Disclaimer: Most of the content of this assignment is adapted from a book "Python crash course. A hands-on, project-based. Introduction to programming" written by Eric Matthes. Some has been revised and updated as a development of Python Community in 2018 by Emma Nguyen, Pyladies – Ho Chi Minh Chapter.

Warm-up assignment

Estimated amount of time to complete this around 30 minutes. It is a very important for the next part of the workshop. If you encounter any issue, please reach out to any instructor around for assistance.

What Python? Why Python?

Python is a high-level programming language which allows developers to focus on designing of program by eliminating other unnecessary factors in a process such as data structure. It is simple and easy-to-learn for everyone.

Before we talk about why Python becomes more and more popular. Let us talk a little about an open source movement. So what does open source software mean? It is available to everyone(most of the time, along with a license stating about a copyright to study, change and distribute under a specific circumstance).

At the beginning, Python is identified as an open-source language for any kind of use, thus people love it. Communities of Python keep growing in many years at various aspect of engineering, technology and science. For example, Scikit-learn, Tensorflow, Pytorch in Machine learning and Artificial Intelligence in general.

Set-up environment with Ananconda with Python 3.5

There are several projects of Python around communities, and many of them are built on top of the others. To manage version of a package or a library is quite painful tasks for developers and Python users. Annaconda is considered as one of the best solutions for this problem by offering a safe developing virtual environment and simple installation for any kind of project you want to try.

Follow this to set up your Annancoda https://www.pugetsystems.com/labs/hpc/How-to-Install-Anaconda-Python-and-First-Steps-for-Linux-and-Windows-917/

Running Python Programs from a Jupyter Lab

In this workshop, we use Jupyter lab as our first choice for interacting with Python.

Please install Jupyter by following this link https://github.com/jupyterlab/jupyterlab

Try Python in your Operating System

Your first program: "Hello world!"

Let us try to print a phrase "Hello world!" by the following example:

```python
print("Hello Python world !")
```

```

Another way to do this is naming a variable and setting its value equal to “Hello Python world !” like this

```python
message = “Hello Python world !”
print(message)
```

Some more complicating ways for us to customize a message by using `format` syntax. Nonetheless, we will learn more later on with PEP8

PEP8 – A style guide for Python Code

Python is very easy to write, however, because of its convenience, people tend to “invent” their own style to structure their source code. In a group, it causes a lot of annoying problem to standardize and maintenance code base. For that reason, we have some normalized styles. One of the most common one is PEP 8 which has been applied pervasively in a lot of engineering team around the world. We encourage you applying this when writing your code, but still your choice.

Read this tutorial https://www.datacamp.com/community/tutorials/pep8-tutorial-python-code to help you write a beautiful code.

Git and Github

Along side with other set-up for Python, we would like to introduce two other crucial part of our learning workshop. The first is Git, which is a version control management system allowing us have many versions of source code in one folder, (Repository, or repo). In addition, Github is a website to upload, store and share code online.

Here is a step-by-step guide to walk you through to have your own account with Github https://guides.github.com/features/pages/
```