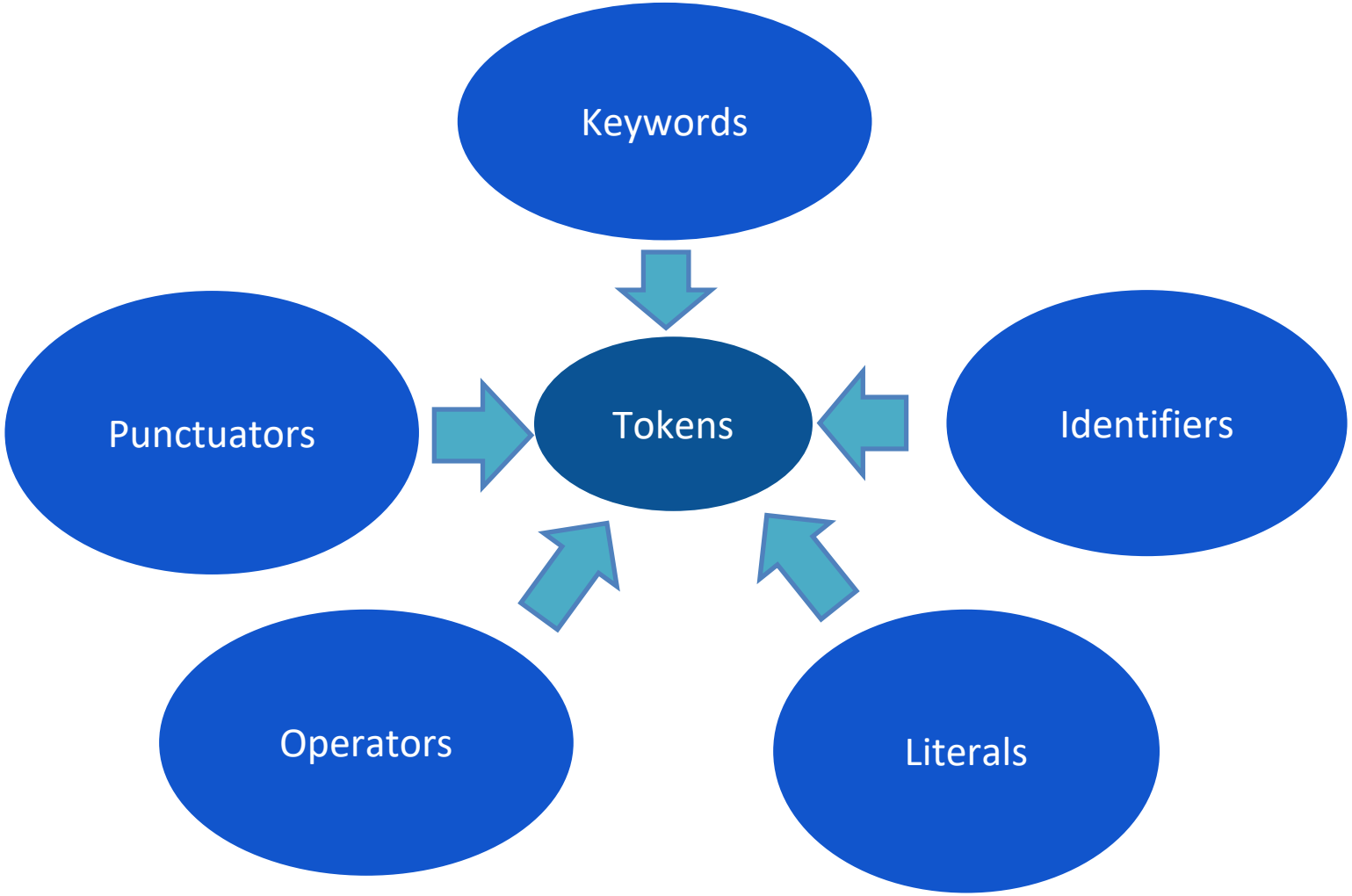
Assignment operators

Logical operators

Membership operators

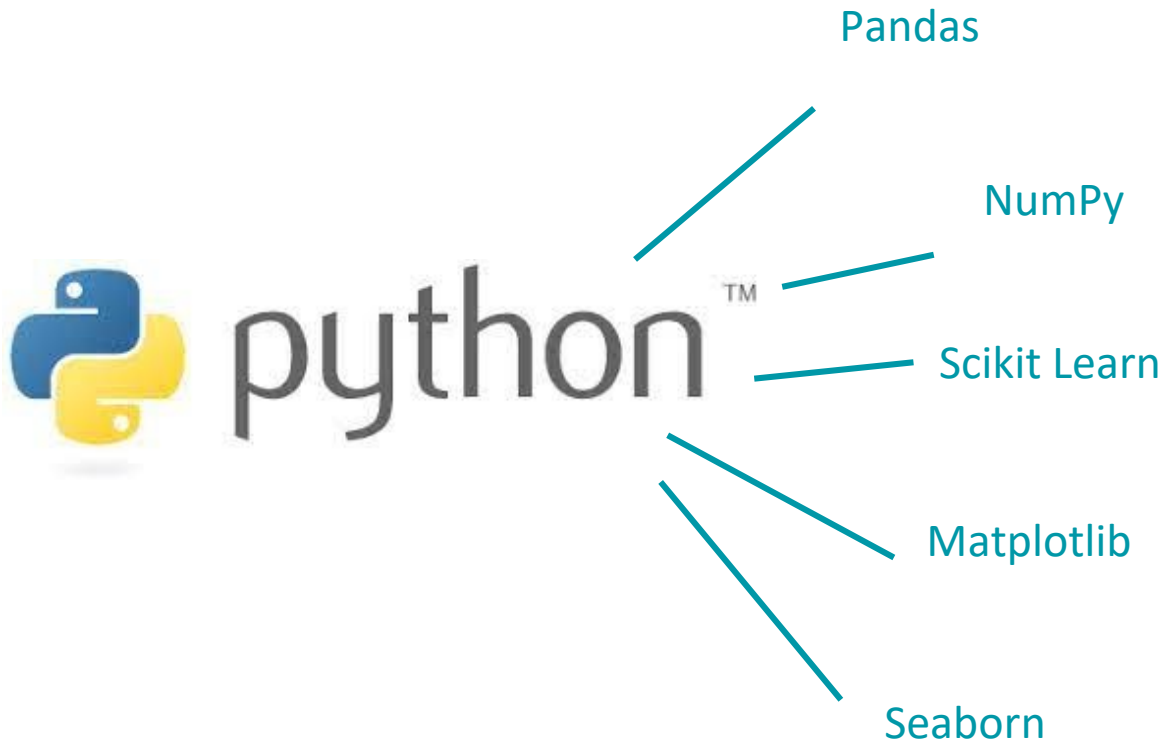
PYTHON TOKENS

Small units of the programming language



PACKAGES OVERVIEW

- A Python package usually consists of several modules.
- Has Build-in modules & Open source modules.
- NumPy, Pandas, Scikit Learn, Matplotlib, Seaborn, SciPy, TensorFlow, Keras, and many more.



OPEN SOURCE PACKAGES

1. NumPy

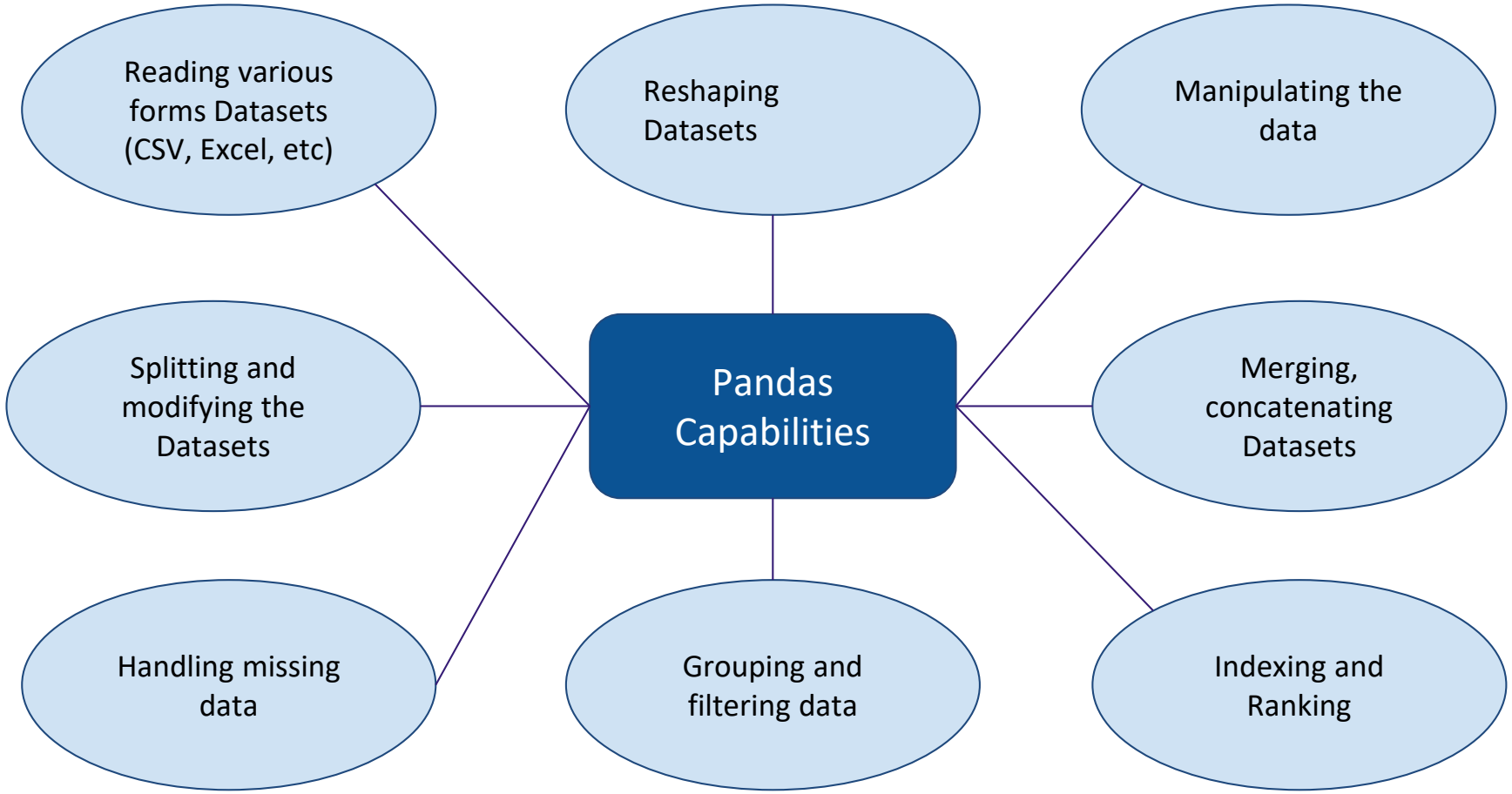
- Adds support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
- Open source library

NumPy

```
In [72]: # Numpy array
         simple_list = [101,102,103,104,105,106,107,108,109,110]
         print(simple_list)

         type(simple_list)

         [101, 102, 103, 104, 105, 106, 107, 108, 109, 110]

Out[72]: list

In [73]: np.array(simple_list)

Out[73]: array([101, 102, 103, 104, 105, 106, 107, 108, 109, 110])

In [74]: simple_list_of_lists = [[10,11,12],[20,21,22],[30,31,32]]
         simple_list_of_lists

Out[74]: [[10, 11, 12], [20, 21, 22], [30, 31, 32]]

In [75]: np.array(simple_list_of_lists)

Out[75]: array([[10, 11, 12],
                [20, 21, 22],
                [30, 31, 32]])
```

2. Pandas

- Mainly used for structured data operations and manipulations.
- Offer powerful data processing capabilities, open source library.



3. Scikit Learn

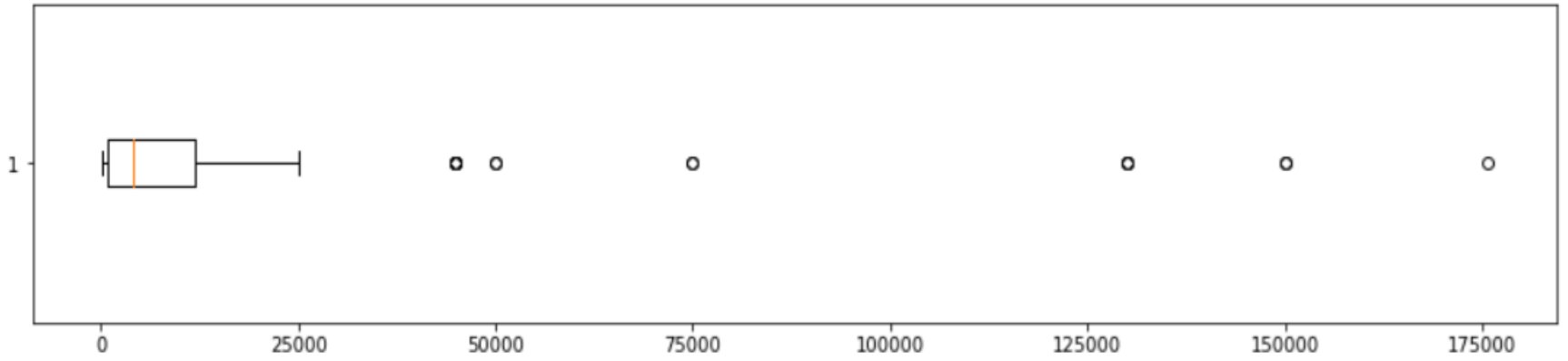
- Contains a huge number of Machine Learning algorithms and other key performance-related libraries
  - Regression
  - Classification
  - Clustering
  - Model Selection
  - Dimensionality reduction and many more

```
In [3]: from sklearn import datasets
        iris = datasets.load_iris()
        digits = datasets.load_digits()

In [4]: digits.target

Out[4]: array([0, 1, 2, ..., 8, 9, 8])

In [5]: from sklearn import svm
        svm_model = svm.SVC(gamma=0.001, C=100.)

In [9]: svm_model.fit(digits.data[:-1], digits.target[:-1])

Out[9]: SVC(C=100.0, gamma=0.001)

In [ ]: svm_model.predict(digits.data[-1:])

In [ ]:
```

4. Matplotlib

- Plotting library for the Python programming language and its numerical mathematics extension NumPy.
- Most of the Matplotlib utilities lies under “pyplot” submodule.

```
In [ ]: # M - Box


fig, ax = plt.subplots(figsize=(15,3))
ax.boxplot(DB['Amount'],vert=False)
plt.show()
```

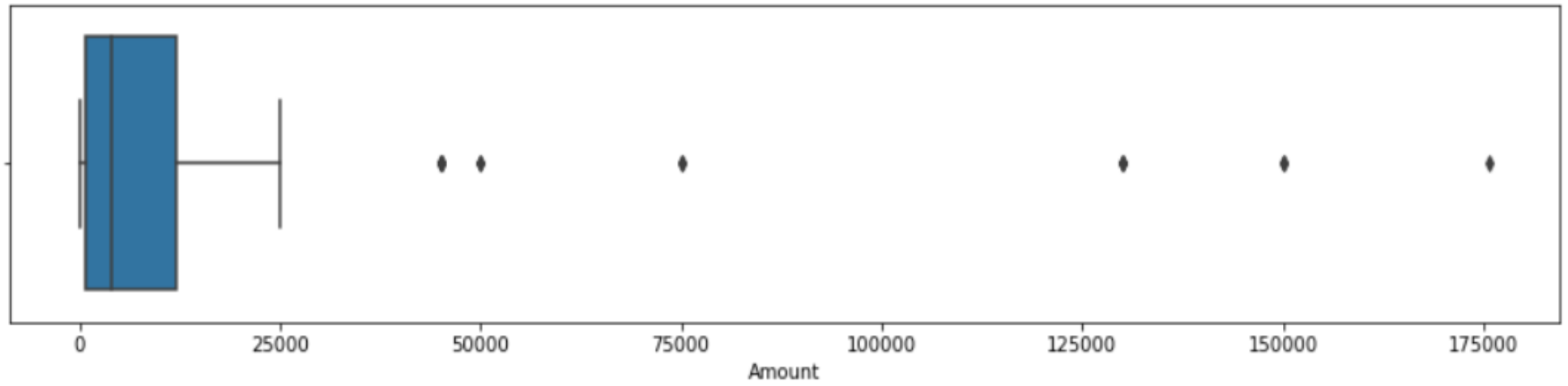


5. Seaborn

- Visualization library for statistical graphics plotting in Python.
- Built on the top of Matplotlib.
- Closely integrated to the data structures from Pandas.
- Seaborn divides plot into the below categories
  - Relational plots
  - Categorical plots
  - Distribution plots
  - Regression plots
  - Matrix plots
  - Multi plot grids
- Examples: displot, boxplot, countplot, etc.

5. Seaborn

```
# S - Box

import seaborn as sns
import matplotlib.pyplot as plt
fig, ax = plt.subplots(figsize=(15,3))
sns.boxplot('Amount', data=DB)
plt.show()
```



PYTHON INSTALLATION – Need for Editor

PYTHON IDEs (Integrated Development Environment)

- A text editor helps to automate the tasks and enhance the **productivity and efficiency** of the developer.
- Why IDE ? :
  - Provides an editor designed to handle code (with, for example, syntax highlighting and auto-completion).
  - Provides build, execution, and debugging tools.
  - Some form of source control.

Jupyter

IDLE

SPYDER

PYTHON INSTALLATION – ANACONDA DISTRIBUTION

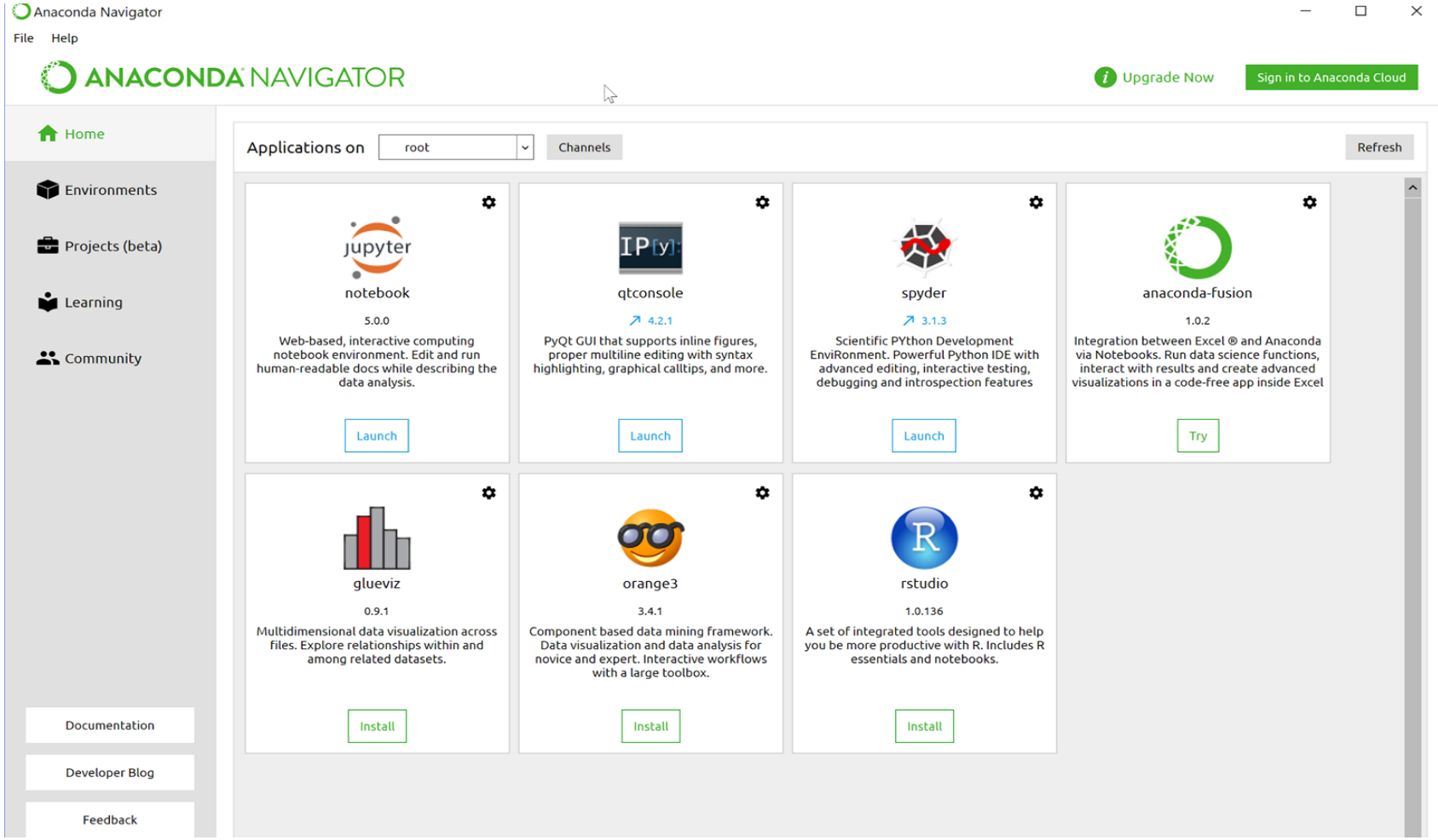
Anaconda distribution comes with over 250 packages automatically installed, and over 7,500 additional open-source packages can be installed from PyPI as well as the conda package.

Anaconda Individual Edition links:

1. Anaconda 5.2 For Linux Installer - https://www.anaconda.com/download/#linux
2. Anaconda 5.2 For macOS Installer - https://www.anaconda.com/download/#macos
3. Anaconda 5.2 For Windows Installer - https://www.anaconda.com/download/#windows

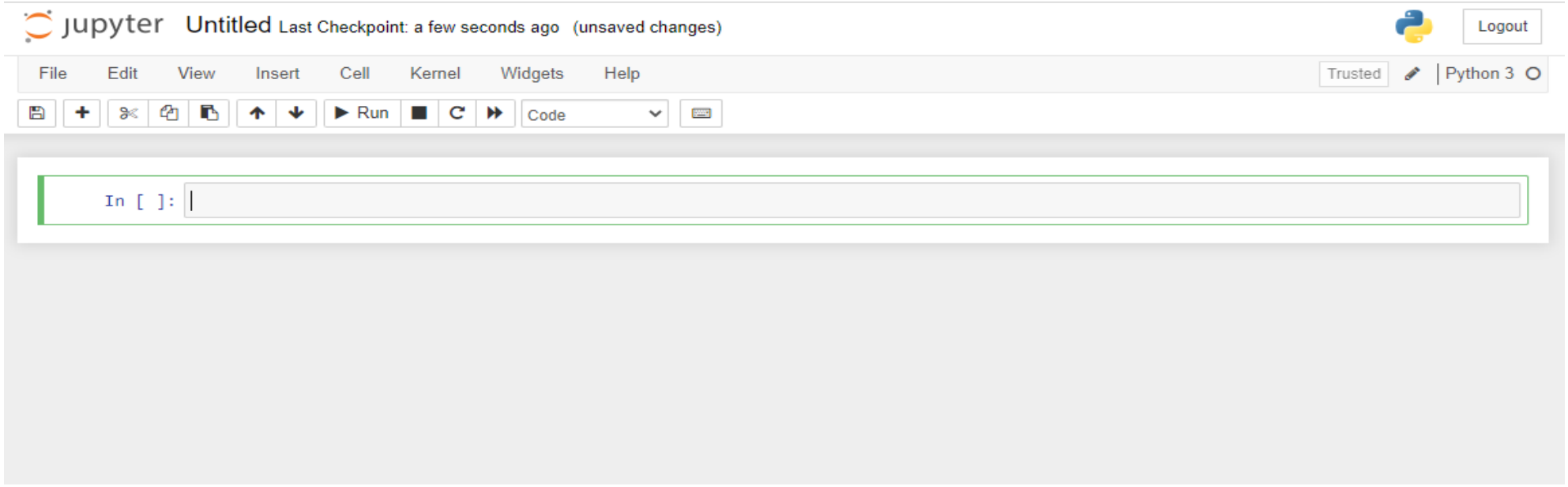
(You need to download the version compatible with your OS.)

ANACONDA NAVIGATOR

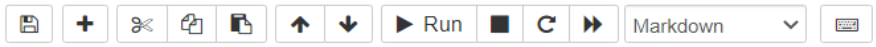


JUPYTER NOTEBOOK FOR PYTHON

- An open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text.
- Anaconda navigator comes with Jupyter notebook.



JUPYTER NOTEBOOK SHORTCUTS



Jupyter Notebook Shortcuts

Keyboard shortcuts

Command	Description
enter	enter edit mode
Command +a; Command +c; Command +v	select all; copy; paste
Command +z; Command +y	undo; redo
Command + s	save and checkpoint
Command + b; Command + a	nsert cell below; insert cell above
Shift + Enter	run cell, select below
Shift + m	merge cells
Command +]; Command + [	indent; dedent
Ctrl + Enter	run cell
Option + Return	run cell, insert cell below
Escape	enter command mode
Escape + d + d	delete selected cell
Escape + y	change cell to code
Escape + m	change cell to markdown
Escape + r	change cell to raw
Escape + 1	change cell to Heading 1
Escape + n	change cell to heading n
Escape + b	create cell below
Escape + a	Insert cell above

SUMMARY

- Discussed need for python
- Basic features of python
- Variables and operators in python
- Packages in Python
- Editor and Jupyter

Hands-On

Thank you!

Happy Learning :)