PYTHON

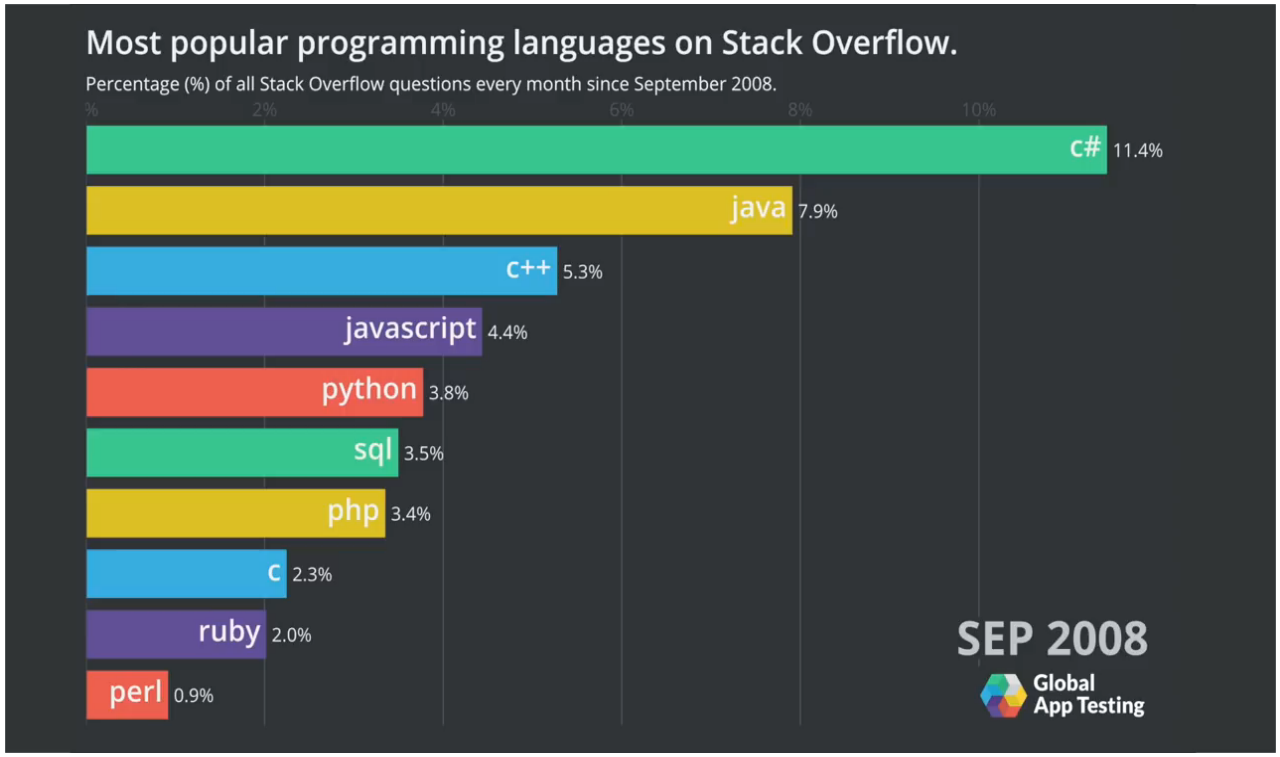
Md. Jalil Piran, PhD
Asst. Professor
Computer Science and Engineering
Sejong University
Fall, 2019

Python

- Python is an **interpreted**, **high-level**, **general-purpose** programming language.
- Created by **Guido van Rossum**
- Released in **1991**.

https://www.python.org/

Guido van Rossum



https://www.youtube.com/watch?v=cKzP61Gjf00

Applications

- Web development (server-side),
- Software development,
- Mathematics,
- Machine learning,
- System scripting.

Advantages

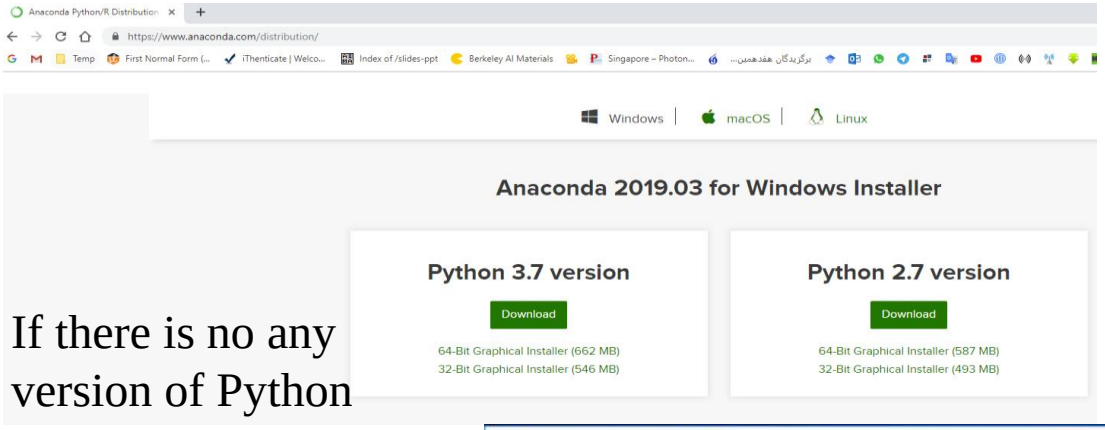
- Simple **syntax**; similar to the English language.
- Runs on an **interpreter** system
- In-built **libraries**
- Moderate **learning curve**
- Easy to **integrate**
- Easy to create **prototypes**
- Free and **open source**
- **Object-oriented** paradigm
- **Portability**
- High **productivity**
- **Platform agnostic** (Windows, Mac, Linux, Respberry Pi, etc.)

Anaconda Navigator

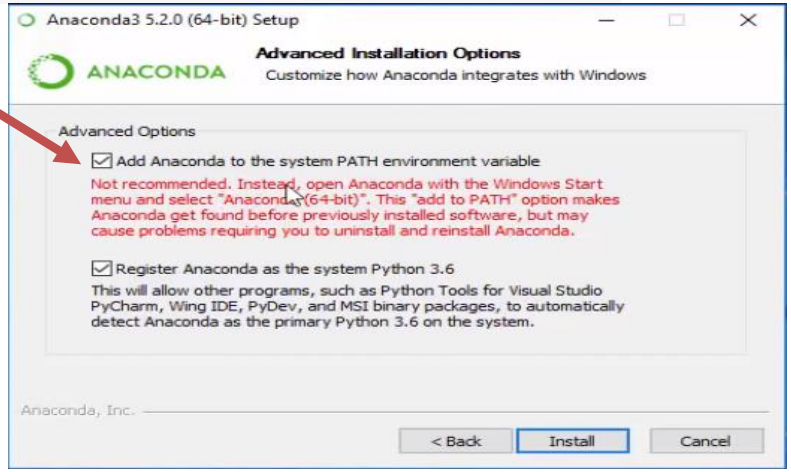
- **Conda** is an open cross-platform language agnostic package management system that is used to install Python packages.
- **Aconda** Navigator is a free and open source Environment of Python and R programming language .
- It is mostly used for data science and machine learning applications.
- It includes some nice tools such as Spyder and Jupyter notebook for Python Coding.

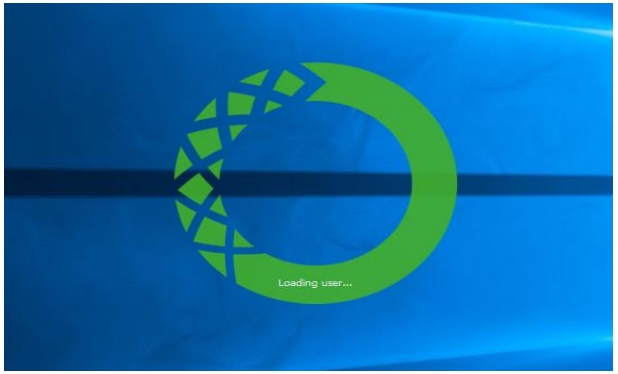
Anaconda Navigator Installation

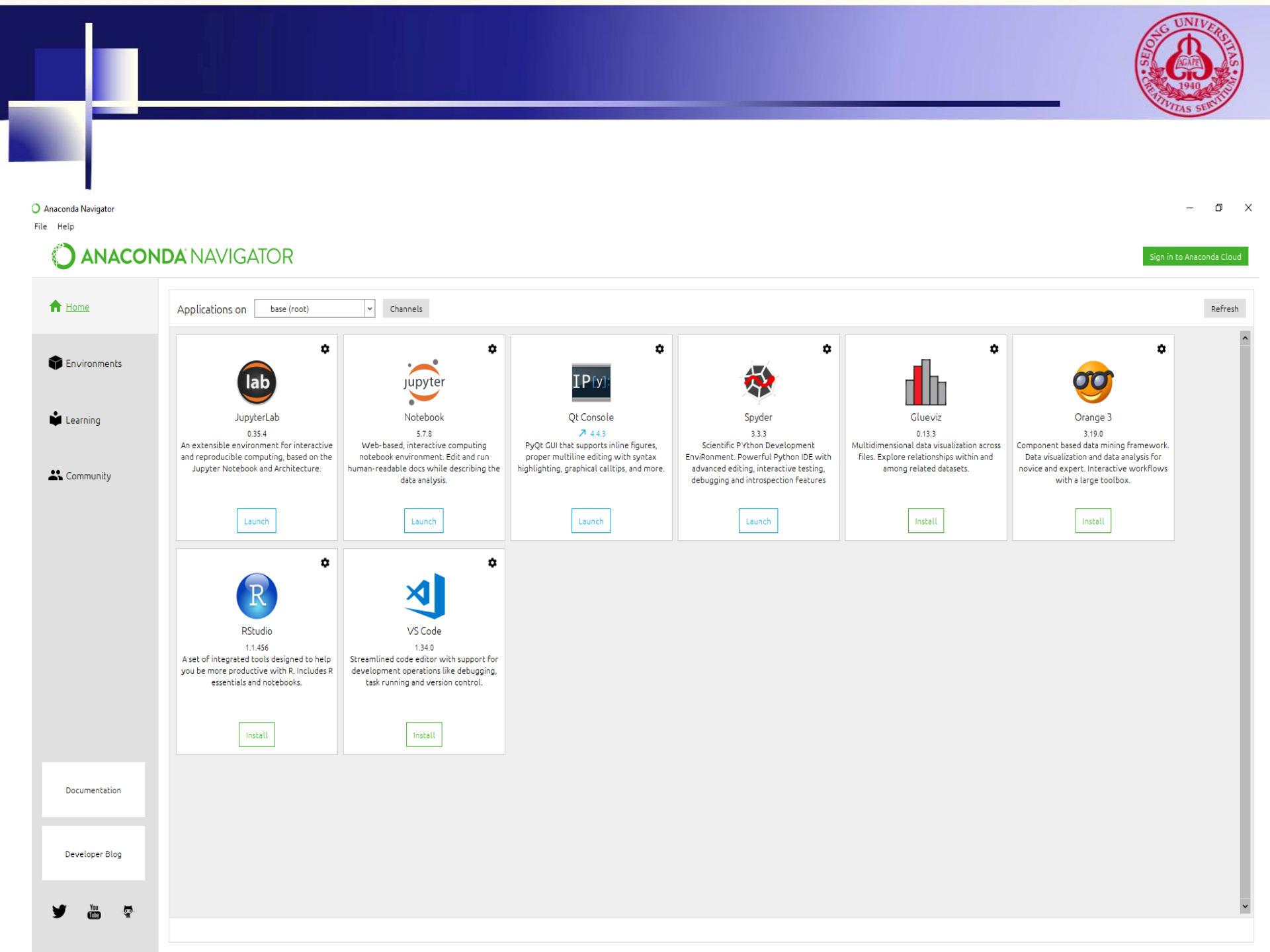
https://www.anaconda.com/distribution/



If there is no any version of Python on your computer check the first option, otherwise uncheck it.



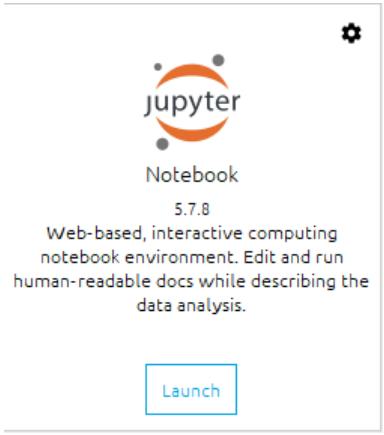


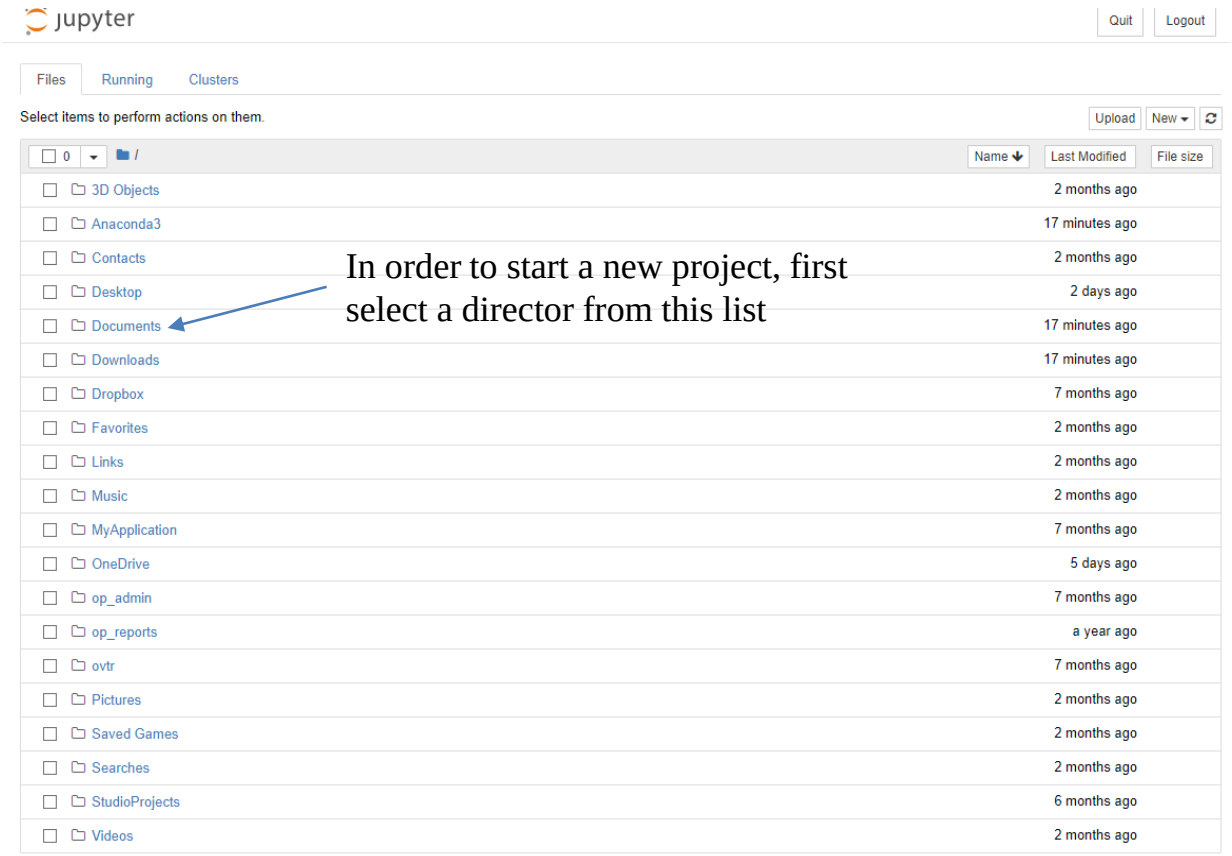

Anaconda Navigator
File Help
ANACONDA NAVIGATOR
Sign in to Anaconda Cloud
Home
Environments
Learning
Community
Documentation
Developer Blog
Applications on
base (root)
Channels
Refresh
JupyterLab
0.35.4
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch
Notebook
5.7.8
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch
Qt Console
4.4.3
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
Launch
Spyder
3.3.3
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Launch
Glueviz
0.13.3
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install
Orange 3
3.19.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install
RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.
Install
VS Code
1.34.0
Streamlined code editor with support for development operations like debugging, task running and version control.
Install

IDEs

- Jupyter
- VSCode
- Rstudio
- PyCharm
- Notepad
- Spyder
- Sublime Text
- Vim / Emacs

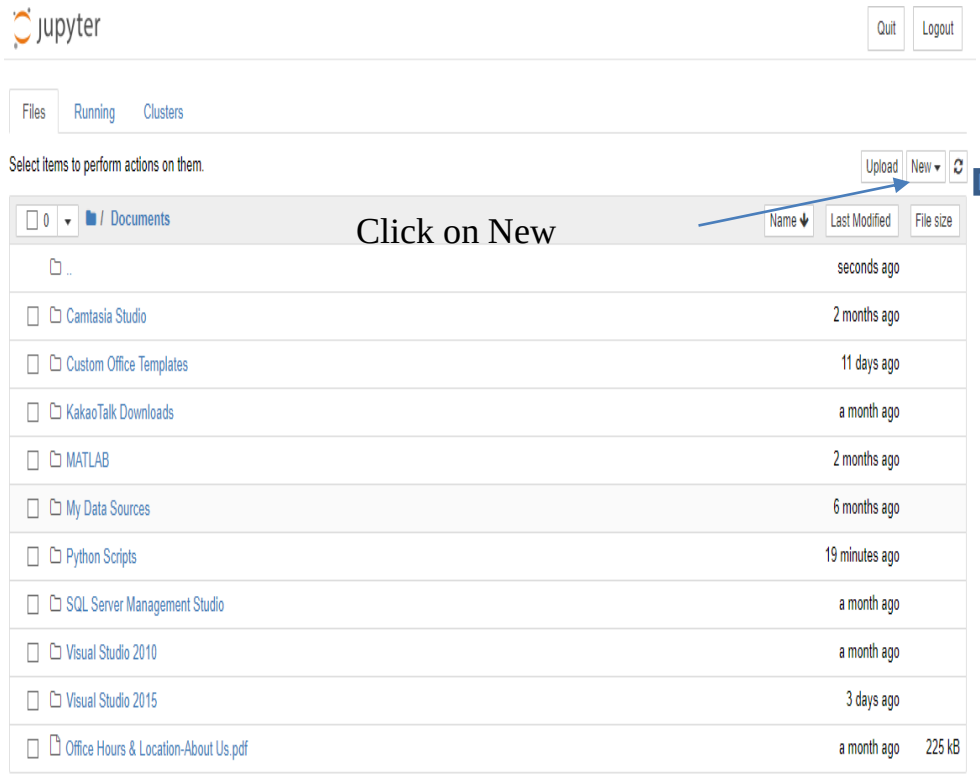
Jupyter Notebook

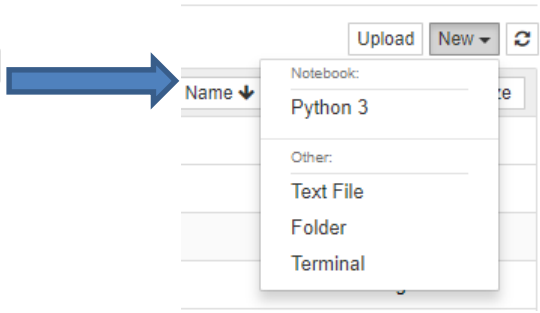


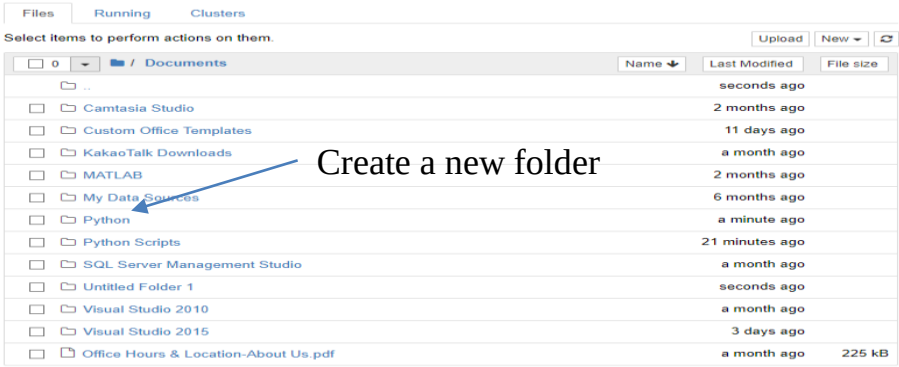


In order to start a new project, first select a director from this list

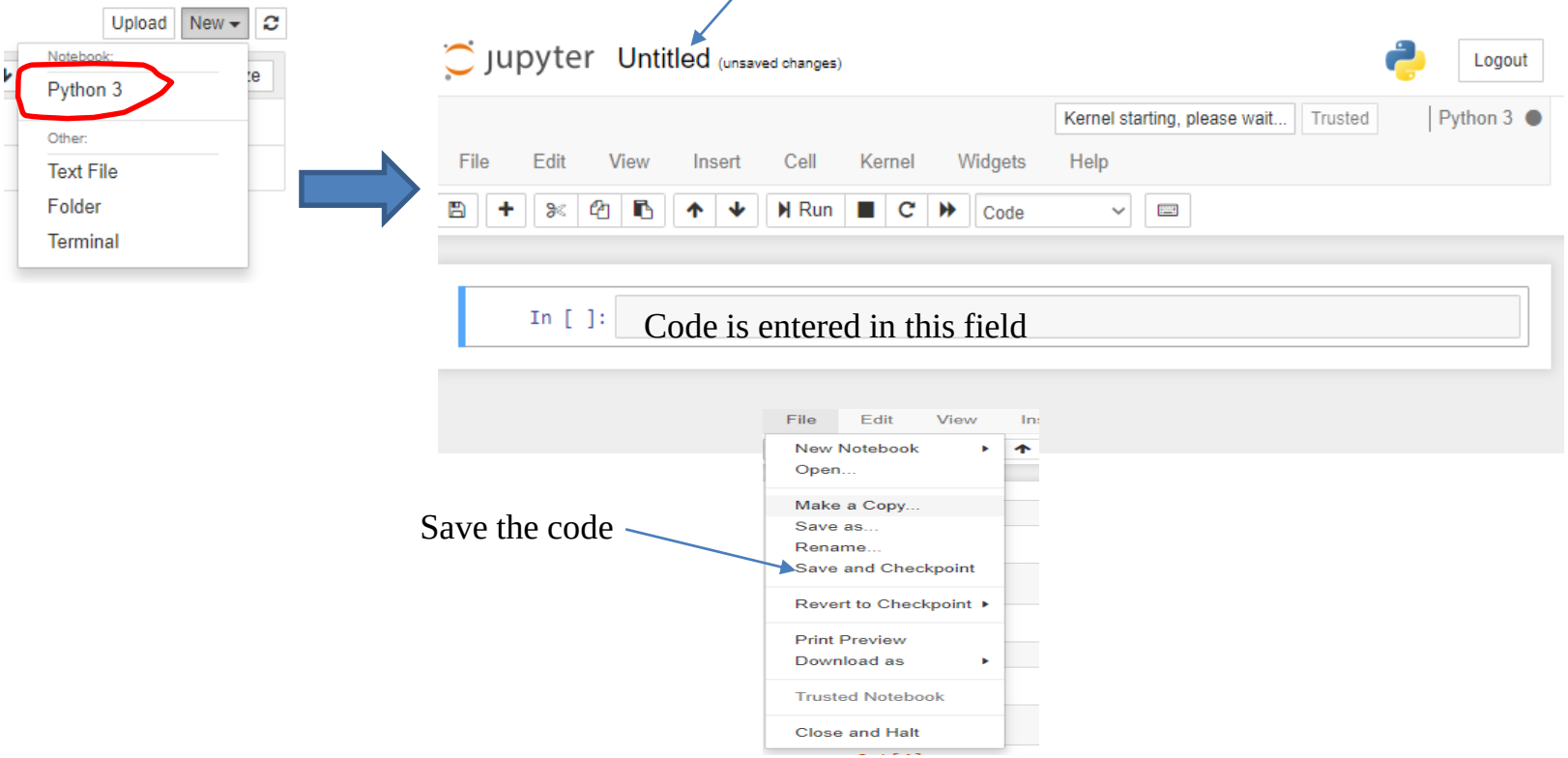
Jupyter Notebook





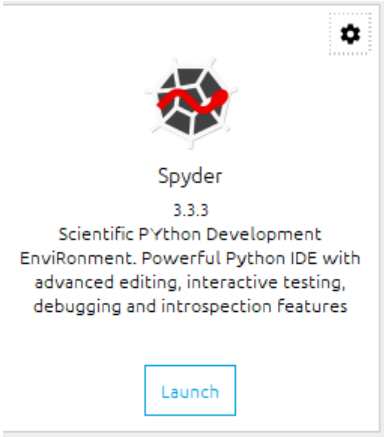


Jupyter Notebook

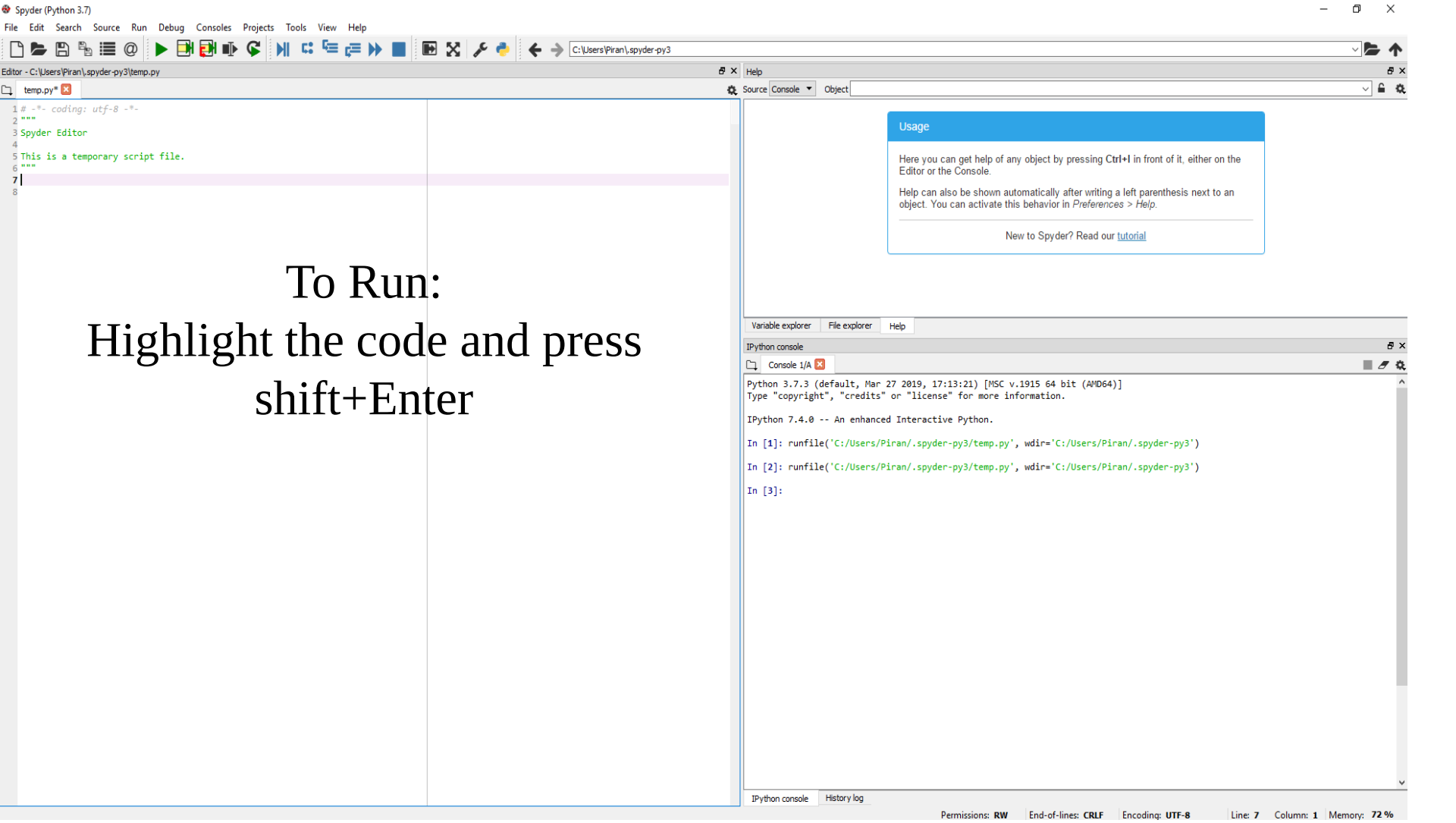


Spyder IDE

- **Spyder** is an open source cross-platform integrated development environment (IDE) for scientific programming in the Python language.



- Spyder integrates with a number of prominent packages in the scientific Python stack, including **NumPy**, **SciPy**, **Matplotlib**, **pandas**, **IPython**, **SymPy** and **Cython**, as well as other open source software.



To Run:
Highlight the code and press shift+Enter

Python Tools

- **Pandas**: used for data analysis
- **Numpy**: multidimensional arrays
- **TensorFlow**: machine learning
- **Matplotlib**, **seaborn**, **Bokeh**: data visualization
- **Keras**: high-level neural network API
- **scikit-learn**: machine learning algorithms
- **SciPy**: algorithms to use with Numpy
- **SQLAlchemy**: Python SQL Toolkit
- **Theano**: Deep Neural Networks
- **SymPy**: Symbolic math
- **AirFlow**, **Dask**, **Luigi**: data engineering tool
- **PyBrain**: machine learning algorithms
- **Pattern**: natural language processing

Google TensorFlow

- A **tensor** is a container which can house data in N-dimensions.
- **TensorFlow** is an open-source machine learning library used for research.
- Use TF to describe computations as a **graph**
- TF **schedules** computations on devices - CPU, GPU...
- Performs **automatic differentiation** (like JuMP!)

Keras

- **Keras** is a model-level library, providing high-level building blocks for developing deep learning models.

- It is an **open-source neural-network** library written in Python.
- It does not handle low-level operations such as tensor products, convolutions and so on itself. Instead, it relies on a specialized, well optimized tensor manipulation library to do so, serving as the "backend engine" of Keras.
- It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML.
- Keras is designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.

install Keras

```
conda install -c anaconda keras
```

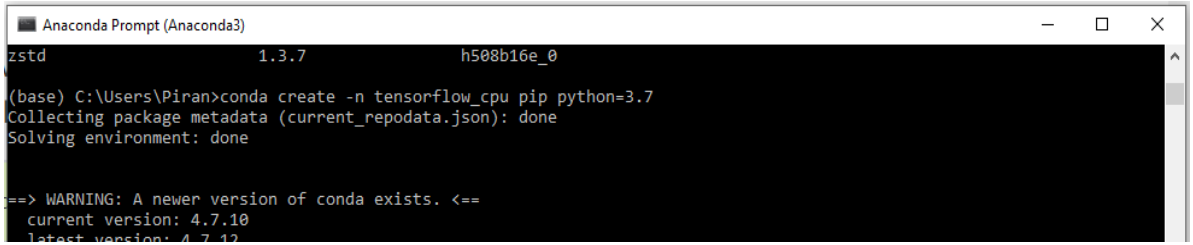
TensorFlow

- Two types:
  - **TensorFlow CPU**; simple to install, slow performance.
  - **TensorFlow GPU**; recommended if a Nvidia graphic card is installed.

- Installation if TensorFlow CPU
  - Open a new *Anaconda/Command Prompt* window
  - Create a new virtual environment with name ‘tensorflow_cpu’

    conda create -n tensorflow_cpu pip python=3.6
  - Activate the newly created virtual environment:

    activate tensorflow_cpu



conda install pip

- Once you have activated your virtual environment, the name of the environment should be displayed within brackets at the beginning of your cmd path specifier, e.g.:

- Then

pip install --ignore-installed --upgrade tensorflow==1.9

- **Note**) the Python version must be 3.6.x! You can see your python version using "conda list" command.

- Test your installation by running

```
python
>>> import tensorflow as tf
>>> hello = tf.constant('Hello, TensorFlow!')
>>> sess = tf.Session()
```

If you see the following, it means successfully done!

```
2019-02-28 11:59:25.810663: I T:\src\github\tensorflow\tensorflow\core\p
```

- Test it:

```
>>> print(sess.run(hello))
b'Hello, TensorFlow!‘
```

- Install Keras

```
pip install Keras
```

Note you no longer need to import Keras, use ‘tf.keras’ instead

Kaggle

- An **online community** of data scientists and machine learning.

- It allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges.

Office: #432, Dayang AI Center, **Phone**: 010-8999-8586, **email**: *piran@sejong.ac.kr*