Pelatihan ABCD
Modul 1-2: Python IDE Setup

Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung

Unviersitas Singaperbangsa Karawang

Python IDE

Setup and Installation

Python Development Environment

Two options:

1. Google Colab (Online)
   - https://colab.research.google.com/
2. Local Development (Offline)
   - https://www.anaconda.com/

1. Google Colab (Online)

- https://colab.research.google.com/
- Free, only requires Google account
- Online, no installation necessary, internet required
- allow to run on GPU for free as well, with limited disk and RAM

2. Local Installation (Offline)

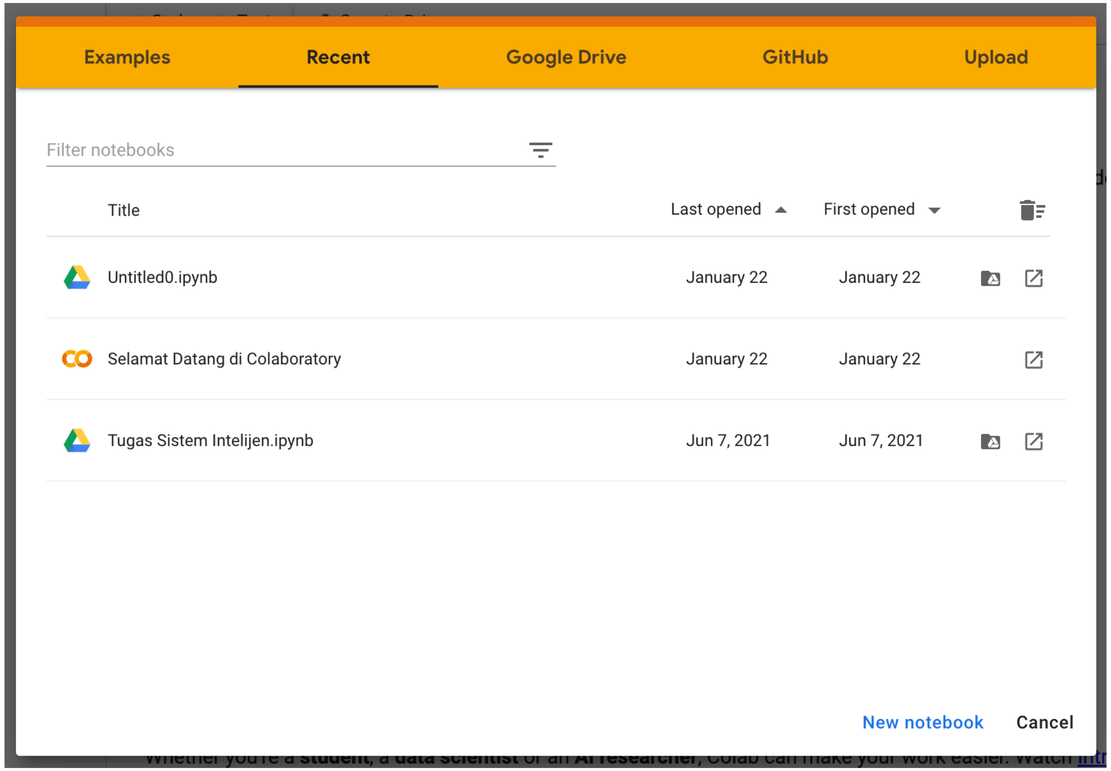
- Requires installation, Pyhton and few libraries
- Offline, does not require internet access
- No memory limits, depend on our local hardware

Google Colab

https://colab.research.google.com/

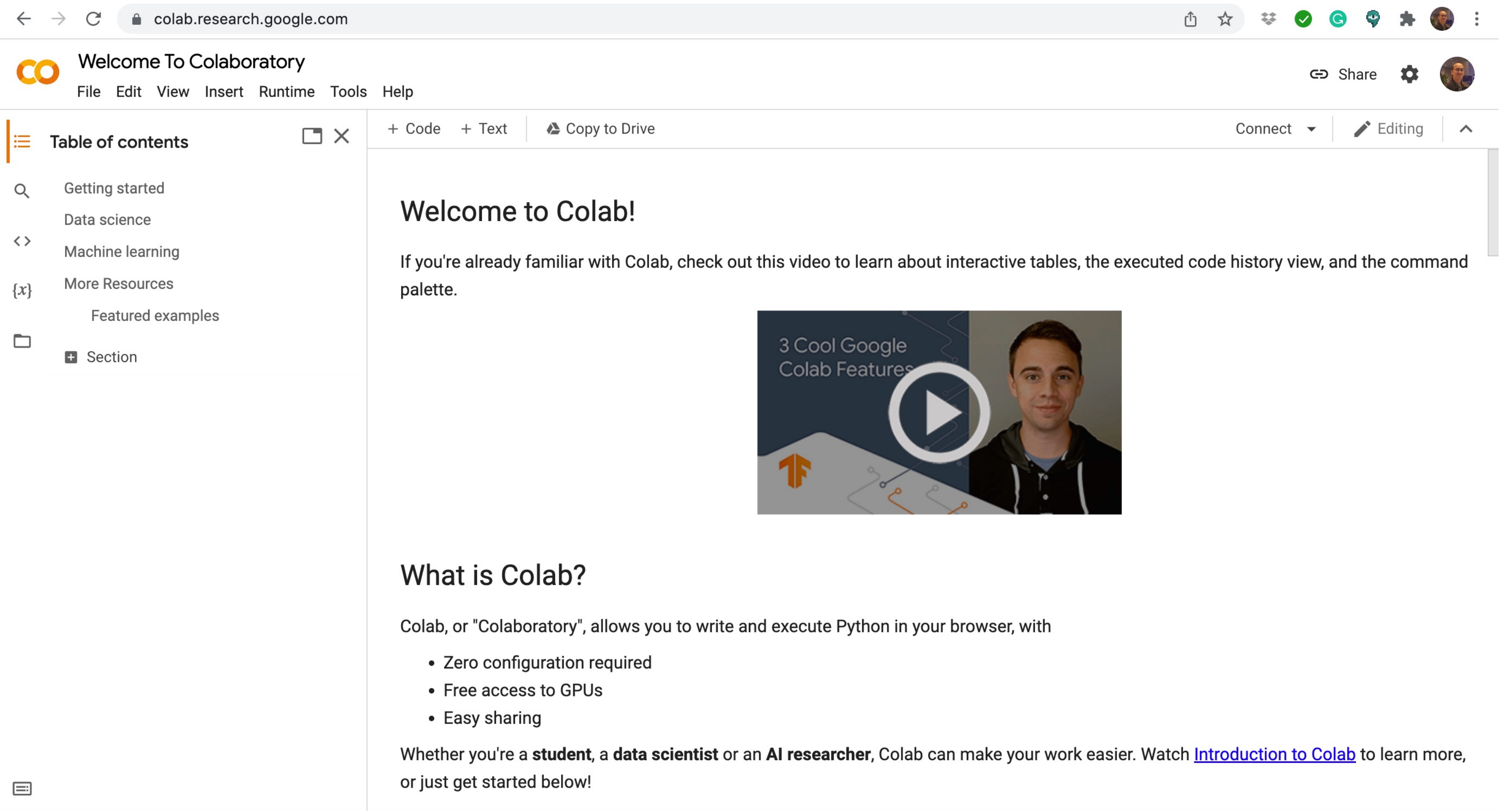
Google Colab First Project

- Login to google, browse to: https://colab.research.google.com/
- Explore the the welcome menu
- Choose and Click New Notebook or File > New Notebook



Title	Last opened	First opened
Untitled0.ipynb	January 22	January 22
Selamat Datang di Colaboratory	January 22	January 22
Tugas Sistem Intelijen.ipynb	Jun 7, 2021	Jun 7, 2021

Welcome to Colaboratory



Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view, and the command palette.

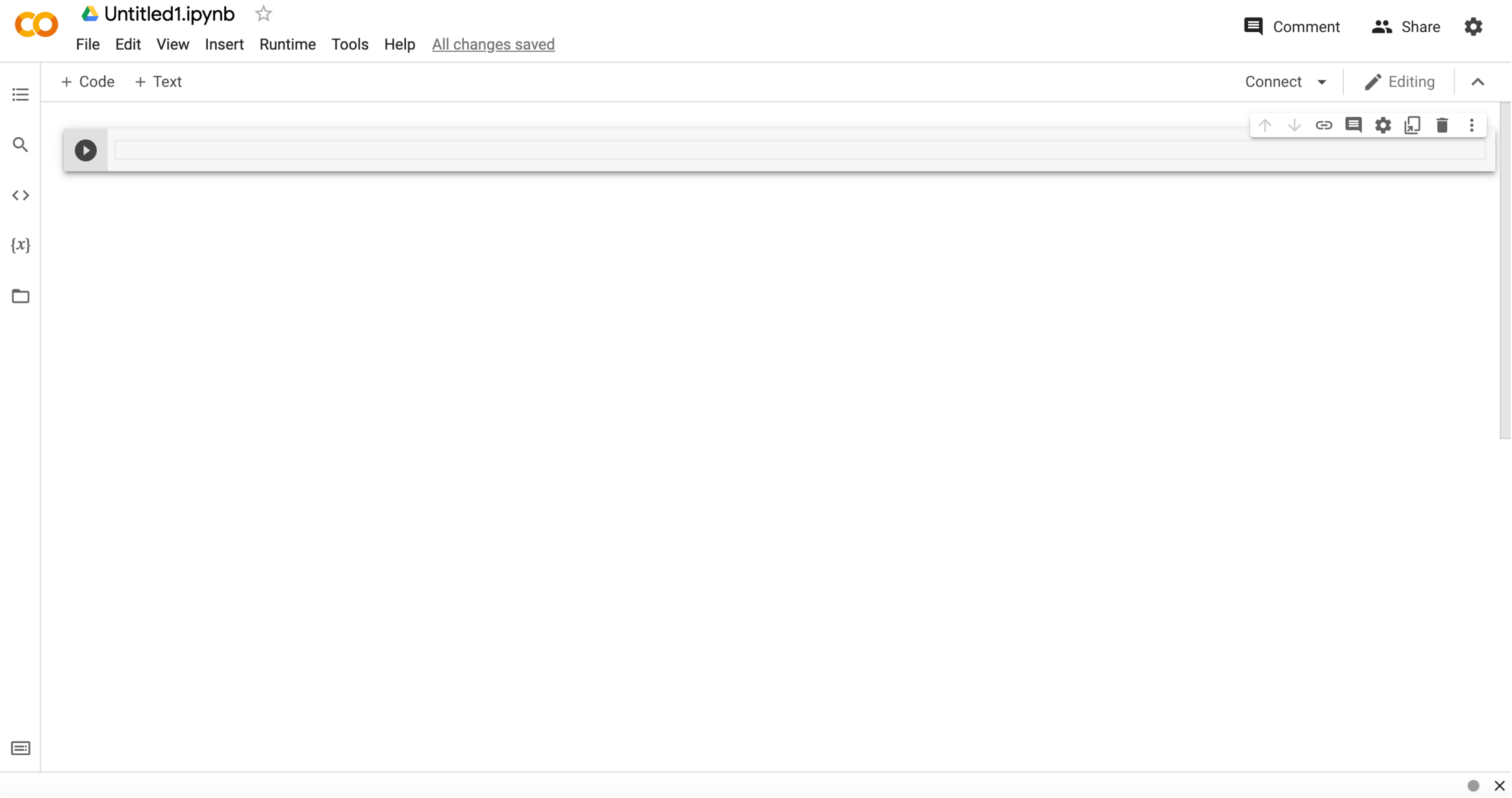
What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

- Zero configuration required
- Free access to GPUs
- Easy sharing

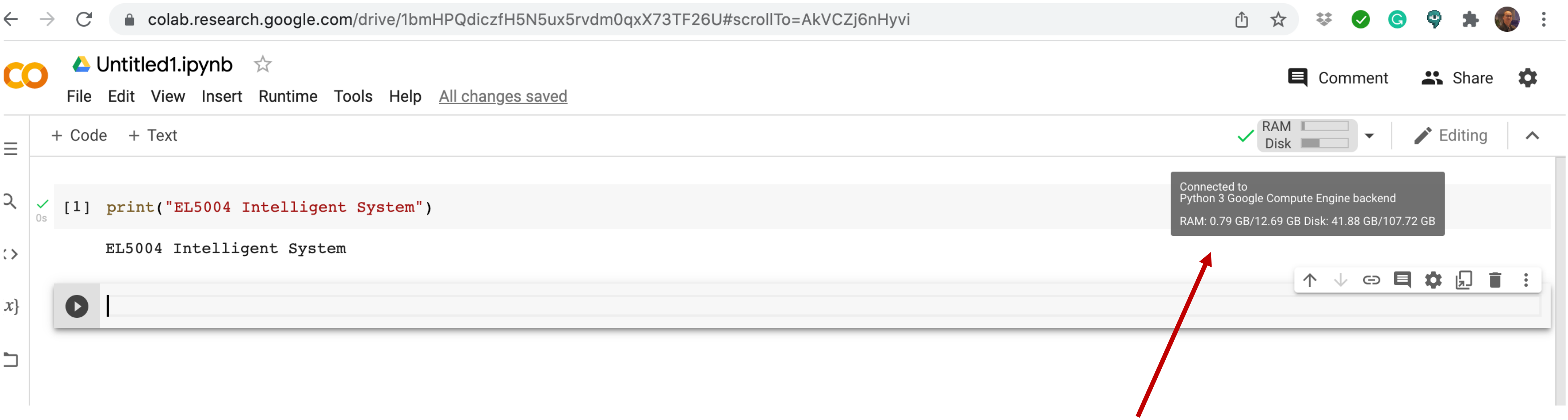
Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch Introduction to Colab to learn more, or just get started below!

Colab IDE


CO
Untitled1.ipynb
File
Edit
View
Insert
Runtime
Tools
Help
All changes saved
Comment
Share
+ Code
+ Text
Connect
Editing
{x}

Type and Run Code Inside the Cell

- Type the code, and press **Shift + Enter** to Run the code



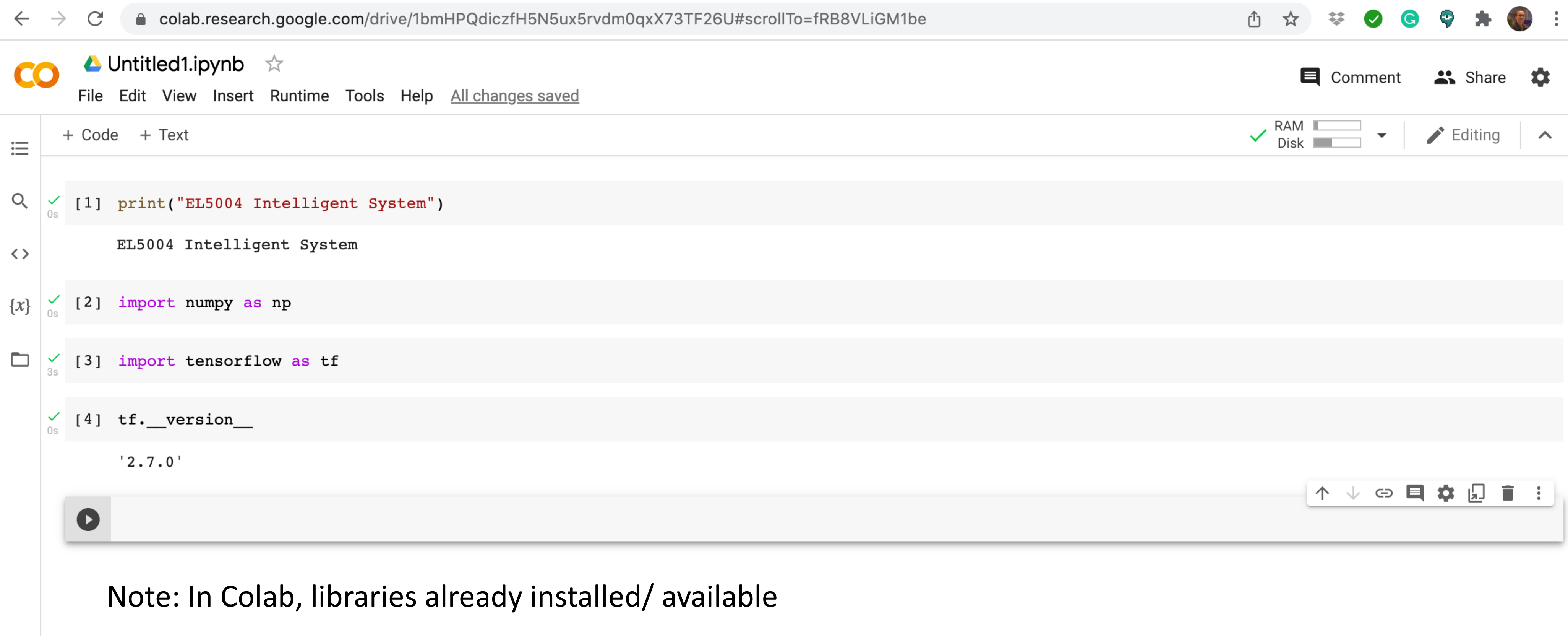
```
[1] print("EL5004 Intelligent System")

    EL5004 Intelligent System
```

Connected to
Python 3 Google Compute Engine backend

RAM: 0.79 GB/12.69 GB Disk: 41.88 GB/107.72 GB

Shows how much disk space and RAM we have at the back end

Import Libraries



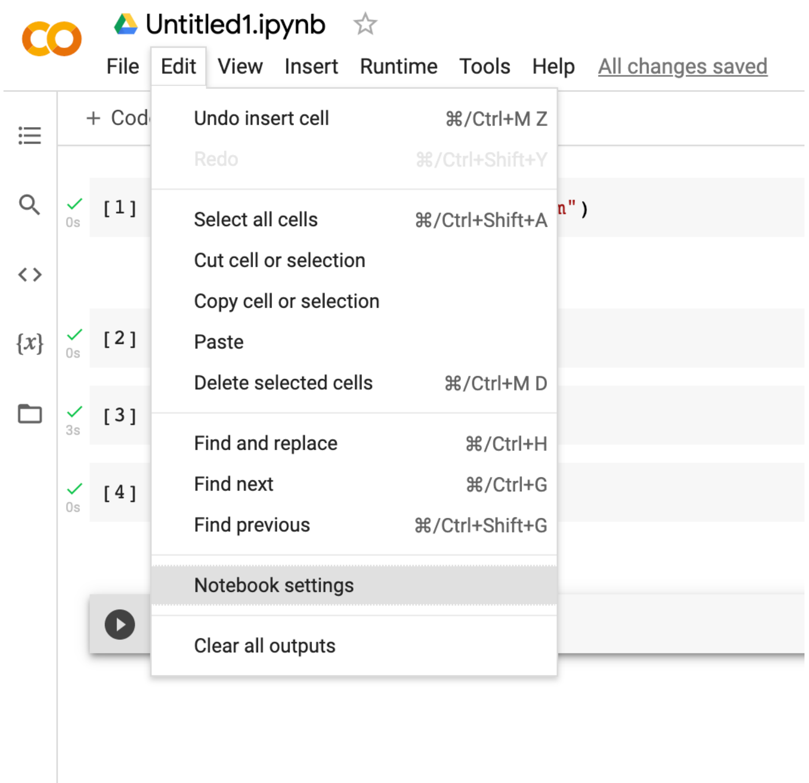
colab.research.google.com/drive/1bmHPQdiczfH5N5ux5rvdm0qxX73TF26U#scrollTo=fRB8VLiGM1be
Untitled1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing

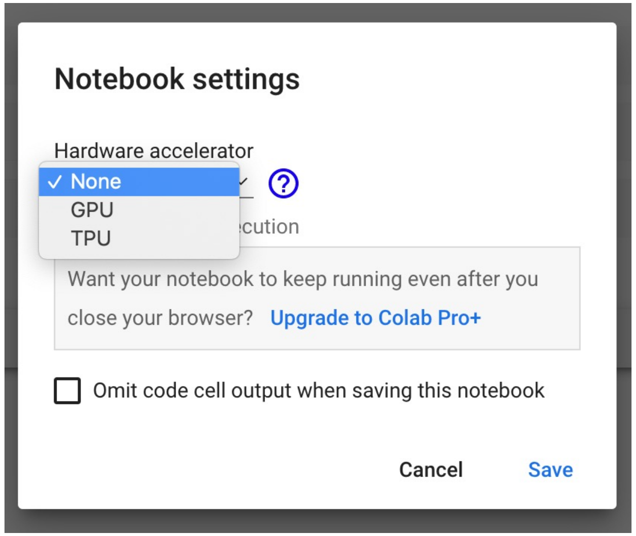
```
[1] print("EL5004 Intelligent System")

    EL5004 Intelligent System

[2] import numpy as np

[3] import tensorflow as tf

[4] tf.__version__

    '2.7.0'
```

Note: In Colab, libraries already installed/ available

Using GPU/TPU

- Choose Edit > Notebook Setting





Notebook settings

Hardware accelerator

✓ None
GPU
TPU

cution

Want your notebook to keep running even after you close your browser? Upgrade to Colab Pro+

□ Omit code cell output when saving this notebook

Cancel Save

GPU = Graphics Processing Unit
TPU = Tensor Processing Unit

Local Development using Jupyter Notebook

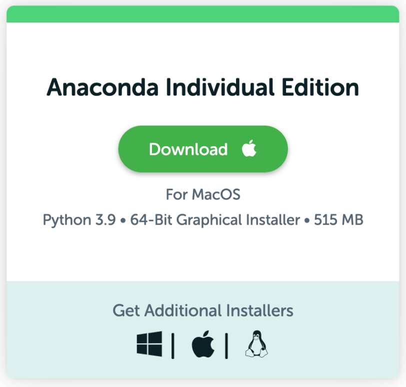
https://www.anaconda.com/

Setup and Installation Step by Step

1. Download and install Anaconda
2. Creating the environment
3. Activating the environment
4. Install required libraries
5. My First jupyter notebook project

1. Anaconda

- Most popular Python data science (DS) and machine learning (ML) platform
- Thousands of open-source packages and libraries
- Browse to https://www.anaconda.com/, download and install **Anaconda Individual Edition** (Free edition). Documents: https://docs.anaconda.com/



Anaconda Individual Edition

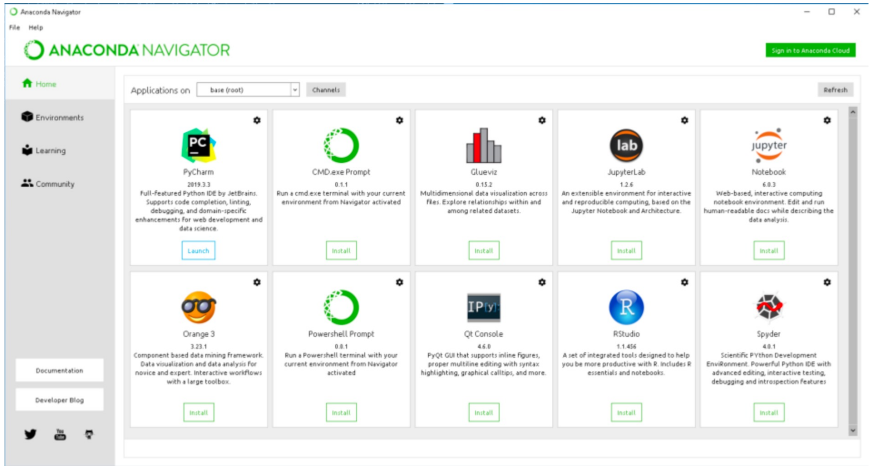
Download

For MacOS

Python 3.9 • 64-Bit Graphical Installer • 515 MB

Get Additional Installers

jupyter
SciPy
SPYDER
NumPy
DASK
bokeh
scikit learn
TensorFlow
Numba
Panel
pandas
$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$
matplotlib
PYTORCH
HoloViews
CONDA



Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home

Environments

Learning

Community

Applications on base (root) Channels Refresh

PyCharm
2019.3.3
Full-featured Python IDE by JetBrains. Supports code completion, linting, debugging, and domain-specific enhancements for web development and data science.
Launch

CMD.exe Prompt
0.1.1
Run a cmd.exe terminal with your current environment from Navigator activated
Install

Glueviz
0.15.2
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install

JupyterLab
1.2.6
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Install

Notebook
6.0.3
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Install

Orange 3
3.23.1
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install

Powershell Prompt
0.0.1
Run a Powershell terminal with your current environment from Navigator activated
Install

Qt Console
4.6.0
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
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

Spyder
4.0.1
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Install

Documentation

Developer Blog

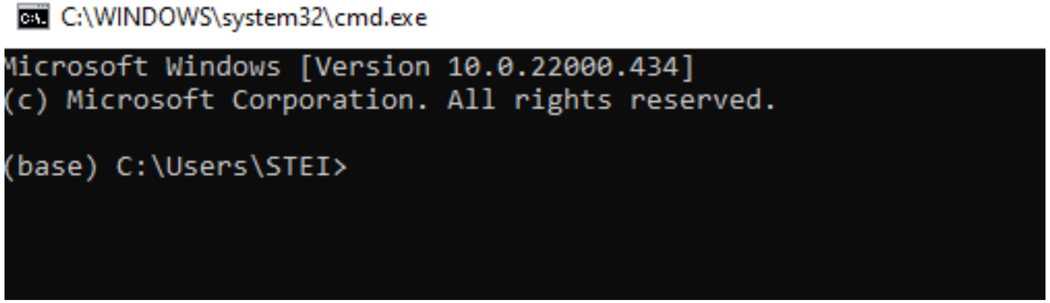
2. Creating the Environment

Create a “room” for us doing experiment

- If there’s a something wrong with the experiment, it will cause the problem only to this experimental room environment. It will not harm to our computer system.

2.1 Restart the computer. Open command line prompt

- Use, Anaconda Prompt, or cmd, or Terminal (MacOs/ Linux only)



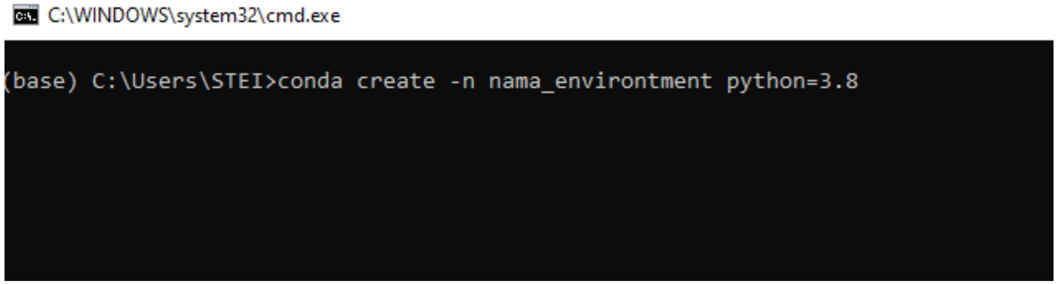
```
Microsoft Windows [Version 10.0.22000.434]
(c) Microsoft Corporation. All rights reserved.

(base) C:\Users\STEI>
```

Base environment (shown explicitely on Anaconda Prompt)

2. Creating the (Virtual) Environment

2.2. Create the environment

- Type: ***conda create –n nama_environtment python==version***
- Example: *conda create –n my_tf_env python==3.8*



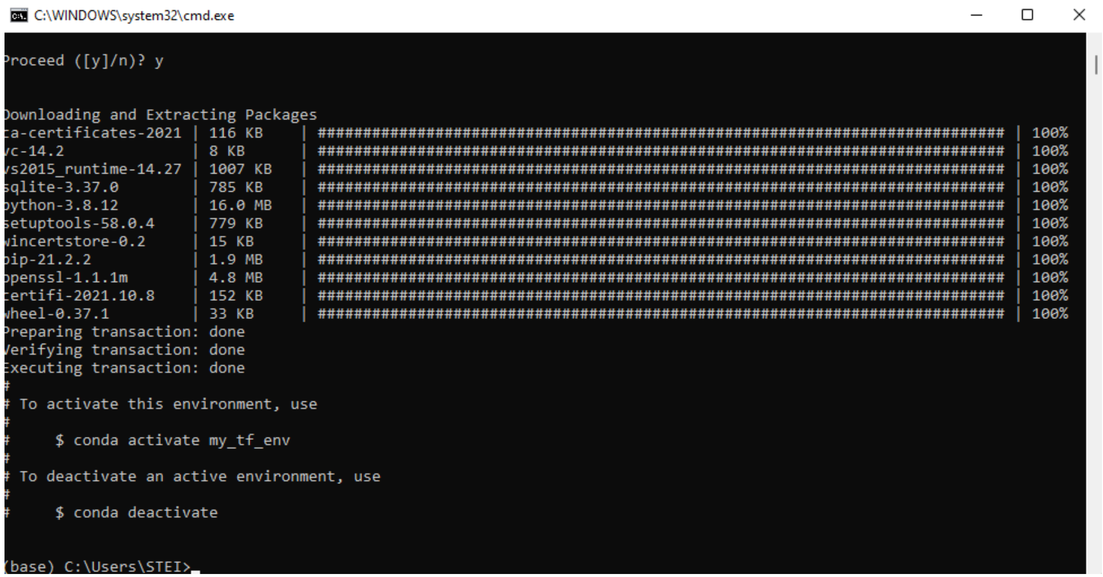
```
(base) C:\Users\STEI>conda create -n nama_environtment python=3.8
```

Notes: Creating the environment using yml file

- Type: ***conda env create –f name_of_env_files.yml***

3. Activating the (Virtual) Environment

- Use, Anaconda Prompt, or cmd, or Terminal (MacOs/ Linux only)
- Type: **conda activate** **nama_environment**
- To deactivate active environment: **conda deactivate**



```
Proceed ([y]/n)? y


Downloading and Extracting Packages
ca-certificates-2021 | 116 KB    | ############################################################################ | 100%
vc-14.2              | 8 KB      | ############################################################################ | 100%
vs2015_runtime-14.27 | 1007 KB   | ############################################################################ | 100%
sqlite-3.37.0        | 785 KB    | ############################################################################ | 100%
python-3.8.12        | 16.0 MB   | ############################################################################ | 100%
setuptools-58.0.4    | 779 KB    | ############################################################################ | 100%
wincertstore-0.2     | 15 KB     | ############################################################################ | 100%
pip-21.2.2           | 1.9 MB    | ############################################################################ | 100%
openssl-1.1.1m       | 4.8 MB    | ############################################################################ | 100%
certifi-2021.10.8    | 152 KB    | ############################################################################ | 100%
wheel-0.37.1         | 33 KB     | ############################################################################ | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate my_tf_env
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(base) C:\Users\STEI>_
```

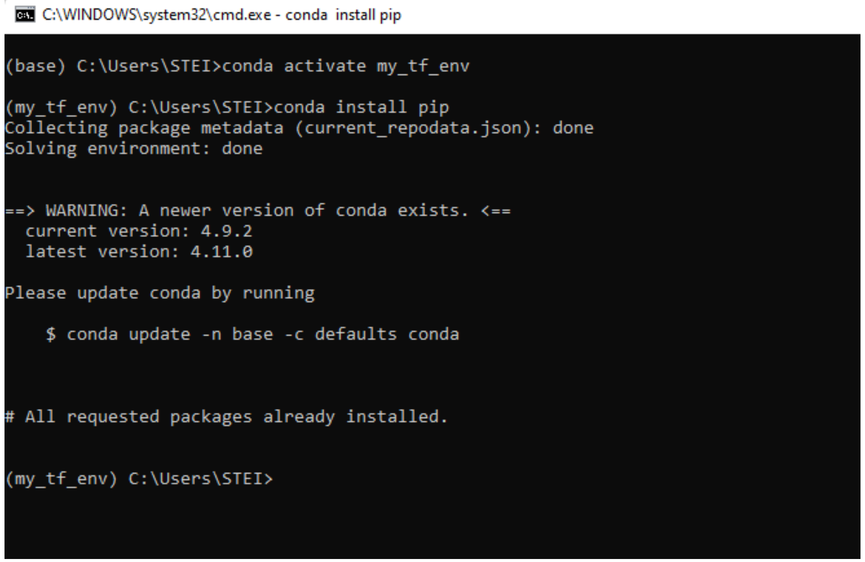
4. Install Required Libraries

4.1 Install pip installer

- From prompt, type: **conda install pip**

4.2 Install required libraries

- From prompt, type: **pip install library_name==version**
- Example: pip install jupyter==1.0.0



```
(base) C:\Users\STEI>conda activate my_tf_env

(my_tf_env) C:\Users\STEI>conda install pip
Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.9.2
  latest version: 4.11.0

Please update conda by running

    $ conda update -n base -c defaults conda


# All requested packages already installed.


(my_tf_env) C:\Users\STEI>
```

List of needed libraries:

- jupyter==1.0.0
- lxml==4.7.1
- matplotlib==3.5.1
- pandas==1.4.0
- Pillow==9.0.1
- scikit-learn==1.0.2
- numpy==1.22.2
- seaborn==0.11.2
- tensorflow==2.8.0

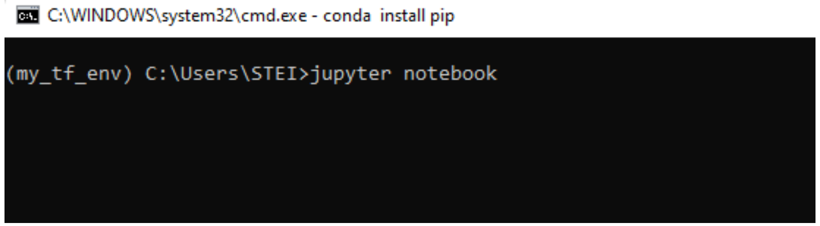
Note: library version may vary. Check the last update.

5. My First Jupyter Notebook Project

5.1 Run jupyter notebook

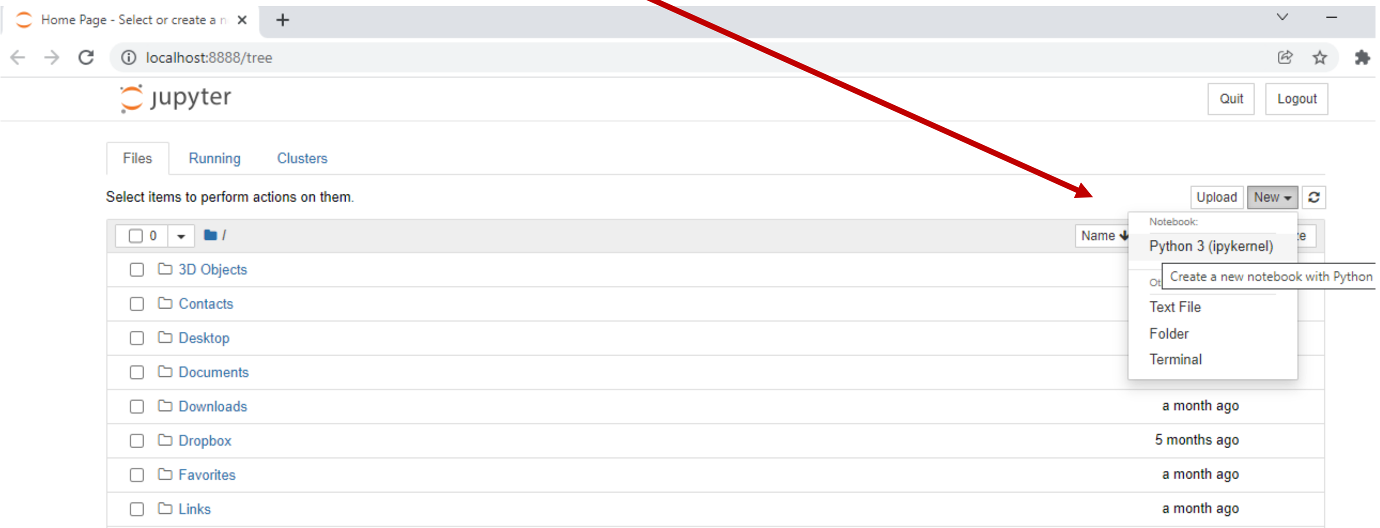
- From prompt, type: **jupyter notebook**

5.2 Create New Python Notebook



C:\WINDOWS\system32\cmd.exe - conda install pip

```
(my_tf_env) C:\Users\STEI>jupyter notebook
```

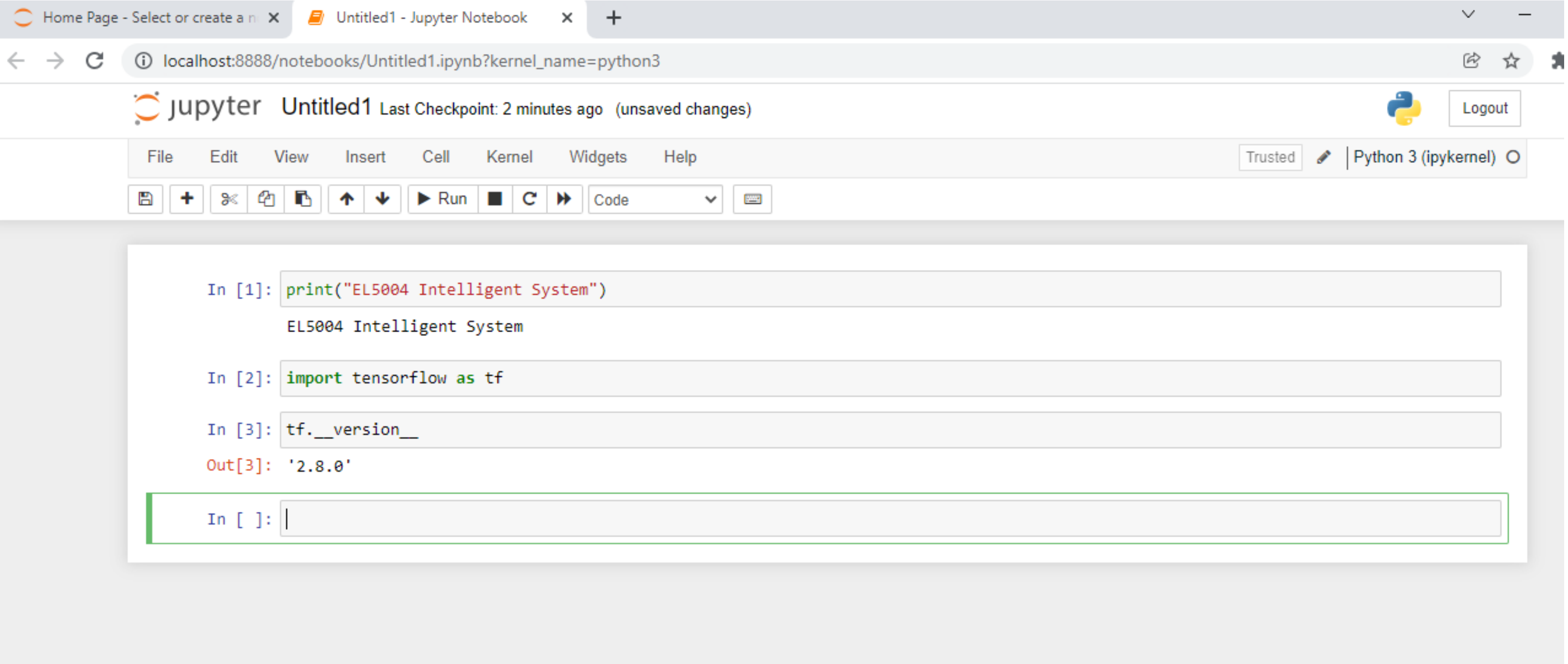


Select items to perform actions on them.

3D Objects	
Contacts	
Desktop	
Documents	
Downloads	a month ago
Dropbox	5 months ago
Favorites	a month ago
Links	a month ago

5. My First Jupyter Notebook Project

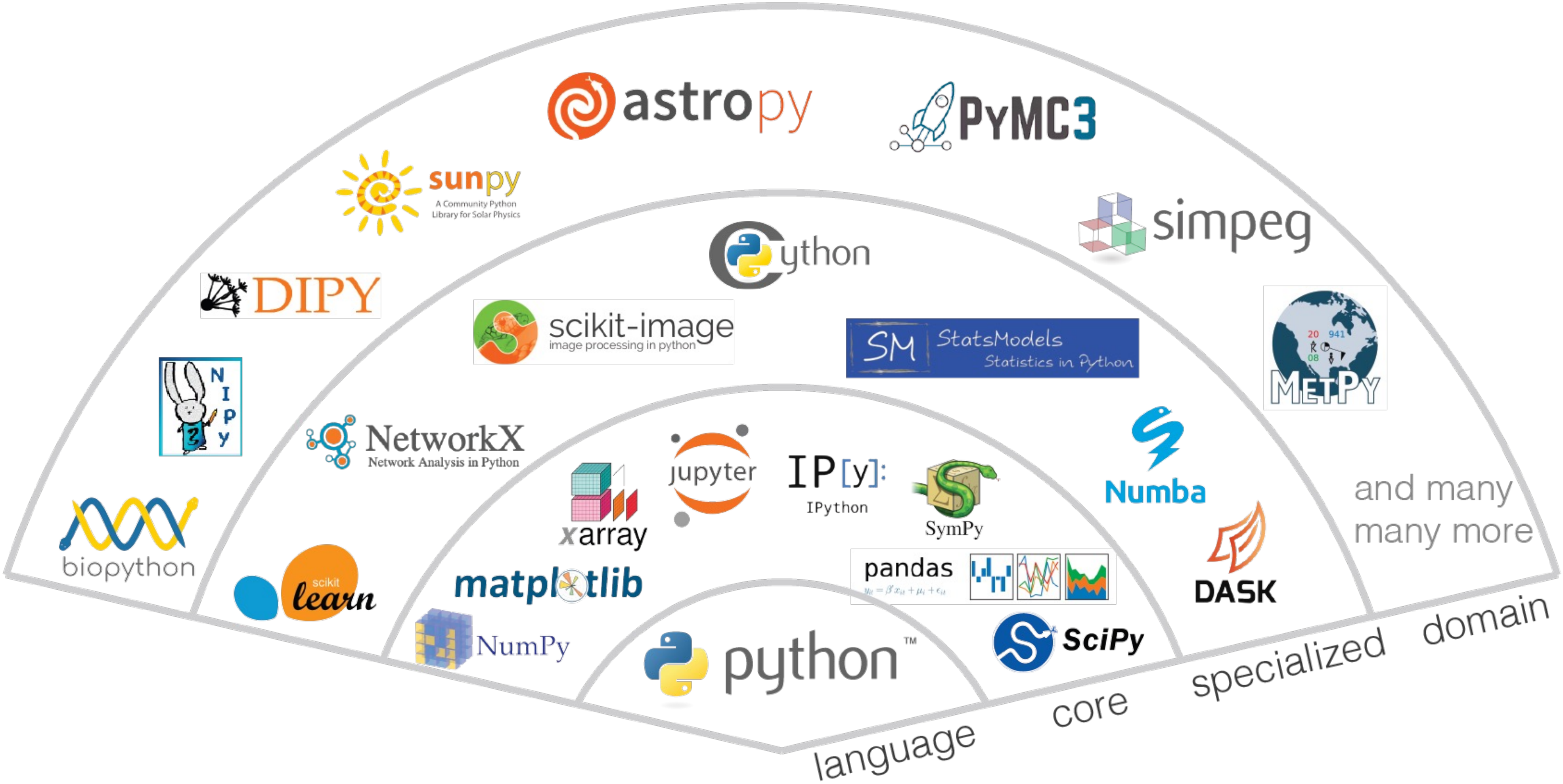
- Type the code, and press **Shift + Enter** to Run the code



Home Page - Select or create a n
Untitled1 - Jupyter Notebook
localhost:8888/notebooks/Untitled1.ipynb?kernel_name=python3
jupyter
Untitled1 Last Checkpoint: 2 minutes ago (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code

```
In [1]: print("EL5004 Intelligent System")

        EL5004 Intelligent System

In [2]: import tensorflow as tf

In [3]: tf.__version__

Out[3]: '2.8.0'

In [ ]: 
```

Python Ecosystem


astropy
PyMC3
sunpy
A Community Python Library for Solar Physics
simpeg
Cython
DIPY
scikit-image
image processing in python
SM
StatsModels
Statistics in Python
NIPY
MetPy
NetworkX
Network Analysis in Python
Numba
jupyter
IP[y]:
IPython
SymPy
xarray
and many
many more
biopython
scikit
learn
matplotlib
pandas
DASK
NumPy
python™
SciPy
language
core
specialized
domain