# Power Systems Resilience Analysis during Extreme Weather Software Implementation: Python Packages Installation Instruction

The resilience analysis of regional power systems under extreme events is prevalent due to the crippling social impacts and substantial economic losses caused by bulk power outages. The study of hurricane's impact on the power system infrastructure helps find the weak links in the system and helps in clarifying the efficient restoration and recovery procedures. To evaluate the performance of the power systems under hurricane events, a data generation engine is developed using Python programming language. Using the developed Python scripts, some necessary packages, software are needed. This Python package installation instruction aimed at providing guidelines to successfully running the scripts.

## Table Of Contents

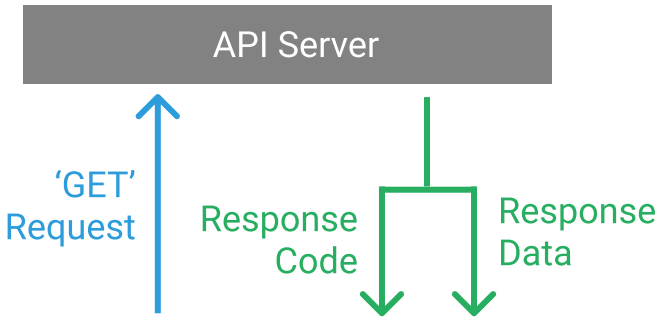


## Basic Python Programming Language Terminology

In this section, a few Python Programming language terminologies will be introduced to help understan the Python basics. The full Python programming language introduction can be found at Python Introduction.

- **Python version**: python.org will release different version of Python every year. Even though the main functionality is the same, each version of Python has slightly different functionalities. The Python version is important when some libraries (packages) only support specific versions of Python. If Python has been installed, the version of Python can be checked by:
  - Windows: Open PowerShell and type `python --version`
- Mac: Open Terminal and type `python --version`

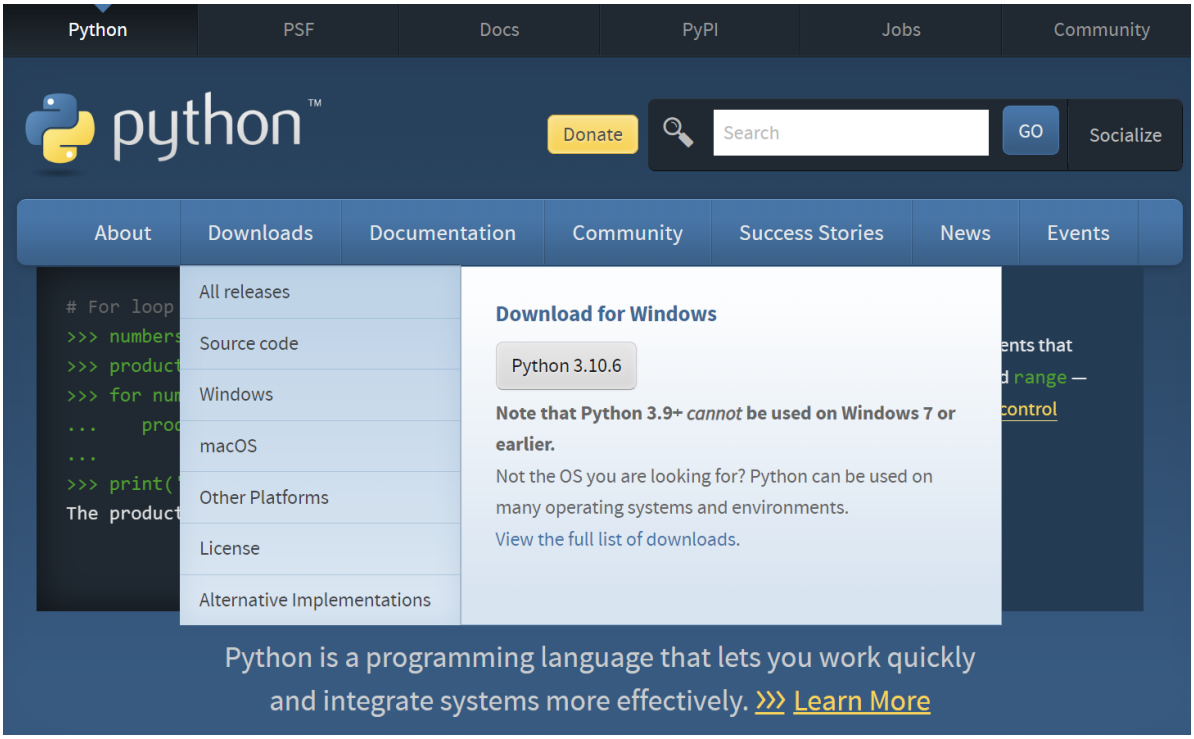
- **IDE**: IDE is short for Integrated Development Environment. Python built-in IDLE is useful for running simple codes. But using IDEs, larger and more complex programming projects can be easily handled. In the market, there are lots of free and proprietary IDEs, for example (Pycharm, Spider). All IDEs can help manage and organize the project, the preference is based on personal choice.
- **Virtual Environment**: Python virtual environment is a tool used for Python package management and project isolation. It allows Python packages to be installed locally in a separate directory for a particular project.
- **Packages**: A python packages consists of several modules. A module is a Python program that is a reusable code that serves particular purpose. In essence, Python packages contains a cluster of functions that can be used repeatedly.
- **API**: API is short for Application Programming Interface. It is a server that the user can use to retrieve and send data via code. The interaction between the code and API is illustrated in the following figure. The full Python API introduction can be found at Python API tutorial: Getting Started with APIs.




## Python Installation

This software supports Python version Python3.6 or above. If the users has installed Python version 3.6 or above, this section can be skipped.

- Go to Python.org to download the latest Python version.

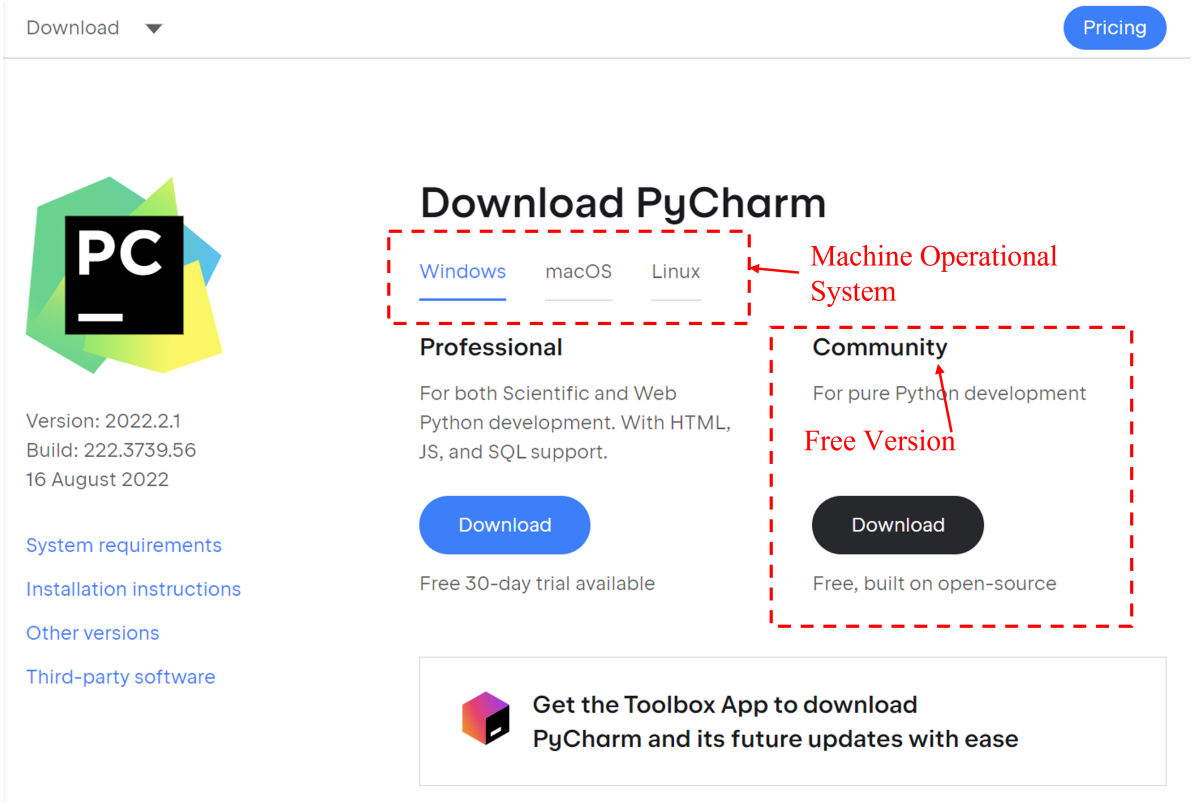



- Click the downloaded .exe for Windows or .dmg file for Mac and follow up the default settings.
- Check whether Python has been successfully installed:
  - **Windows**: Open PowerShell and type `python --version`
  - **Mac**: Open Terminal and type `python --version`

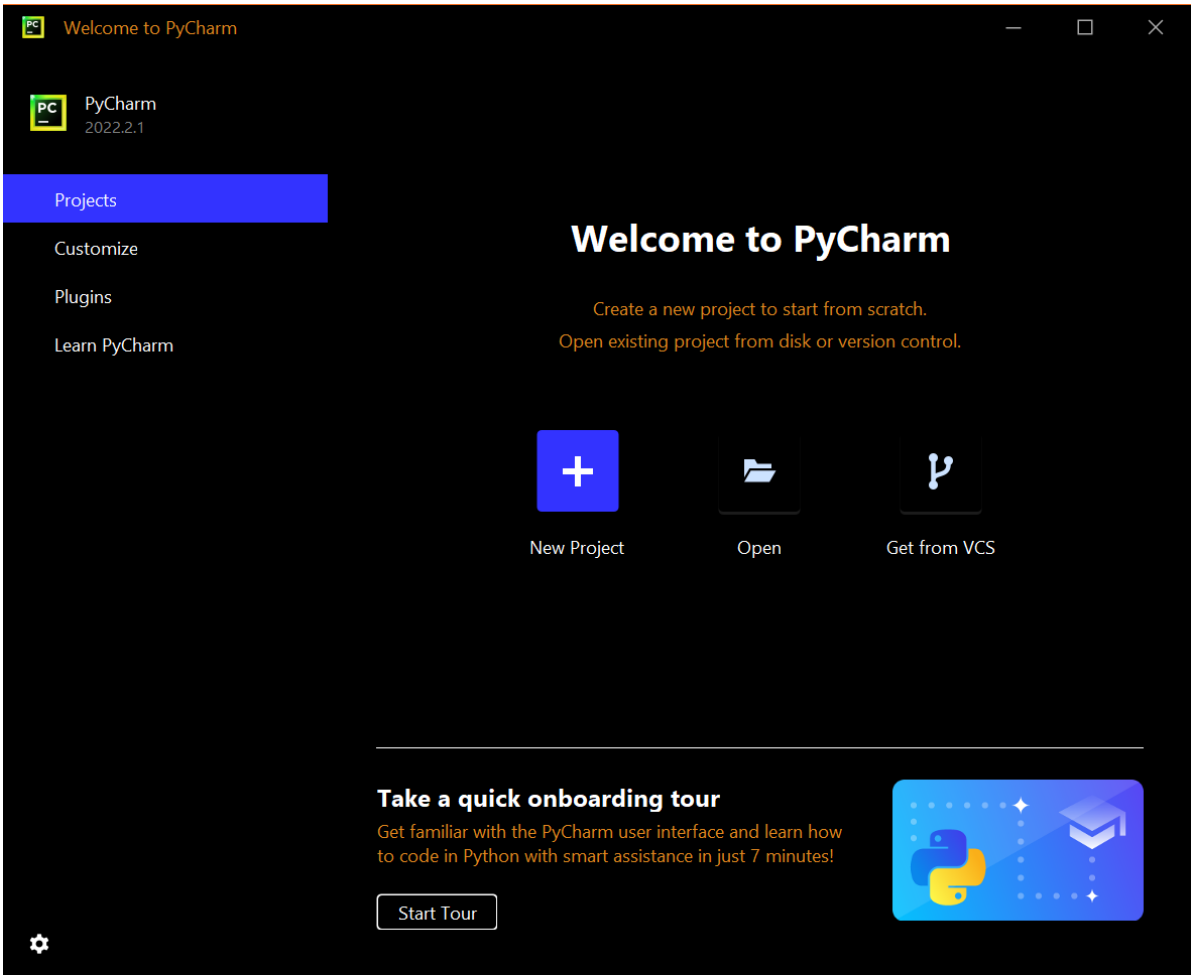
## Python IDEs

Python IDE is an editor that helps manage large, complex Python projects.  In the market, there are lots of free and charged IDEs, for example (Pycharm and Spider). In this instruction, Pycharm is used as a demonstration for the current and latter instructions. For other user-preferred IDEs, the user can visit their website to know the details about installation.

- Go to  Pycharm and click download button. The download button is on the upper-right in the website.
- In the downloading page, choose which operational system (Windows, Mac, Linux) the current machine is used and choose *Community* version.




- Check the installation. If Pycharm has been successfully installed, after opening it, Pycharm will look like:



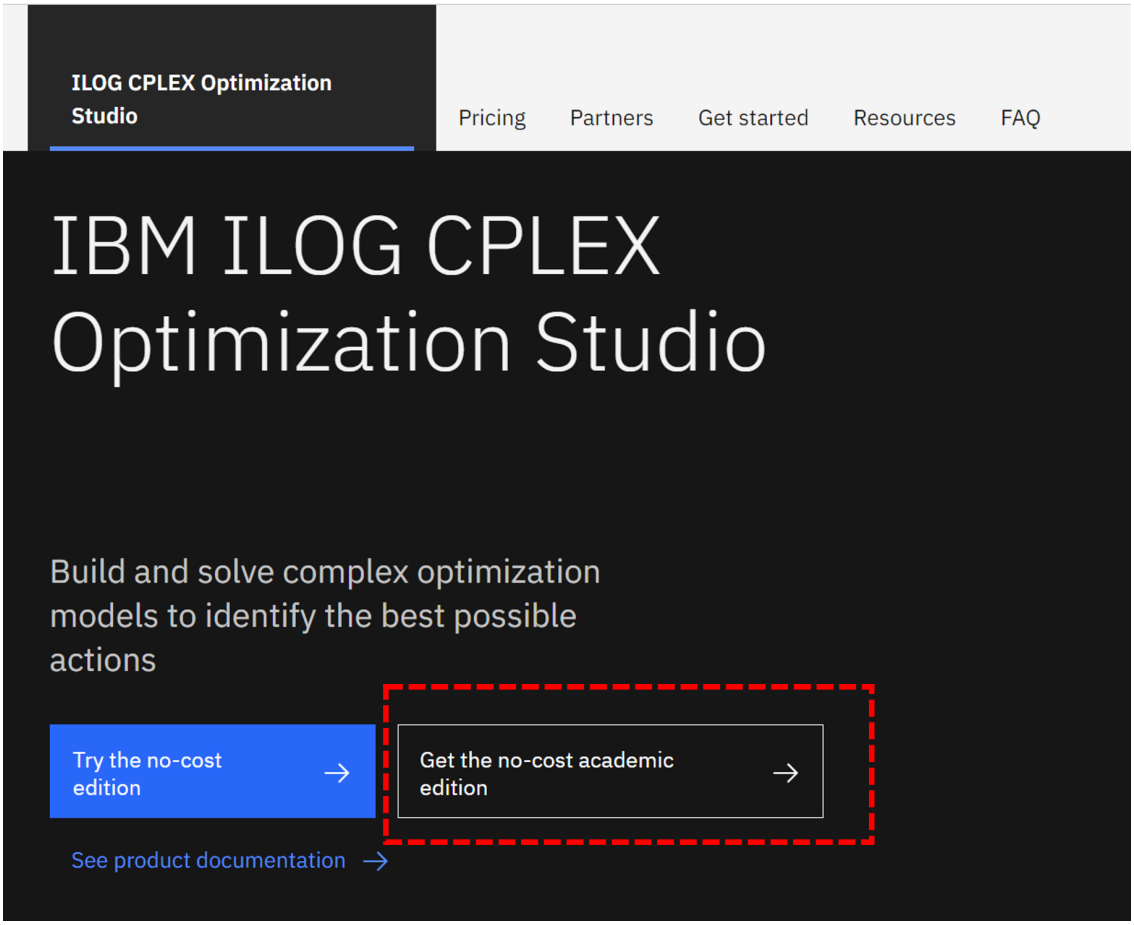

- **Install Anaconda**: In this software, the virtual environment will be created using Anaconda. Therefore, the users need to install Anaconda. The Anaconda downloading page can be found at Anaconda.

# IBM CPLEX Optimization software installation

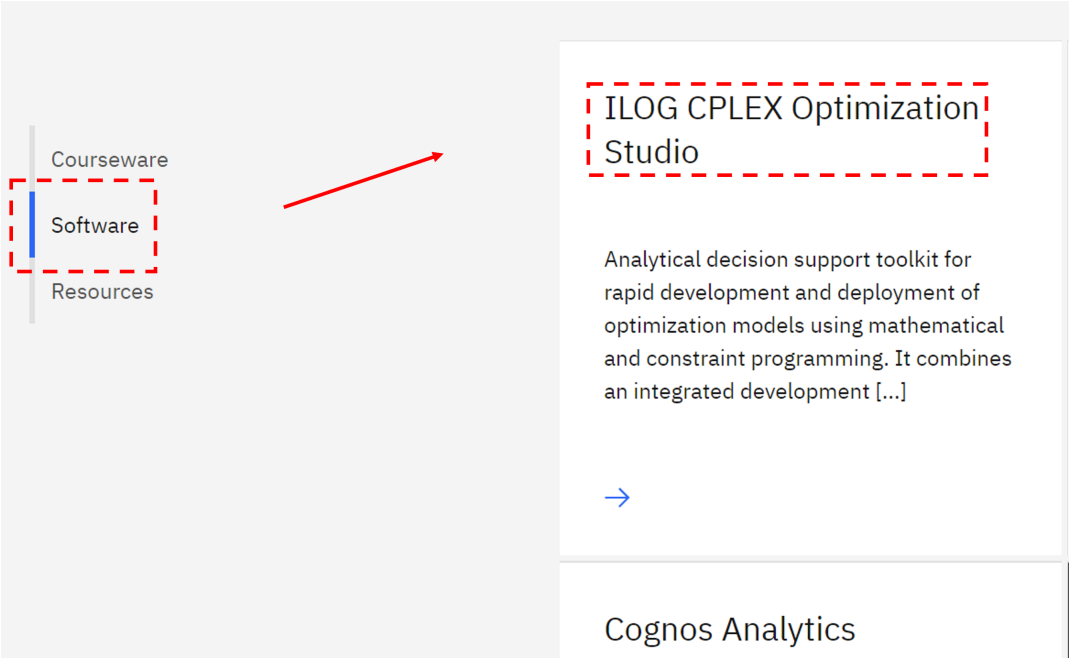
One objective of this project is to find the optimal dispatch strategy during hurricane events. Therefore, optimization package is needed to perform such tasks.  In this project, IBM CPLEX package is used. The installation of IBM CPLEX is as follow:

- **Java**: Check whether Java is installed in the computer by:
    - **Windows**: Open PowerShell and type `java -version`
    - **Mac**: Open Terminal and type `java -version`

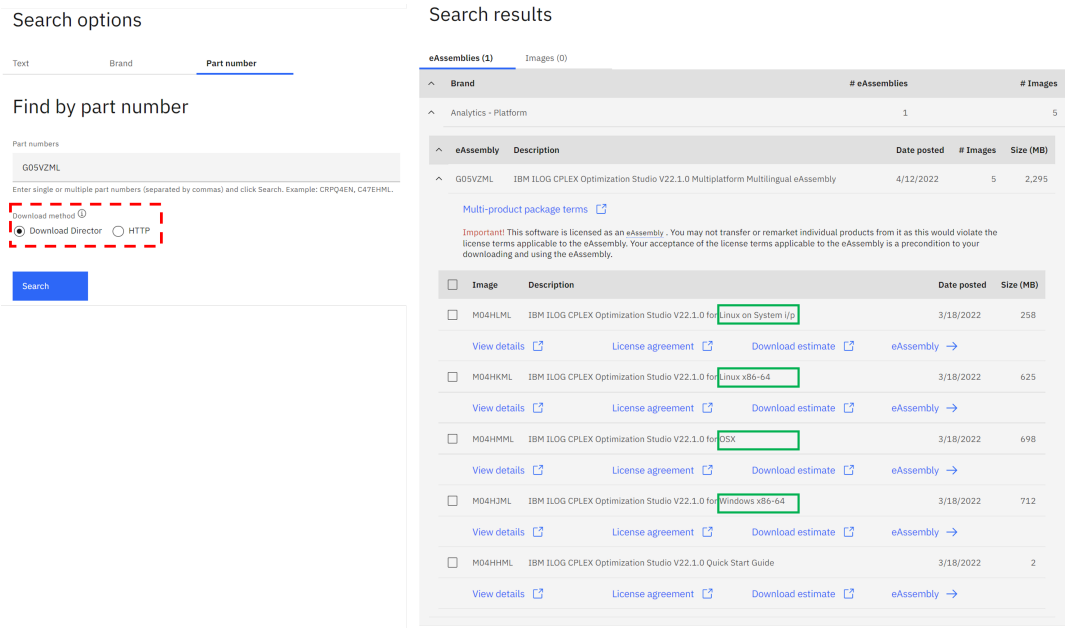
  If java is not installed, visit Java to download the latest version.
- **IBM CPLEX**: Go to IBM CPLEX Optimization Studio official website.
    - In the website, choose the no-cost academic edition as shown in the dashed-red box figure below.




    - In the `Data Science` page, click `Login` if you have IBM account. Otherwise click `Register`.
    - After login, scroll down to find `software` option and find `ILOG CPLEX Optimization Studio`.




- In the downloading page, choose `Download Director` and the version for current machine operational system.



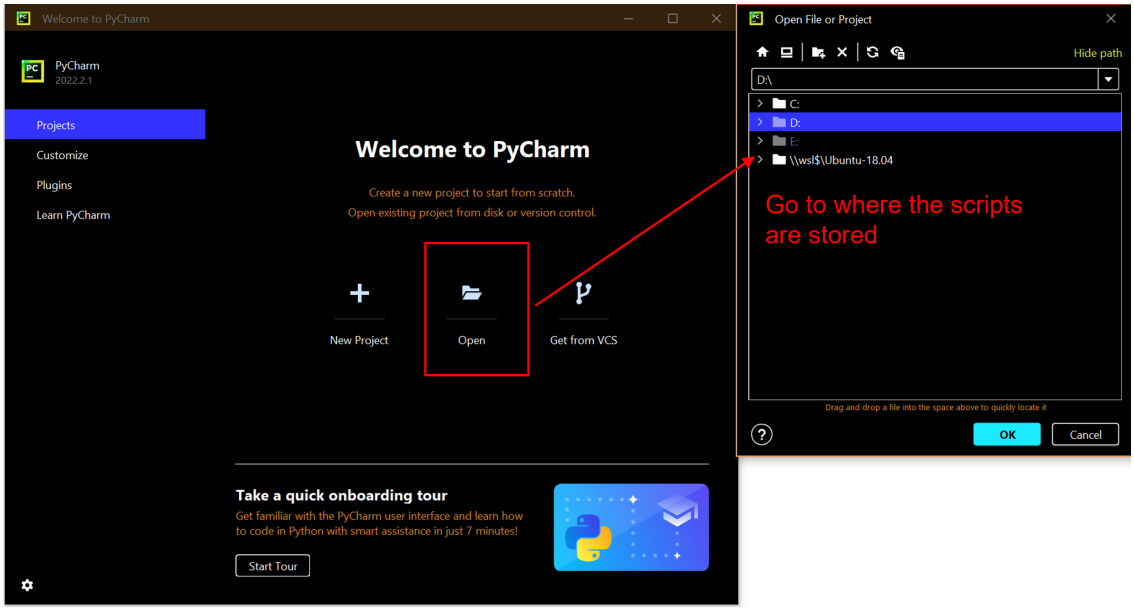

- After downloading the IBM CPLEX software, follow the default setting to install. If the user wants to install in non-default way, please remember the PATH for installed CPLEX directory.

# Setup IBM CPLEX Python API

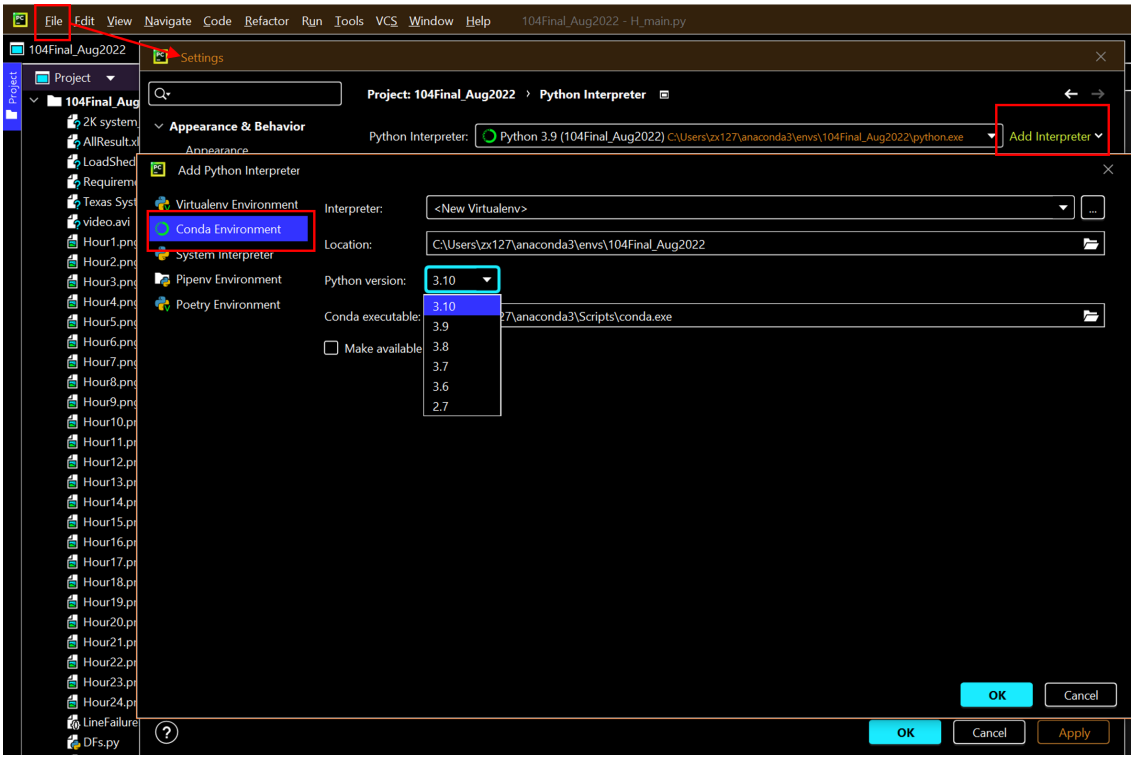
Even though IBM CPLEX has been installed, to use this optimizer in Python, the users need to set up CPLEX Python API so that Python can find and use CPLEX. To setup IBM CPLEX Python API, the instructions on Windows and Mac system will be illustrated separately.

To set up CPLEX Python API, the user need to set up the Python Virtual Environment First.

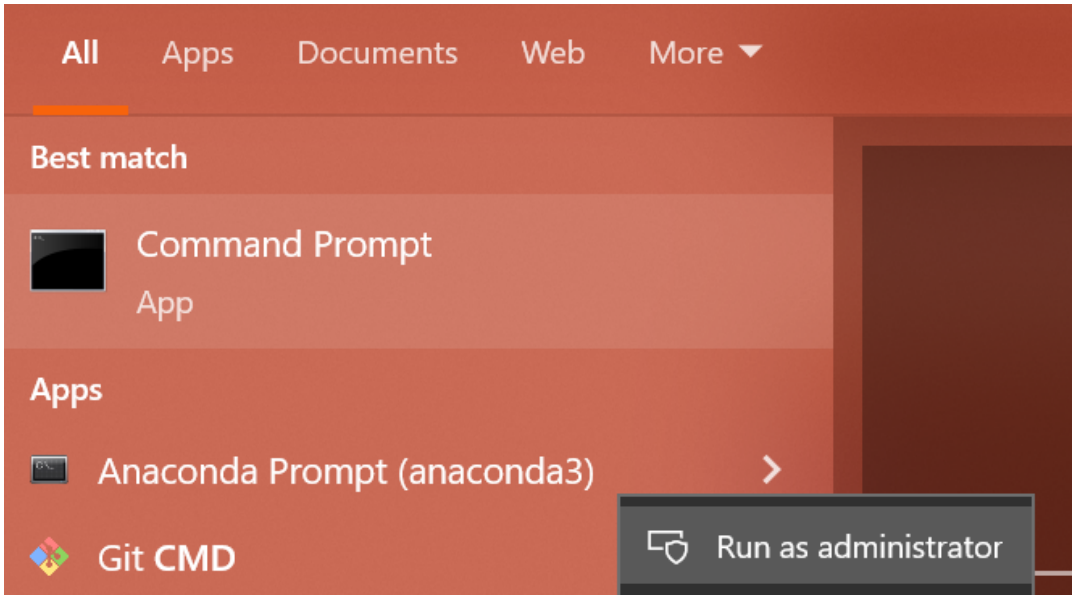
- Creating Virtual Environment
  - Open Pycharm → Open where the Python scripts are stored




- Open setting in Pycharm ribbon → Settings → Interpreter → Add local interpreter → `Conda Environment`. In the `Conda Environment` select the Python version no latter than 3.6 and click `OK`.



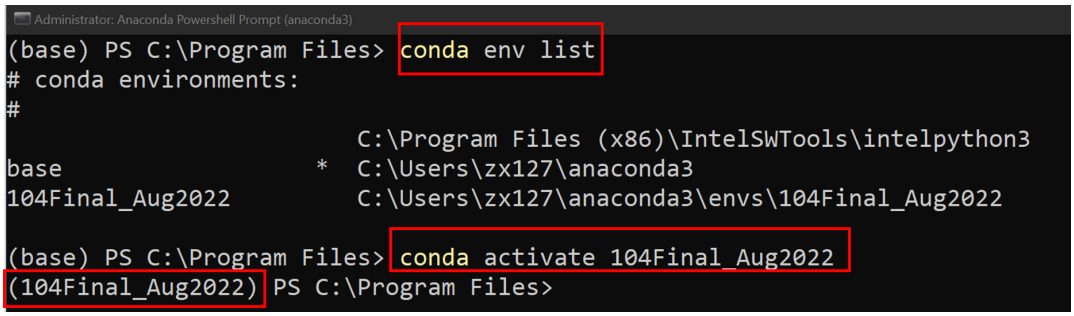

- Setup CPLEX Python API
  - Windows:
    - in the search bar type `cmd` → choose `Anaconda Prompt (anaconda3)` and run as an administrator




- In PowerShell type the following commands to activate the local virtual environment

```
conda env list
conda activate 'user created Virtual Enviornment name'
```

In the figure below, the `*` mean the current working Virtual Environment. If the current working Virtual Environment is base, change to user created virtual environment name. If the local virtual environment is activated, it should appear in the front of the director shown in the figure below.




- Setup CPLEX API

If user follows up default settings, the setup file should be located in `C:\Program Files\IBM\ILOG\CPLEX_Studio221\python`. In PowerShell type the following commands:

```
cd ..
dir
cd '.\Program Files\'
python IBM\ILOG\CPLEX_Studio221\python\setup.py install
```

Then the CPLEX API should be installed properly.

```
(104Final_Aug2022) PS C:\> cd ..
(104Final_Aug2022) PS C:\> dir


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        2020/12/28     20:23                Autodesk
d-----          2020/4/6     22:21                Dell
d-----          2020/4/7     21:08                ESD
d-----          2022/9/4     11:04                Intel
d-----          2020/4/9     14:38                Octave
d-----         2019/12/7      4:14                PerfLogs
d-----          2022/9/2     16:39                Program Files
d-r---          2022/9/1     18:30                Program Files (x86)
d-r---         2022/8/19      9:47                Users
d-----         2022/8/31     18:08                Windows


(104Final_Aug2022) PS C:\> cd '.\Program Files\'
(104Final_Aug2022) PS C:\Program Files> python IBM\ILOG\CPLEX_Studio221\python\setup.py install
```

- Mac
  - Open terminal and type

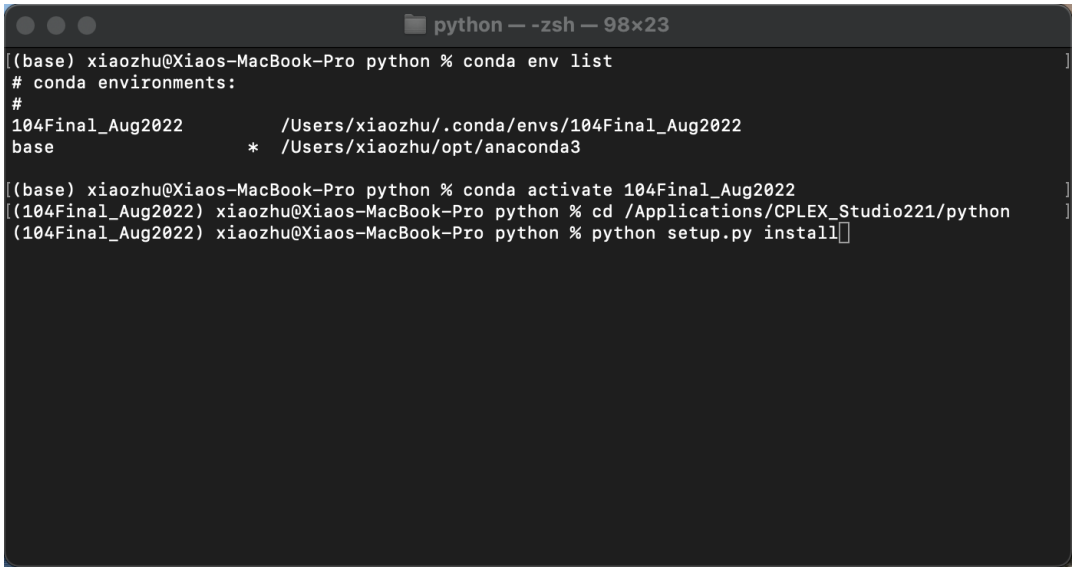
```
conda env list
conda activate 'user created Virtual Enviornment name'
```

  - Setup CPLEX API

If user follows up default settings, the setup file should be located in `/Application/CPLEX_Studio221/python`. In Terminal type the following commands:

```
cd `/Application/CPLEX_Studio221/python`
python setup.py install
```

Then the CPLEX API should be installed properly.




# Install Python Packages

After installing the IBM CPLEX API, the remaining Python packages can be installed automatically by running the following commands in the terminal:

```
pip install -r requirements.txt
```