

# *Coding Practice Setup Guide*

(부제 : RL 실습 셋업 가이드)



*JuSung Kang, LiberVance Co. Ltd.,*  
[jskang.libervance@gmail.com](mailto:jskang.libervance@gmail.com),

# Coding Practice Setup Guide

- **Requirements**

- **OS**

- Windows

- **Packages**

- Anaconda for Python
    - Pytorch
    - Pycharm
    - OpenAI Gymnasium



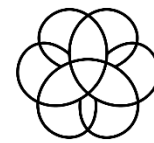
ANACONDA®



PyTorch



PyCharm

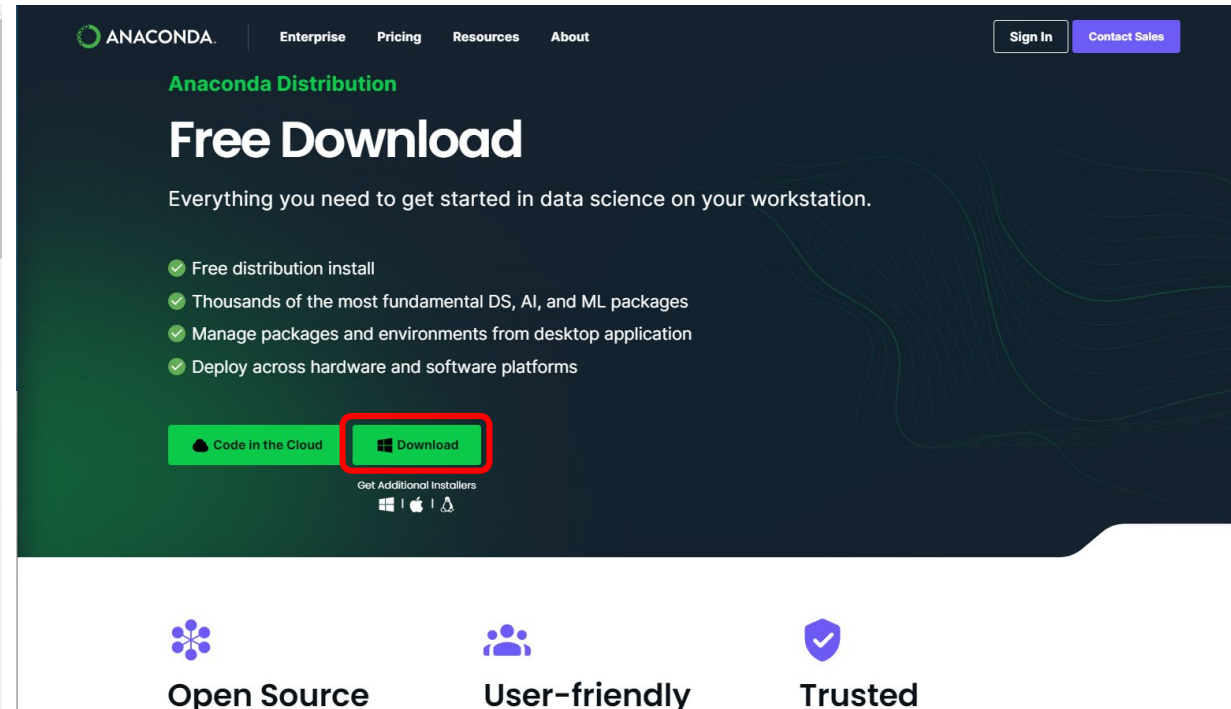
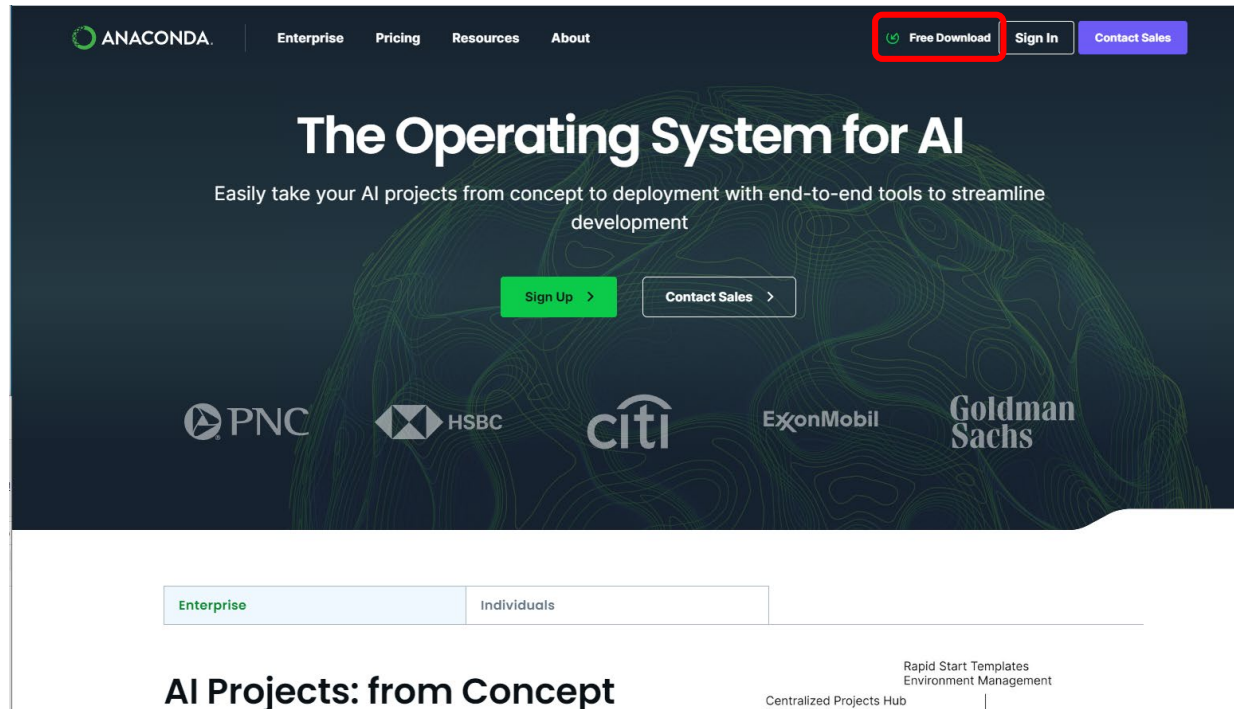


Gymnasium

# Coding Practice Setup Guide - Anaconda

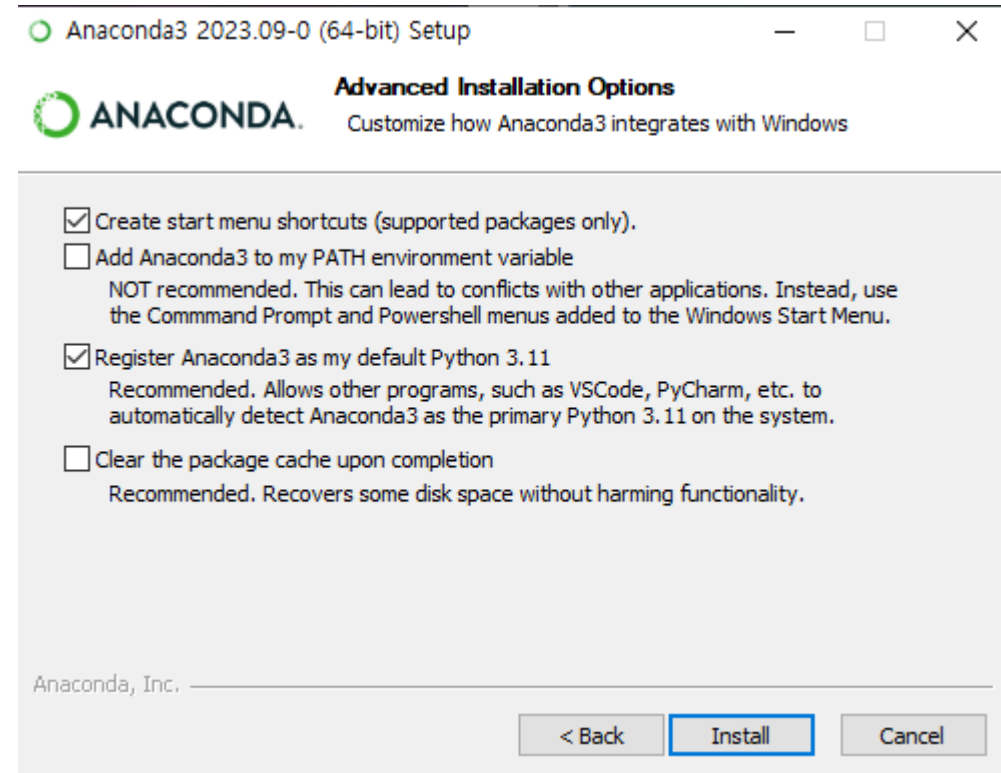
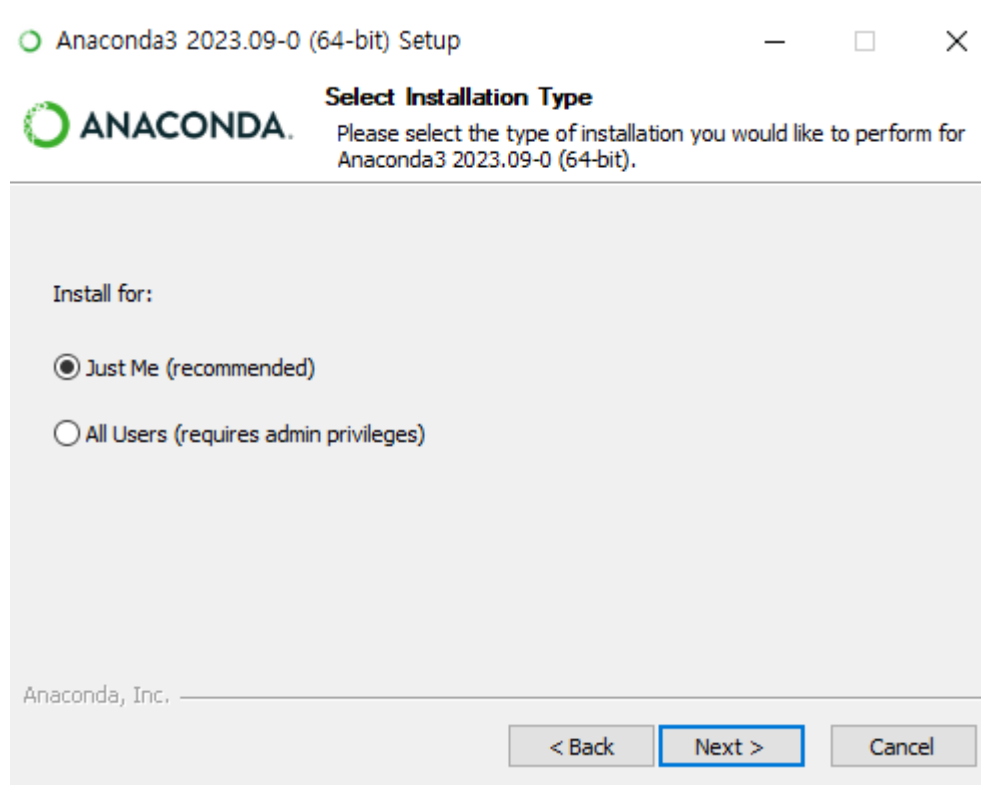
- Anaconda

- <https://www.anaconda.com/>
- Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe



# Coding Practice Setup Guide - Anaconda

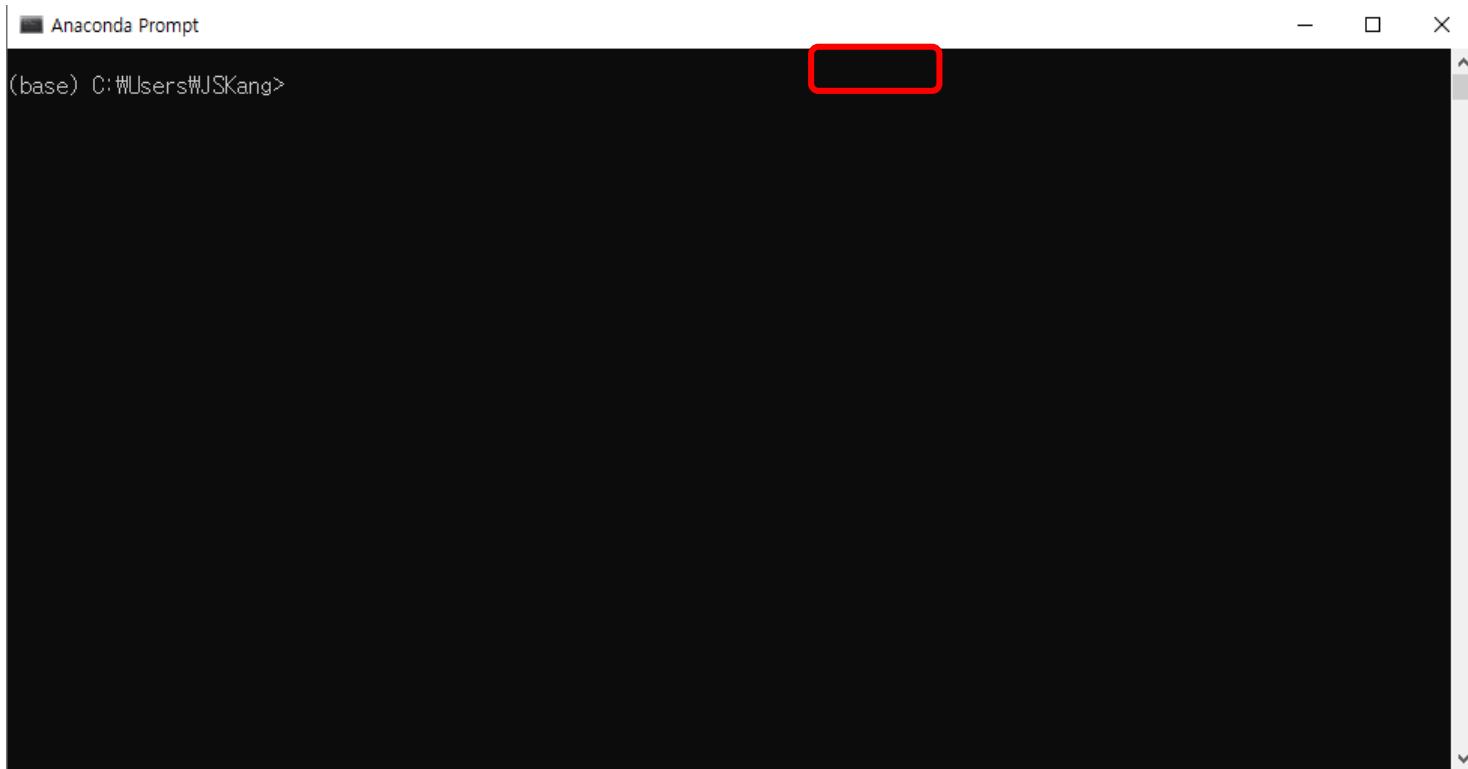
- Anaconda
  - <https://www.anaconda.com/>
  - Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe



# *Coding Practice Setup Guide - Anaconda*

- **Anaconda**

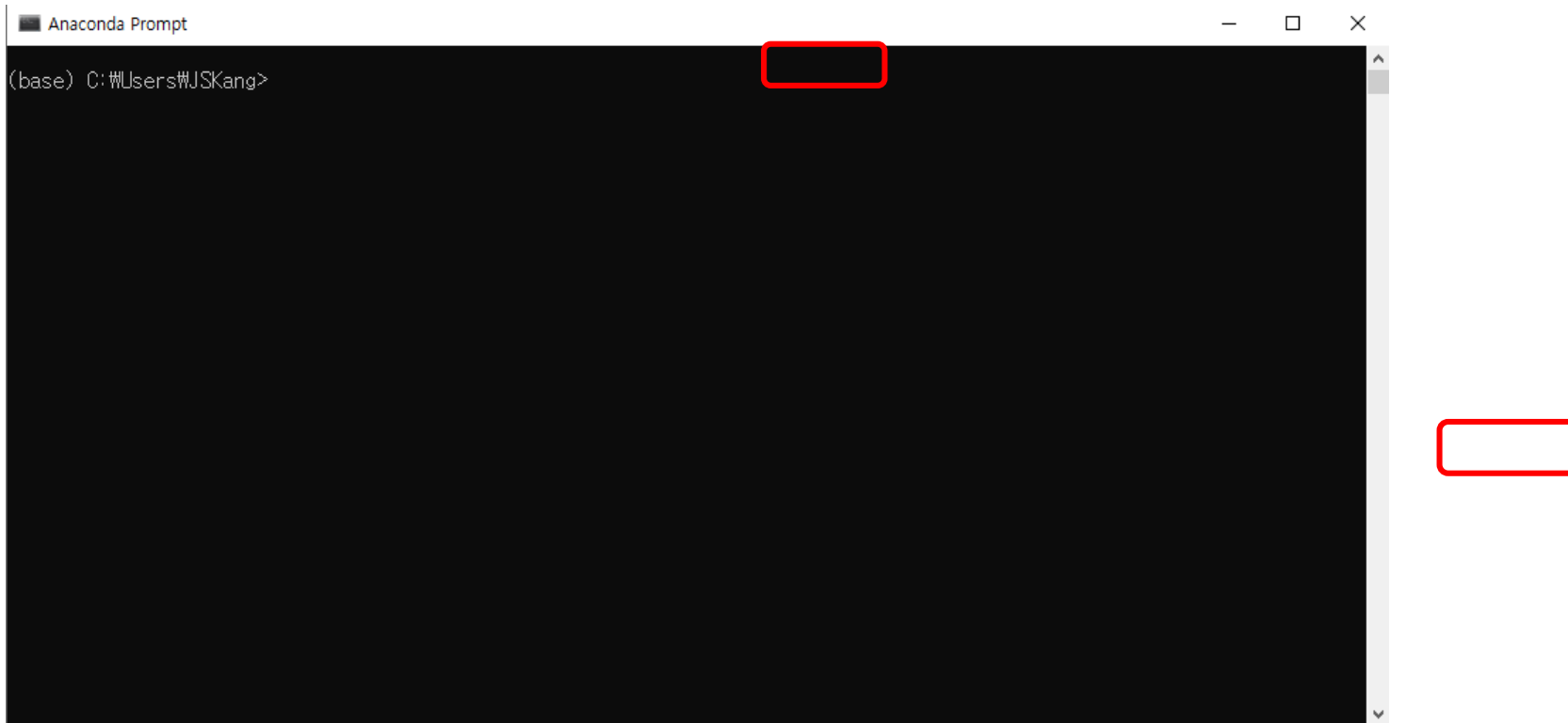
- <https://www.anaconda.com/>
- **Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe**



# Coding Practice Setup Guide - Pytorch

- **Anaconda**

- <https://www.anaconda.com/>
- **Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe**



# *Coding Practice Setup Guide – OpenAI Gym*

- **Anaconda**
  - <https://www.anaconda.com/>
  - Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe



# *Coding Practice Setup Guide – Anaconda (opt)*

- **Anaconda**
  - <https://www.anaconda.com/>
  - Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86\_64.exe





# Coding Practice Setup Guide – Anaconda (test)

- Anaconda

- [ht](#)

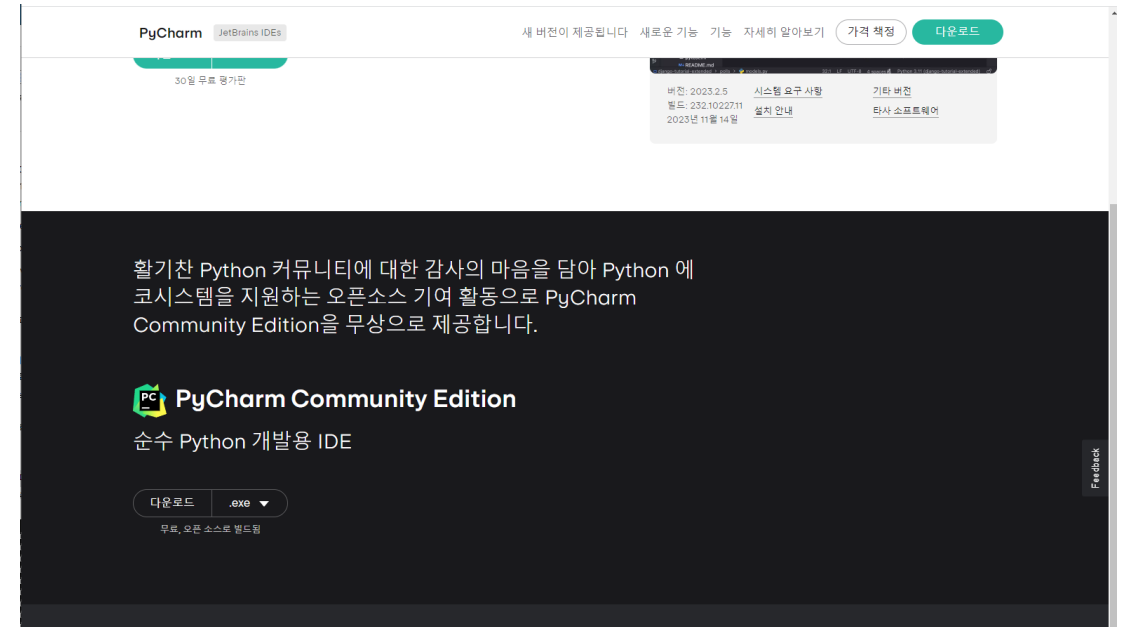
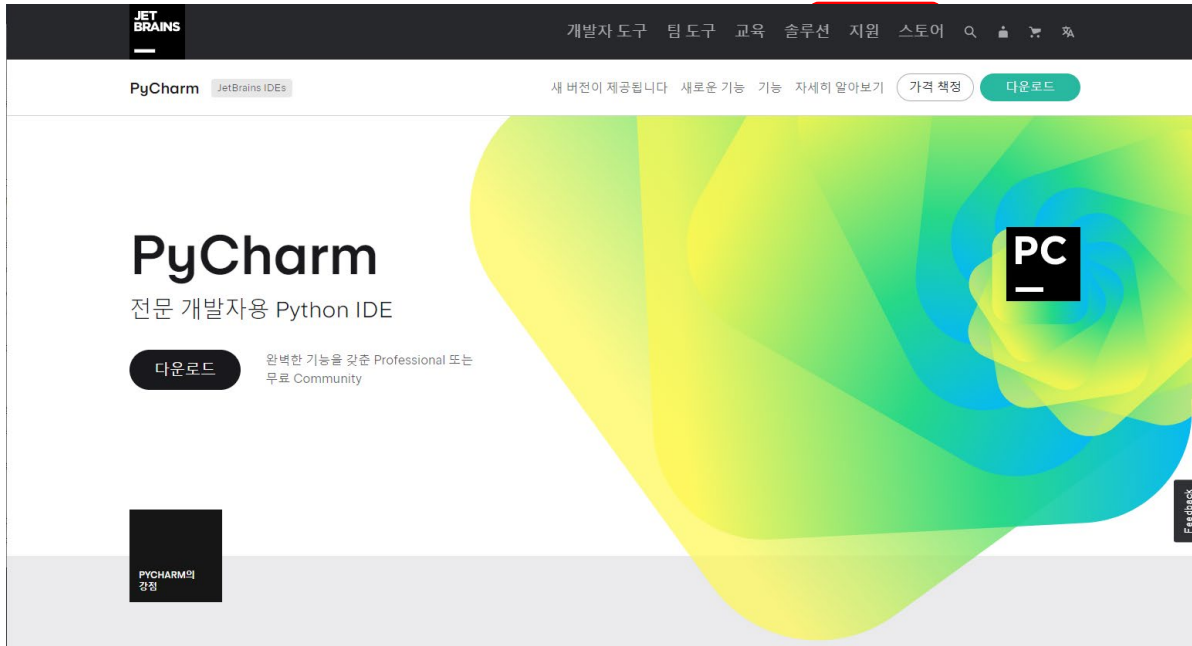
- Fr

```
Anaconda Prompt - conda deactivate - conda install -c conda-forge matplotlib - python
(MNU_RL_Demo) C:\Users\WJSKang>python
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> import gym, gymnasium
>>> import matplotlib
>>> import gym_super_mario_bros
>>> x = torch.rand(5,3)
>>> print(x)
tensor([[0.9932, 0.2308, 0.8958],
        [0.4067, 0.7606, 0.4997],
        [0.3011, 0.3439, 0.8837],
        [0.0985, 0.9594, 0.2512],
        [0.0599, 0.0521, 0.4037]])
>>> _
```

# Coding Practice Setup Guide – Pycharm

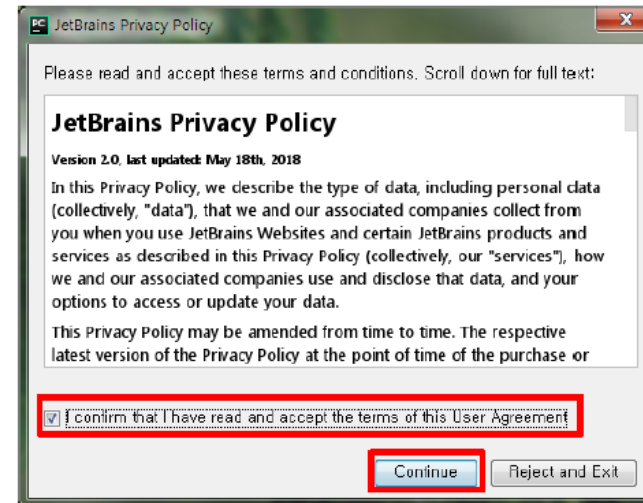
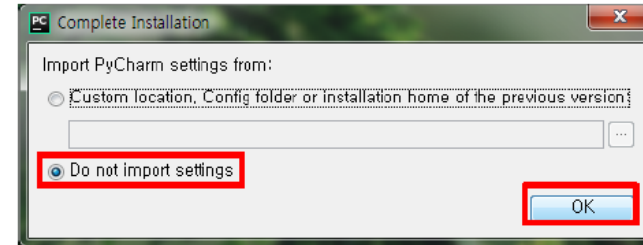
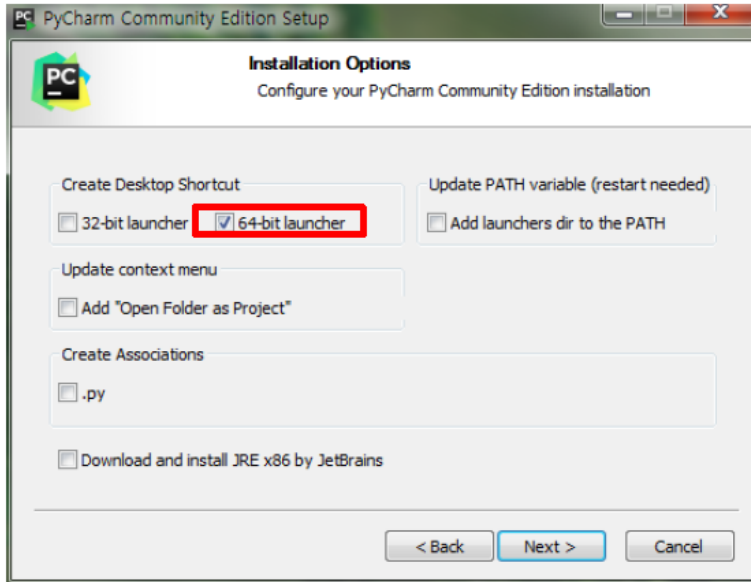
- Anaconda

- <https://www.jetbrains.com/ko-kr/pycharm/>
- 다운로드 > Pycharm Community Edition (다운로드) > (files) pycharm-community-2023.2.5.exe



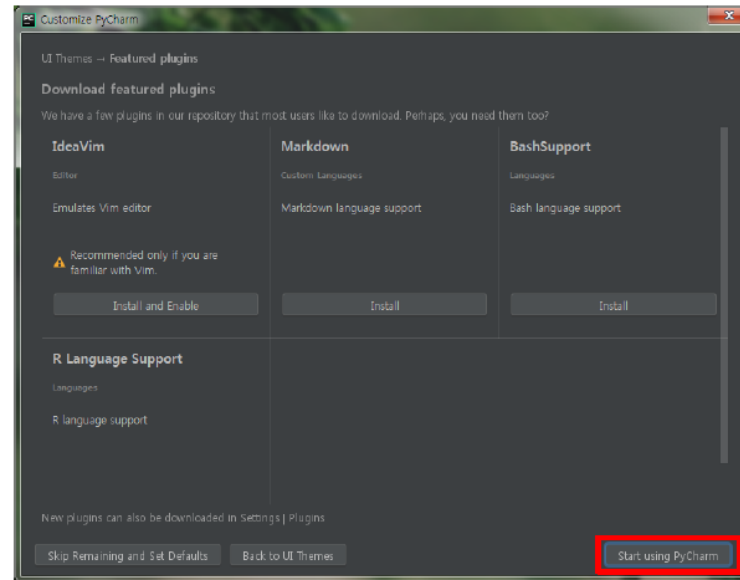
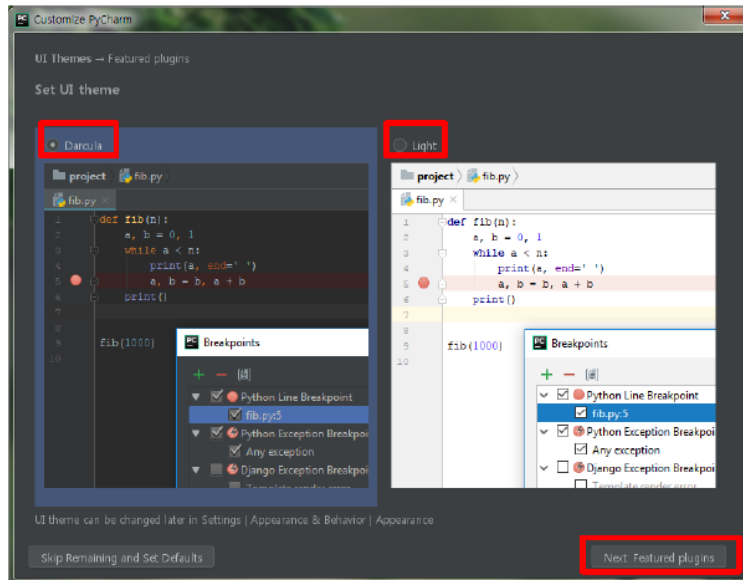
# Coding Practice Setup Guide – Pycharm

- Anaconda
  - <https://www.jetbrains.com/ko-kr/pycharm/>
  - 다운로드 • Install PyCharm



# Coding Practice Setup Guide – Pycharm

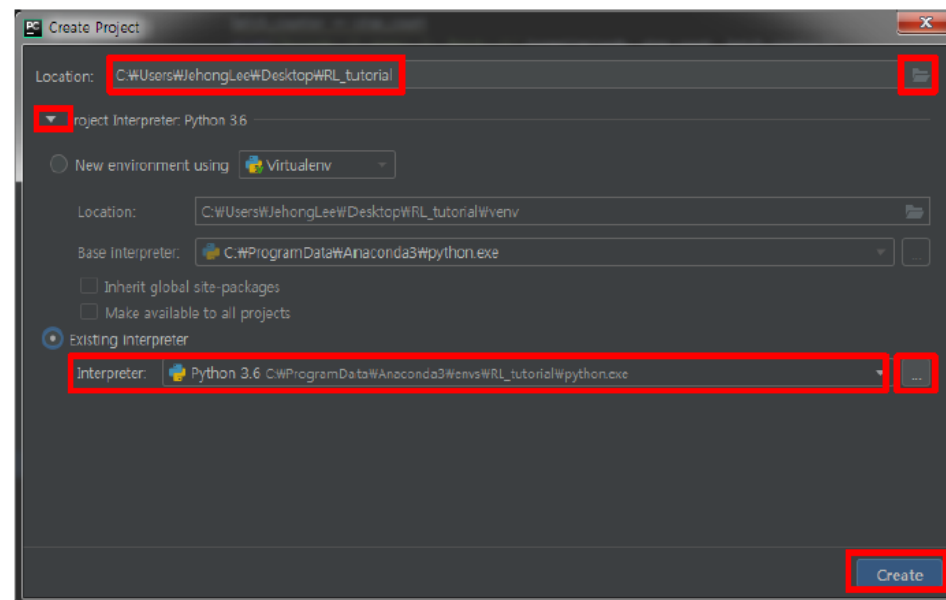
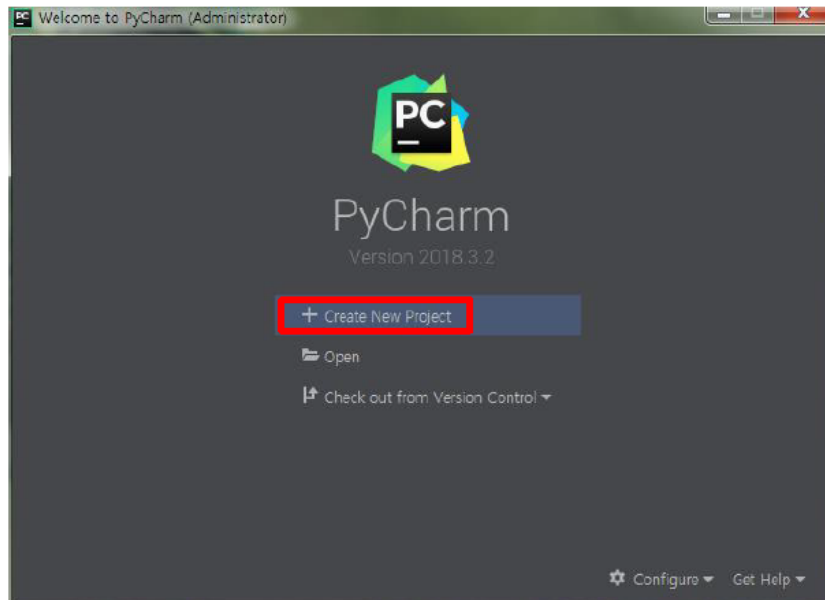
- Anaconda
  - <https://www.jetbrains.com/ko-kr/pycharm/>
  - 다운로드 > Pycharm Community Edition (다운로드) > Pycharm Setting



# Coding Practice Setup Guide – Pycharm

- Anaconda

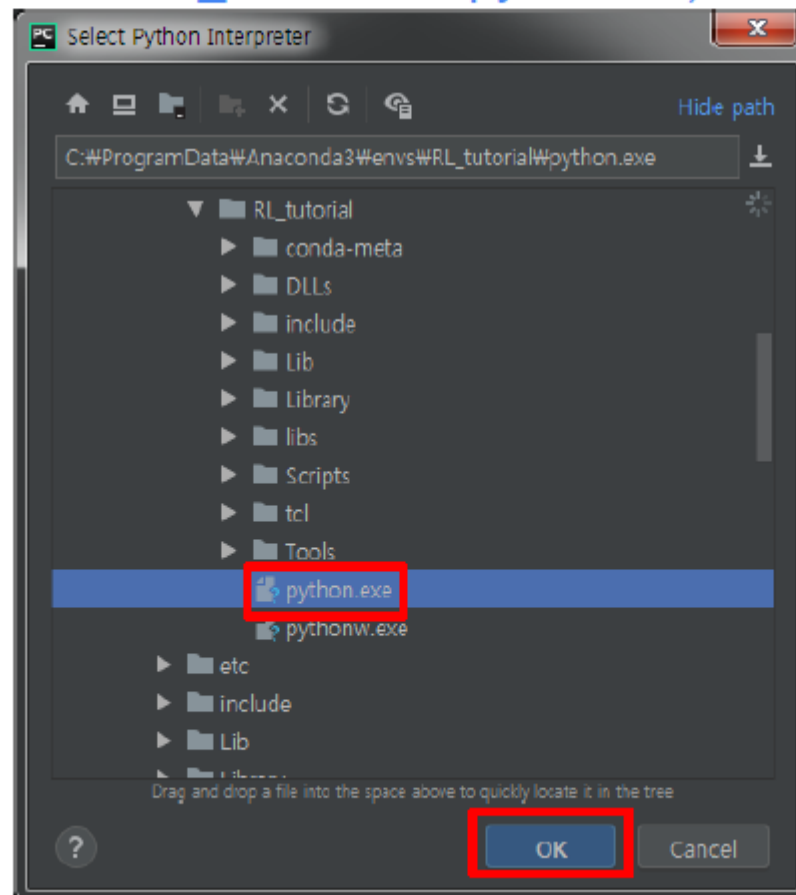
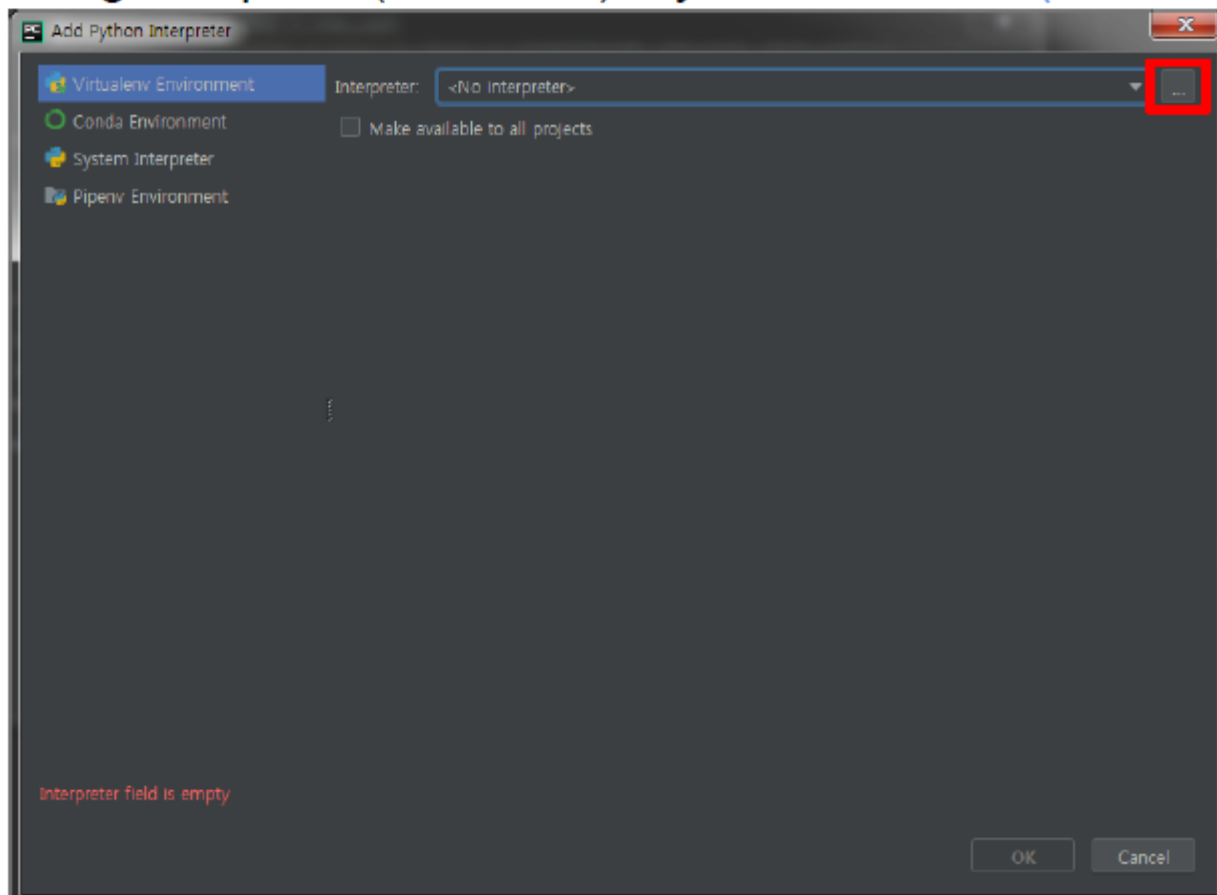
- <https://www.jetbrains.com/ko-kr/pycharm/>
- 다운로드 > Pycharm Community Edition (다운로드) >
  - Pycharm Setting
    - Create New Project
    - Location: 프로젝트 파일 저장 (PycharmProjects/RL\_tutorial)
    - Existing Interpreter(Windows 10): Python 설치 경로 ([user/Anaconda3/envs/RL\\_tutorial/python.exe](#))
    - Existing Interpreter(Mac OSX): Python 설치 경로 ([anaconda3/envs/RL\\_tutorial/bin/python3.6](#))



# PyCharm(cont')

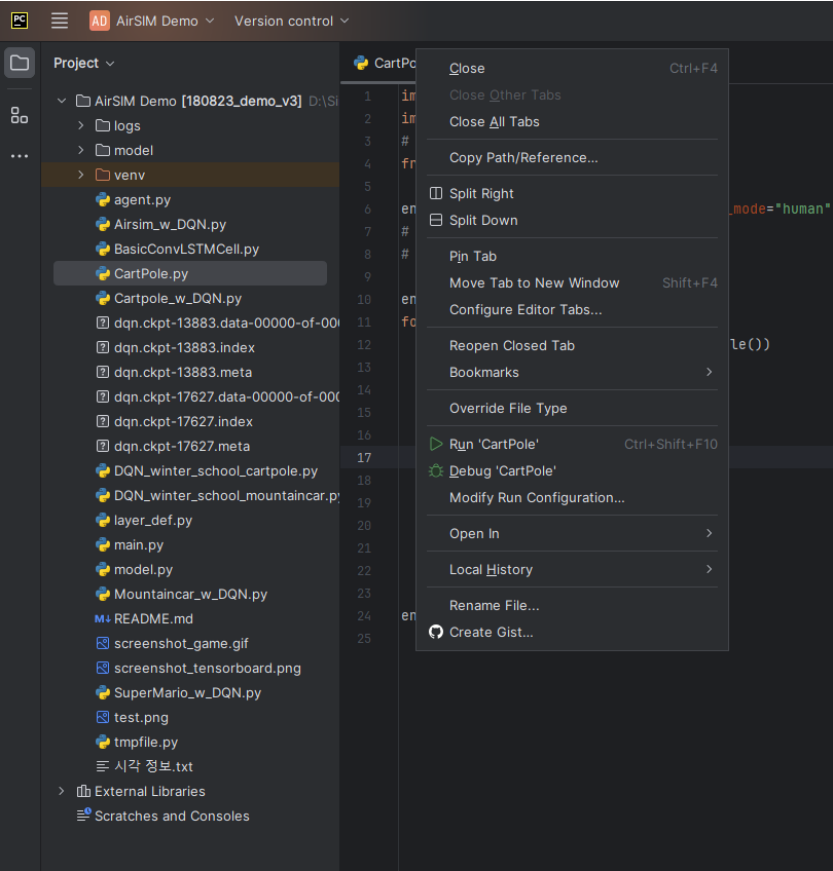
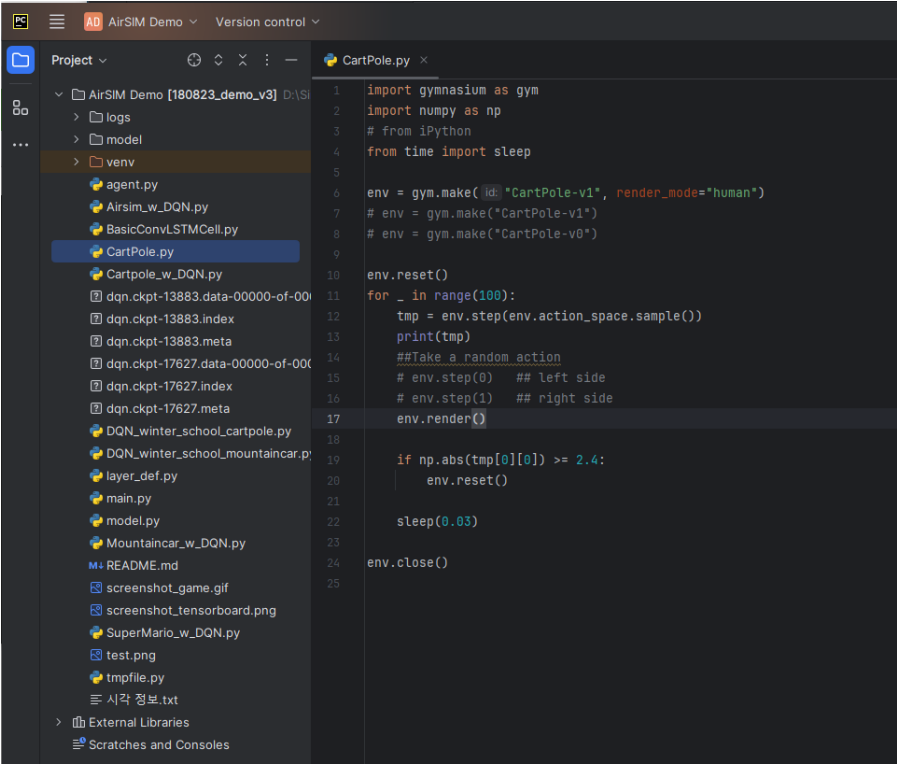
## • Pycharm Setting

- Add python Interpreter
- Select Python Interpreter
- Existing Interpreter(Windows 7): Python 설치 경로 (ProgramData/Anaconda3/envs/RL\_tutorial/python.exe)
- Existing Interpreter(Windows 10): Python 설치 경로 (user/Anaconda3/envs/RL\_tutorial/python.exe)
- Existing Interpreter(Mac OSX): Python 설치 경로 (anaconda3/envs/RL\_tutorial/bin/python3.6)



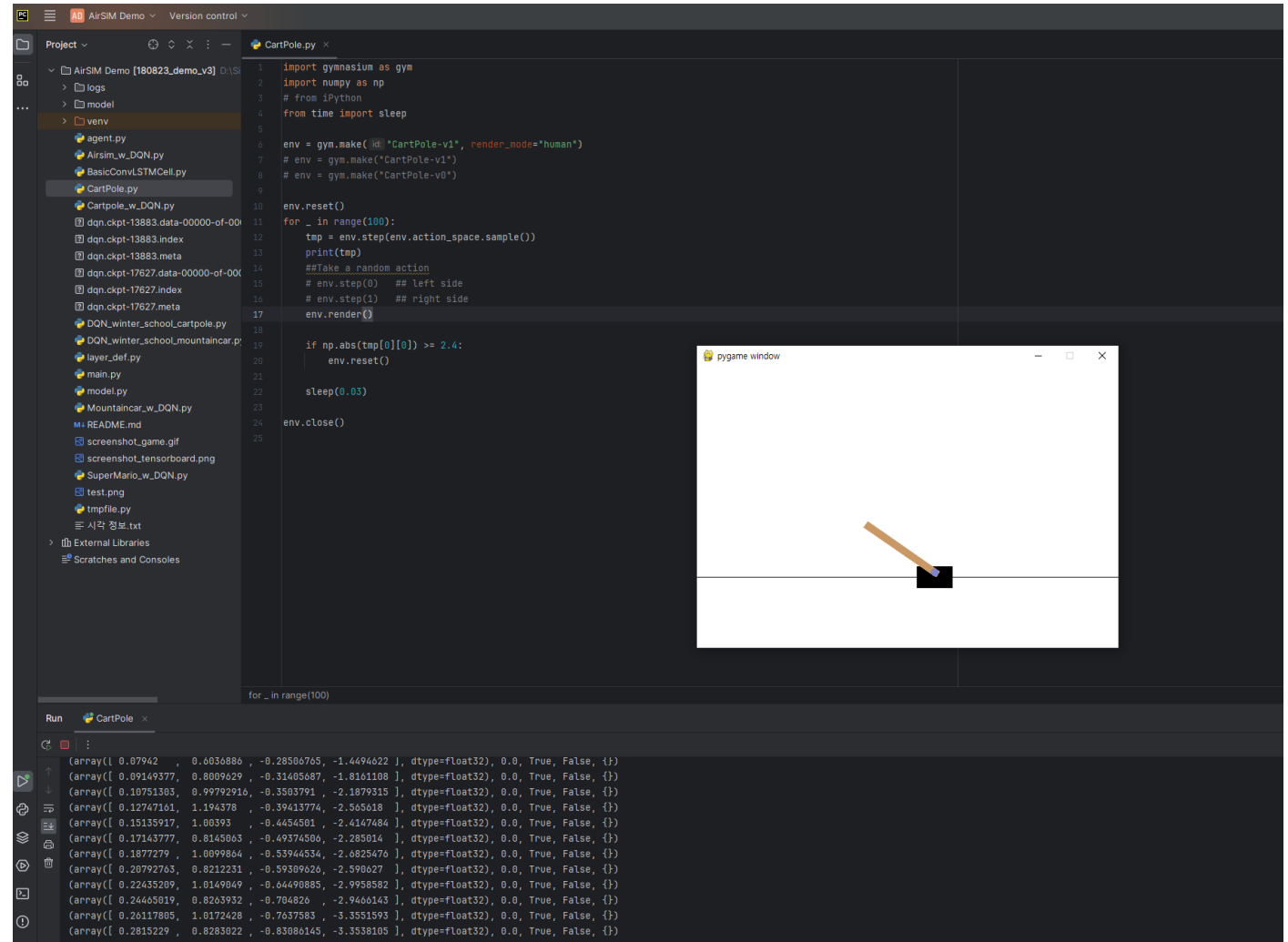
# Coding Practice Setup Guide – Test Sample Code

- Test Sample Code



# Coding Practice Setup Guide – Test Sample Code

- Test Sample Code





# Coding Practice Setup Guide – Test Sample Code

- **Test Sample Code**
  - [https://gymnasium.farama.org/environments/classic\\_control/cart\\_pole/](https://gymnasium.farama.org/environments/classic_control/cart_pole/)
- **Action Space**
  - 0 : Push Cart to the left
  - 1 : Push Cart to the right
- **Observation Space**
  - {(Position, Velocity, Angle, Angular Velocity)}
- **Rewards**
  - +1 for every step taken.
- **Terminates**
  - Angle greater than  $\pm 12^\circ$
  - Position greater than  $\pm 2.4$
  - Step length greater than 500

## Cart Pole #



This environment is part of the Classic Control environments which contains general information about the environment.

Action Space	Discrete(2)
Observation Space	Box([-4.8000002e+00 -3.4028235e+38 -4.1887903e-01 -3.4028235e+38], [4.8000002e+00 3.4028235e+38 4.1887903e-01 3.4028235e+38], (4,), float32)
import	<code>gymnasium.make("CartPole-v1")</code>

## Description

This environment corresponds to the version of the cart-pole problem described by Barto, Sutton, and Anderson in ["Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problem"](#). A pole is attached by an un-actuated joint to a cart, which moves along a frictionless track. The pendulum is placed upright on the cart and the goal is to balance the pole by applying forces in the left and right direction on the cart.

## Action Space

The action is a `ndarray` with shape `(1,)` which can take values `{0, 1}` indicating the direction of the fixed force the cart is pushed with.

- 0: Push cart to the left
- 1: Push cart to the right