

Development Env. Setup Guide

(부제 : RL 실습 환경 셋업 가이드)



JuSung Kang, LiberVance Co. Ltd.,
jskang.libervance@gmail.com,

Development Env. Setup Guide

- **Requirements**

- **OS**

- Windows 10/11

- **Packages**

- Anaconda3 for Python 3.x
 - Pytorch
 - Pycharm
 - OpenAI Gymnasium



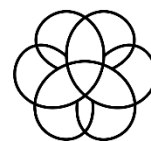
ANACONDA®



PyTorch



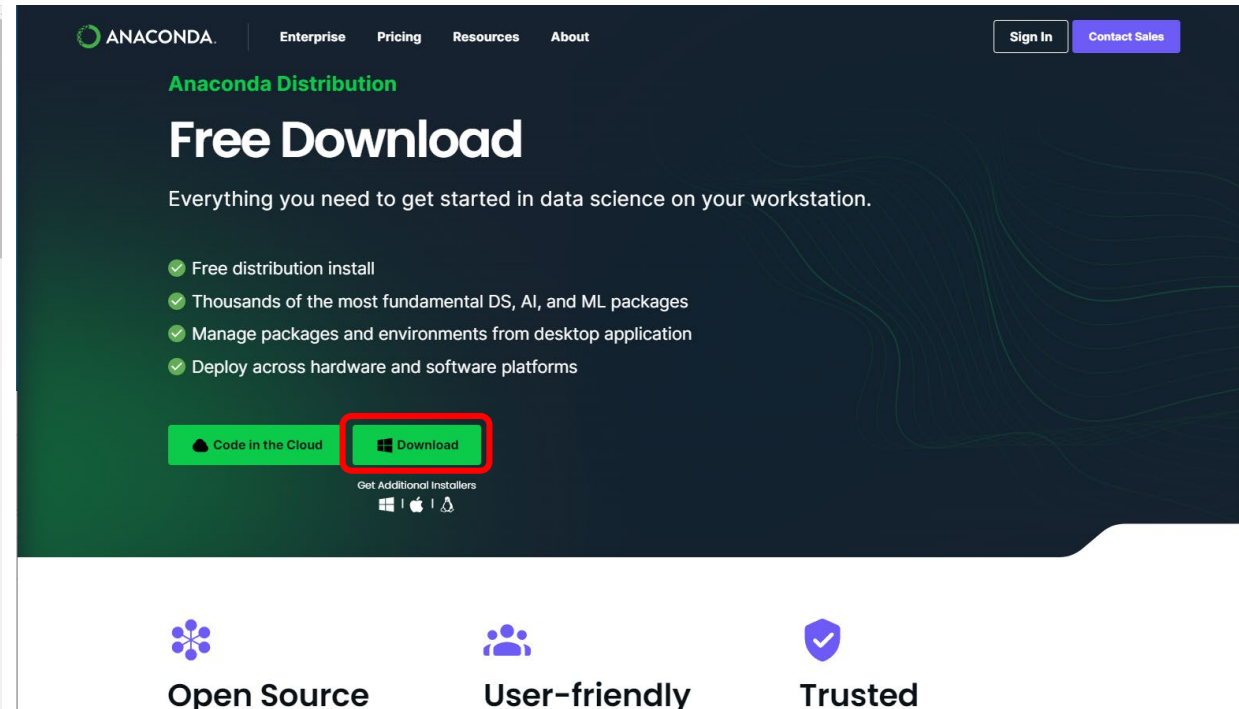
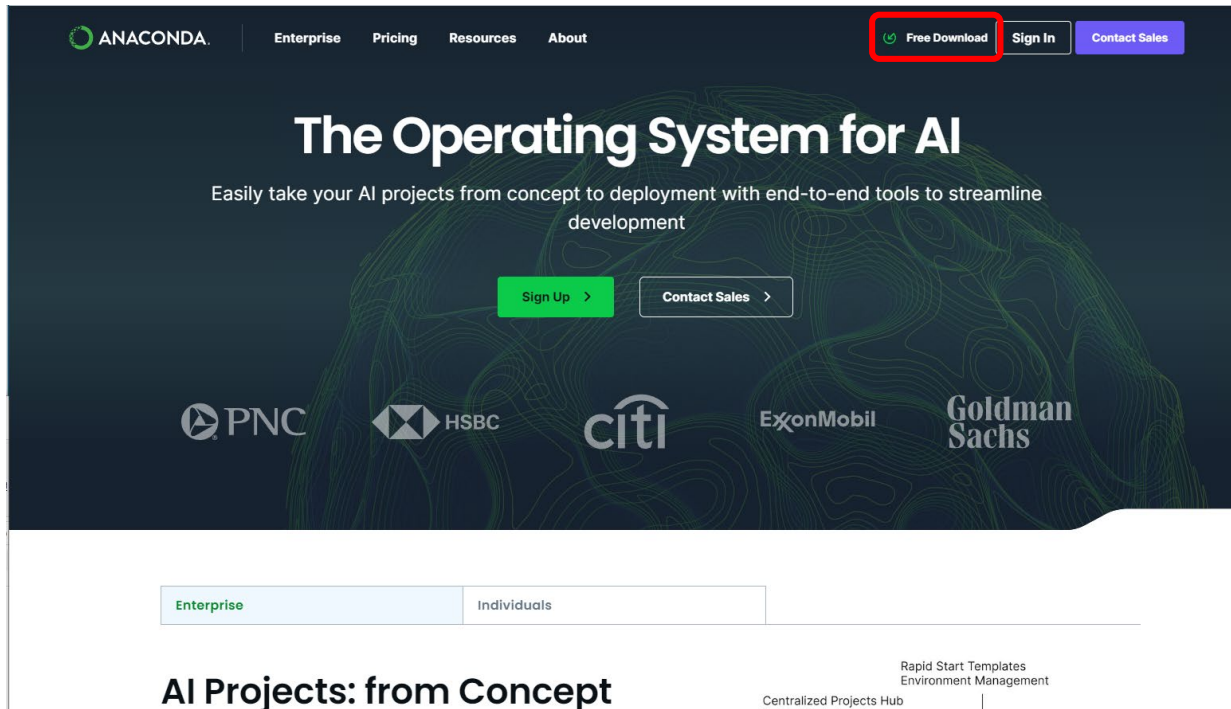
PyCharm



Gymnasium

Development Env. Setup Guide – Anaconda3

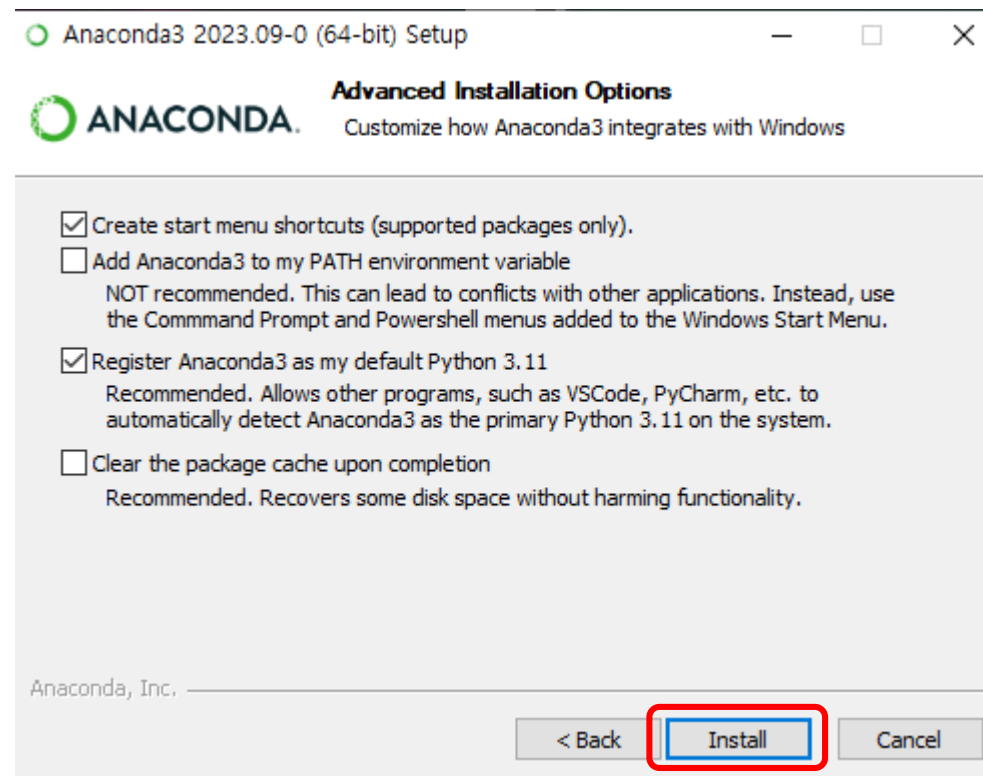
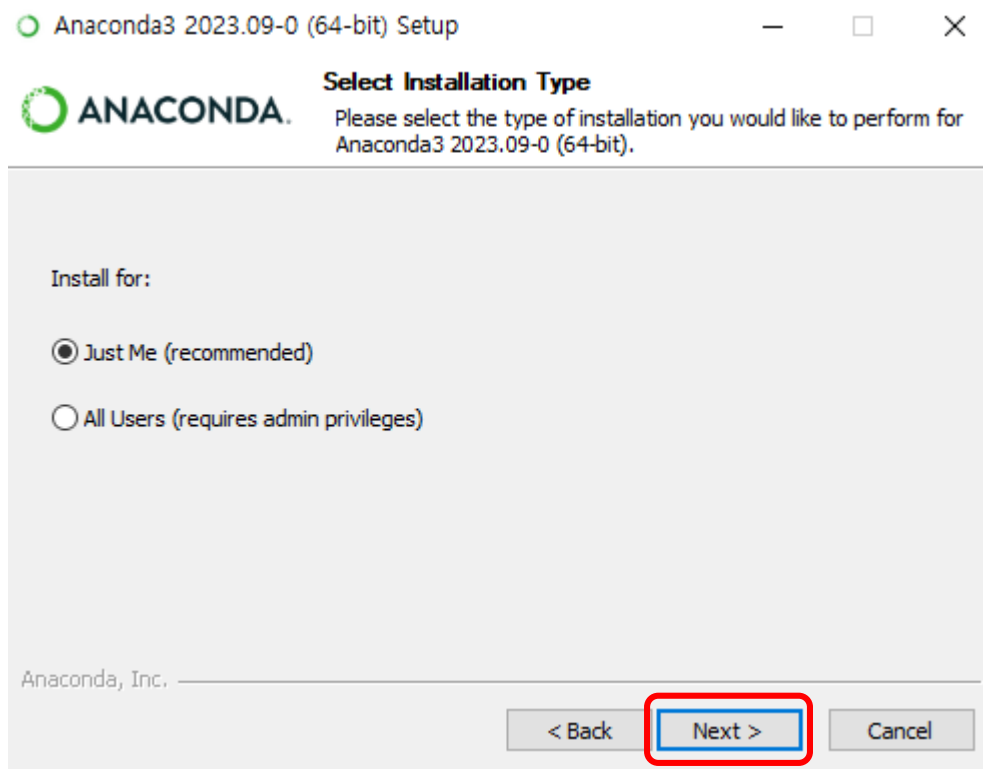
- Anaconda3 (Download)
 - <https://www.anaconda.com/>
 - Free Download >> Download >> (files) Anaconda3-2023.09-0-Windows-x86_64.exe



Development Env. Setup Guide – Anaconda3

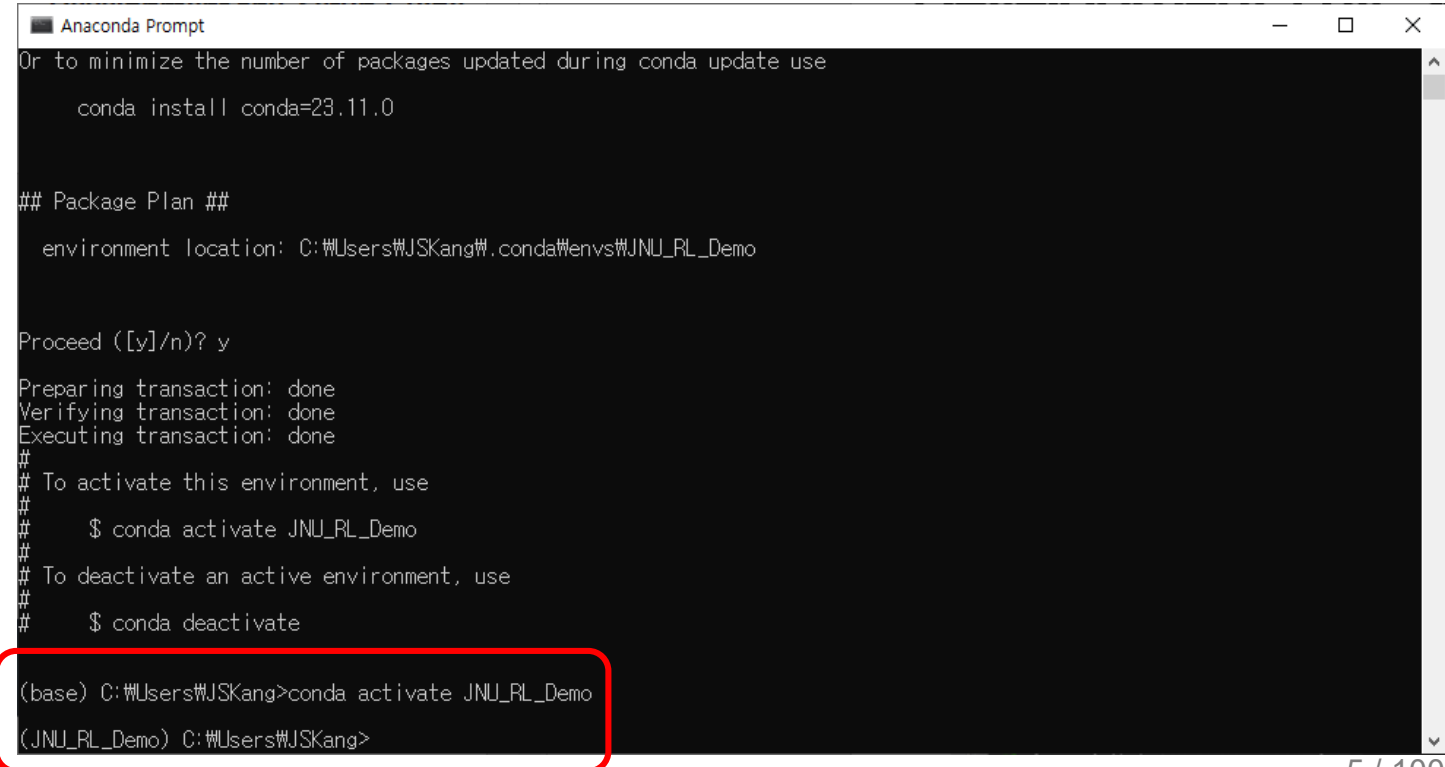
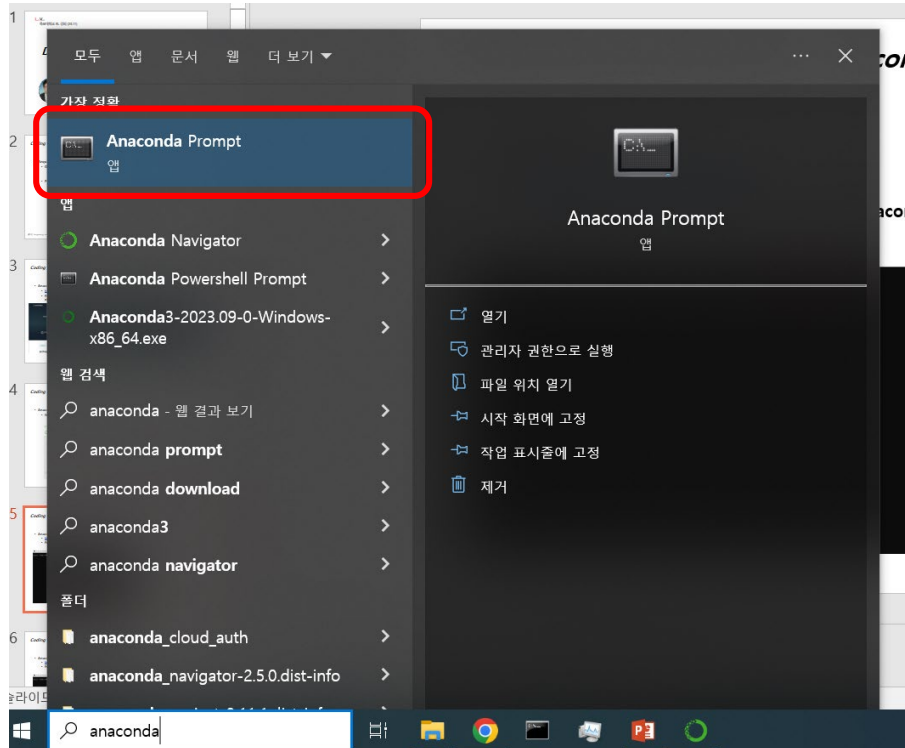
- **Anaconda3 (Install)**

- **Next >> I Agree >> Next >> Next >> Install >> Next >> Finish**



Development Env. Setup Guide – Anaconda3

- Anaconda3
 - 검색창 >> Anaconda Prompt 실행
 - >(base) conda create -n JNU_RL_Demo
 - >(base) conda activate JNU_RL_Demo



Development Env. Setup Guide – Pytorch

- Pytorch

- <https://pytorch.org/get-started/locally/>
- > (JNU_RL_Demo) conda install pytorch torchvision torchaudio cpuonly -c pytorch
 - * GPU computing - H/W 에 맞는 Cuda, cuDNN 설치 필요
 - * <https://wannabenice.tistory.com/54>

PyTorch Build	Stable (2.1.1)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.8	CUDA 12.1	ROCm 5.6	CPU
Run this Command:	conda install pytorch torchvision torchaudio cpuonly -c pytorch			

```
Anaconda Prompt - conda install pytorch torchvision torchaudio cpuonly -c pytorch
(JNU_RL_Demo) C:\Users\JUSKang>conda install pytorch torchvision torchaudio cpuonly -c pytorch
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
current version: 23.7.4
latest version: 23.11.0

Please update conda by running

    $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

    conda install conda=23.11.0

## Package Plan ##

environment location: C:\Users\JUSKang\Anaconda3\envs\JNU_RL_Demo

added / updated specs:
- cpuonly
- pytorch
- torchaudio
- torchvision
```

Development Env. Setup Guide – OpenAI Gym, Matplotlib

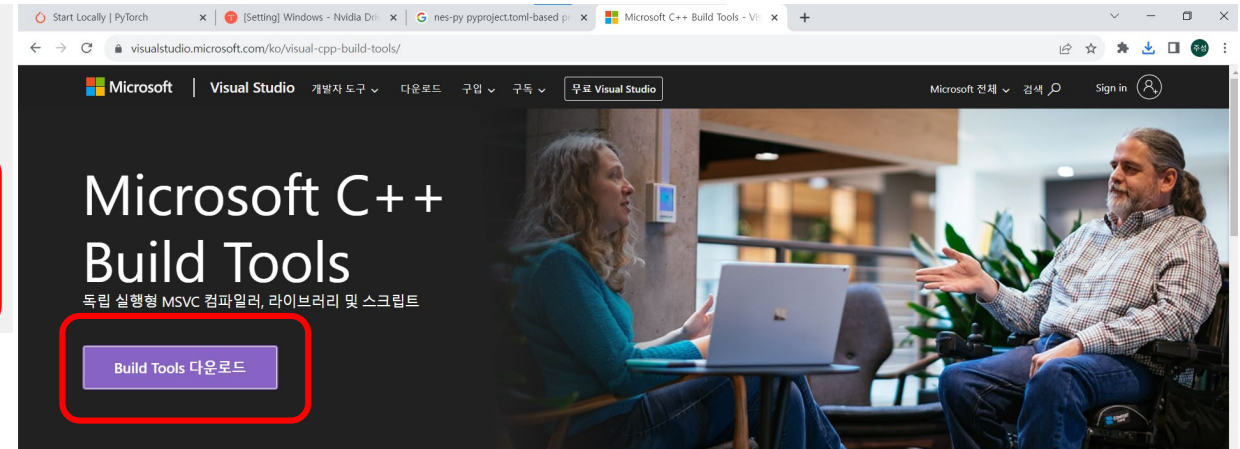
- **OpenAI Gym**
 - **> (JNU_RL_Demo) pip install nes-py**
 - If error message occurs, see the next page
 - “~ pyproject.toml-based projects”
 - **> (JNU_RL_Demo) pip install gymnasium[classic-control] gym-super-mario-bros==7.4.0**
- **Matplotlib**
 - **> (JNU_RL_Demo) conda install -c conda-forge matplotlib**

Development Env. Setup Guide – OpenAI Gym (Opt.)

- OpenAI Gym (Optional – For Debugging)
 - > (JNU_RL_Demo) pip install nes-py
 - If error message occurs, go on the the MS site as bellow:
 - “~ pyproject.toml-based projects”
 - <https://visualstudio.microsoft.com/ko/visual-cpp-build-tools/>
 - vs_BuildTools.exe 실행

```
copying nes_py#app#play_human.py -> build#lib.win-amd64-cpython-311#nes_py#app
copying nes_py#app#play_random.py -> build#lib.win-amd64-cpython-311#nes_py#app
copying nes_py#app#__init__.py -> build#lib.win-amd64-cpython-311#nes_py#app
creating build#lib.win-amd64-cpython-311#nes_py#wrappers
copying nes_py#wrappers#joypad_space.py -> build#lib.win-amd64-cpython-311#nes_py#wrappers
copying nes_py#wrappers#__init__.py -> build#lib.win-amd64-cpython-311#nes_py#wrappers
running build_ext
building 'nes_py.lib_nes_env' extension
error: Microsoft Visual C++ 14.0 or greater is required. Get it with "Microsoft C++ Build Tools": https://visualstudio.microsoft.com/visual-cpp-build-tools/
[end of output]

note: This error originates from a subprocess, and is likely not a problem with pip.
ERROR: Failed building wheel for nes-py
Running setup.py clean for nes-py
Failed to build nes-py
ERROR: Could not build wheels for nes-py, which is required to install pyproject.toml-based projects
```



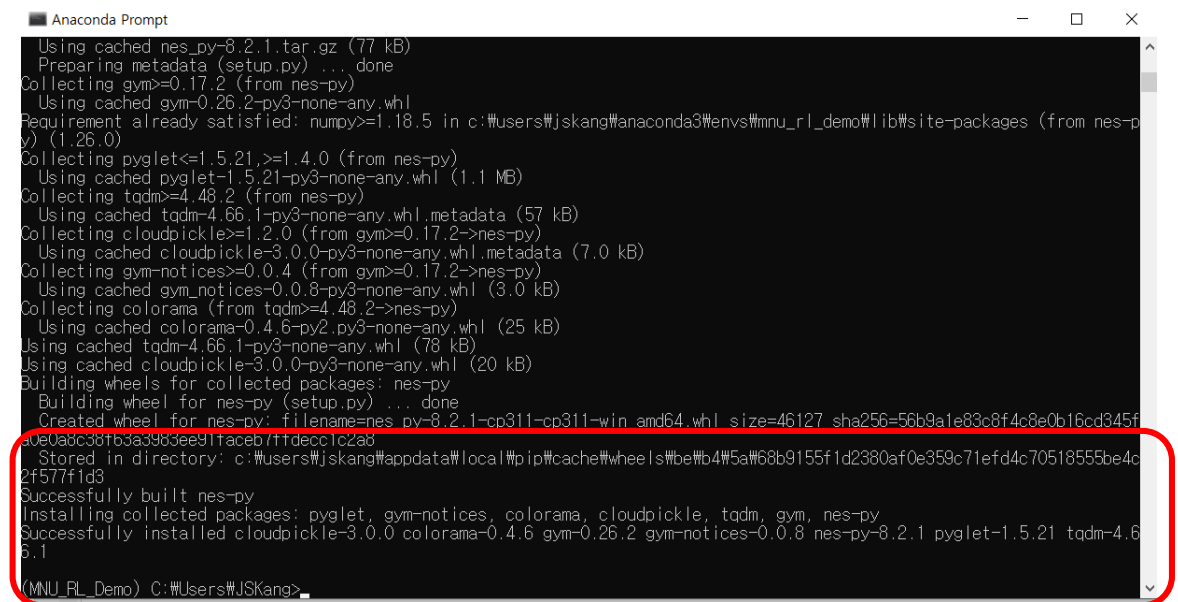
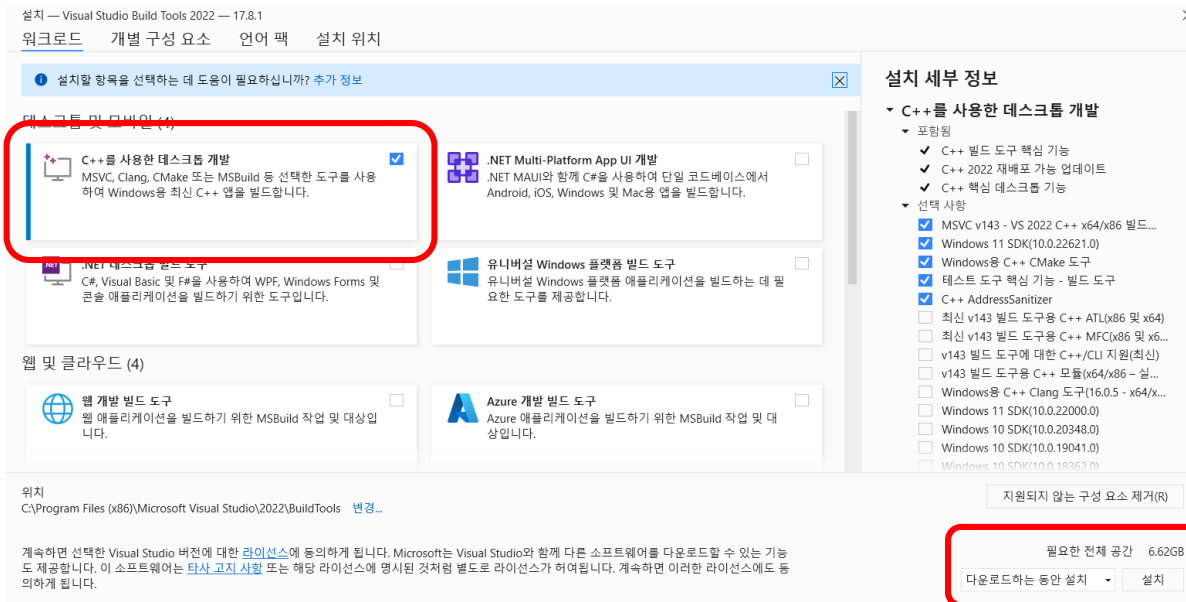
Microsoft C++ Build Tools는 Visual Studio 없이 스크립트 가능한 독립 실행형 설치 관리자를 통해 MSVC 도구 집합을 제공합니다. 명령줄에서 Windows를 대상으로 하는 C++ 라이브러리 및 애플리케이션을 빌드하는 경우에 권장합니다(예: 연속 통합 워크플로의 일부로). Visual Studio 2015 업데이트 3, Visual Studio 2017, Visual

Development Env. Setup Guide – OpenAI Gym (Opt.)

- OpenAI Gym (Optional – For Debugging)

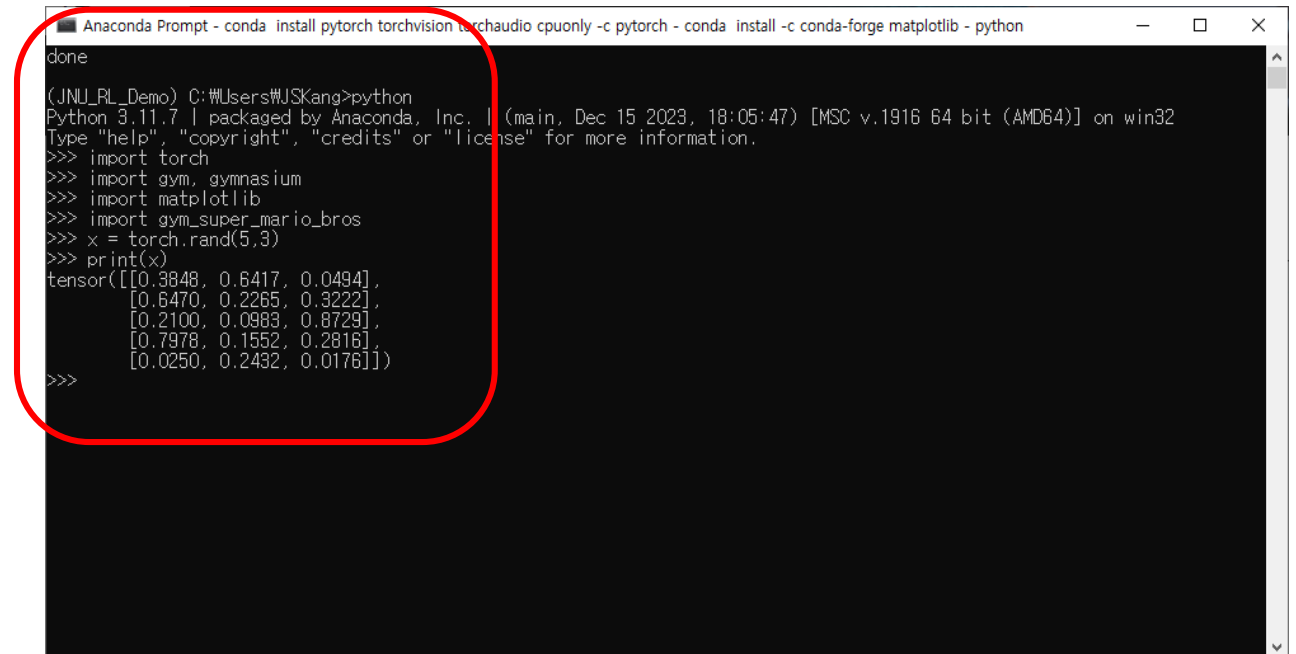
- > (JNU_RL_Demo) pip install nes-py

- 계속 >> 'C++ 를 사용한 데스크톱 개발' 선택 및 설치 >> 설치 (few min.)
 - Nes-py 재설치 (Go back to page 7)



Development Env. Setup Guide – Env. Setup Test

- Env. Setup Test
 - > (JNU_RL_Demo) python
 - >> import torch
 - >> import gym, gymnasium
 - >> import matplotlib
 - >> import gym_super_mario_bros
 - >> x = torch.rand(5,3)
 - >> print(x)



```
Anaconda Prompt - conda install pytorch torchvision torchaudio cpuonly -c pytorch - conda install -c conda-forge matplotlib - python
done
(JNU_RL_Demo) C:\Users\#JSKang>python
Python 3.11.7 | packaged by Anaconda, Inc. | (main, Dec 15 2023, 18:05:47) [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([[0.3848, 0.6417, 0.0494],
        [0.6470, 0.2265, 0.3222],
        [0.2100, 0.0983, 0.8729],
        [0.7978, 0.1552, 0.2816],
        [0.0250, 0.2432, 0.0176]])
>>>
```

Development Env. Setup Guide – Env. Export and Import Method (Opt.)

- Env. Export and Import method (Optional – It takes few min.)
 - > (JNU_RL_Demo) conda deactivate
 - > (base) conda env create -n JNU_RL_Demo2 -f JNU_RL_Demo.yaml
 - Export env.) conda env export -n JNU_RL_Demo > JNU_RL_Demo.yaml

```
Anaconda Prompt - conda install pytorch torchvision torchaudio cpuonly -c pytorch - conda install -c conda-forge matplotlib - conda deactivate...
r-mario-bros=7.4.0 gymnasium=0.29.1 nes-py=8.2.1 pygame=2.5.2 pygamelet=1.5.21 tqdm=4.66.1
done
#
# To activate this environment, use
#
# $ conda activate JNU_RL_Demo2
#
# To deactivate an active environment, use
#
# $ conda deactivate
#

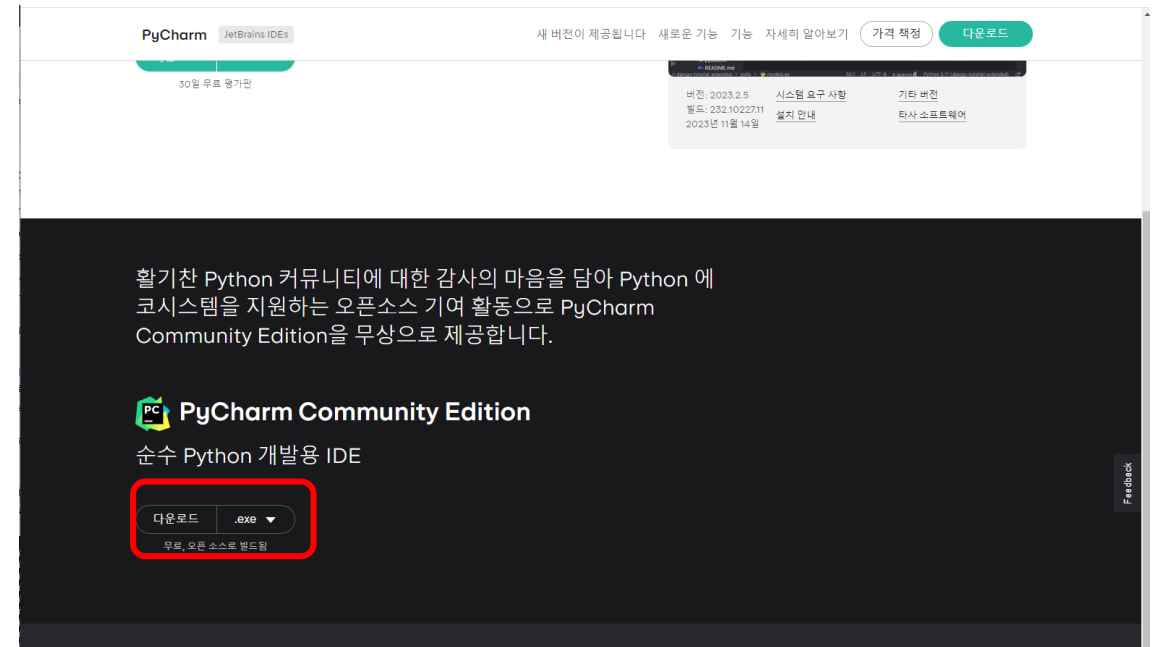
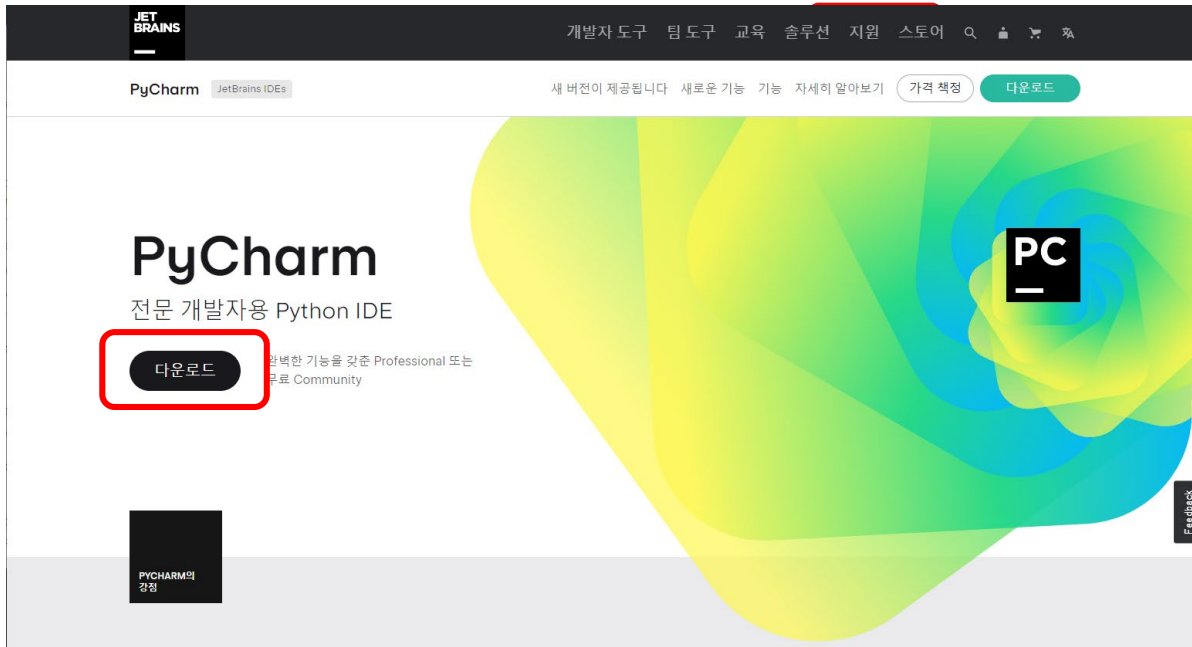
(base) C:\Users\JISKang>conda activate JNU_RL_Demo2
(JNU_RL_Demo2) C:\Users\JISKang>python
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import torch
>>> import gym, gymnasium
>>> import matplotlib
>>> import gym_super_mario_bros
>>> x = torch.rand(5,3)
>>> print(x)
tensor([[0.5292, 0.1673, 0.6159],
        [0.4462, 0.3810, 0.0795],
        [0.3638, 0.7424, 0.4290],
        [0.0237, 0.8891, 0.8027],
        [0.4692, 0.2695, 0.7416]])
>>>
```

```
JNU_RL_Demo - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
name: JNU_RL_Demo
channels:
- pytorch
- conda-forge
- defaults
dependencies:
- blas=1.0=mkl
- brotli=1.0.9=ha925a31_2
- brotli-python=1.0.9=py311hd77b12b_7
- bzip2=1.0.8=he774522_0
- ca-certificates=2023.11.17=h56e8100_0
- certifi=2023.11.17=pyhd8ed1ab_0
- cffi=1.16.0=py311h2bbff1b_0
- charset-normalizer=2.0.4=pyhd3eb1b0_0
- contourpy=1.2.0=py311h59b6b97_0
- cpuonly=2.0=0
- cryptography=41.0.3=py311h89fc84f_0
- cycler=0.12.1=pyhd8ed1ab_0
- filelock=3.13.1=py311haa95532_0
- fonttools=4.25.0=pyhd3eb1b0_0
- freetype=2.12.1=ha860e81_0
- giflib=5.2.1=h8cc25b3_3
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

Development Env. Setup Guide – Pycharm

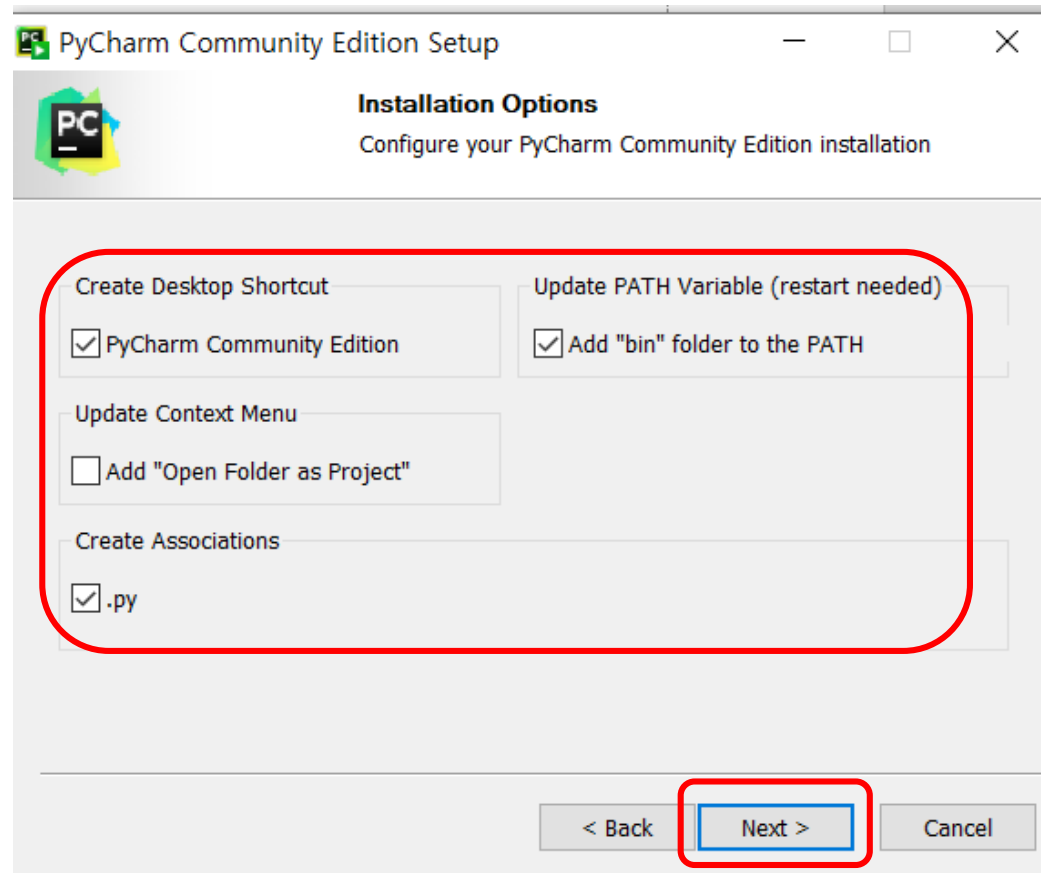
- Pycharm

- <https://www.jetbrains.com/ko-kr/pycharm/>
- 다운로드 > Pycharm Community Edition (다운로드) > (files) pycharm-community-2023.3.2.exe 실행



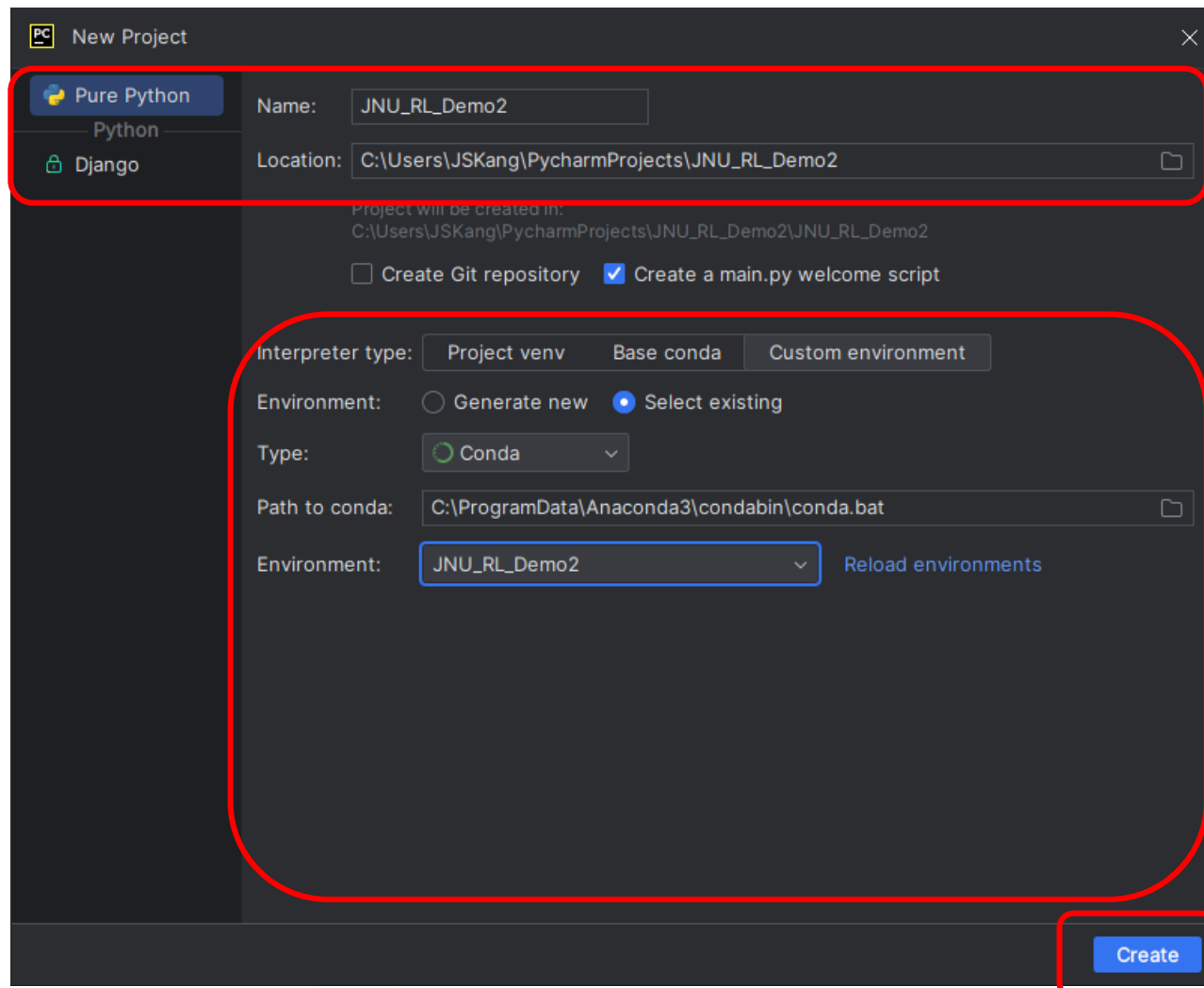
Development Env. Setup Guide – Pycharm

- Pycharm
 - Next >> Next >> Check options and Next >> Install



Development Env. Setup Guide – Pycharm

- **Pycharm Project Setting**
 - 검색창 >> Pycharm Community Edition 실행
- **Create New Project**
 - Location 지정
(Project Name & Location)
 - Interpreter Type
 - Custom Env. >> Select Existing >> Conda >> JNU_RL_Demo >> Create
 - * Interpreter location
 - C:\Users\JSKang\anaconda3\envs\JNU_RL_Demo\python.exe
 - C:\ProgramData\envs\JNU_RL_Demo\python.exe

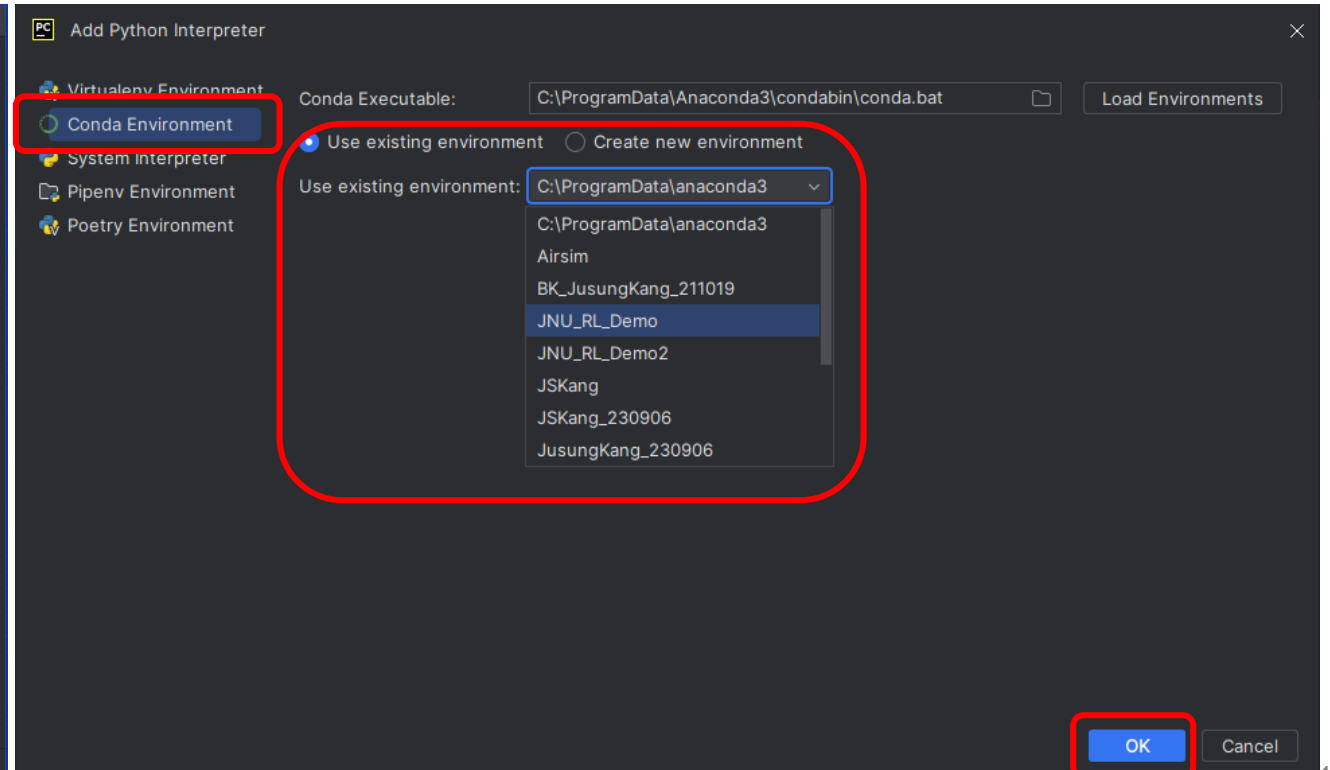
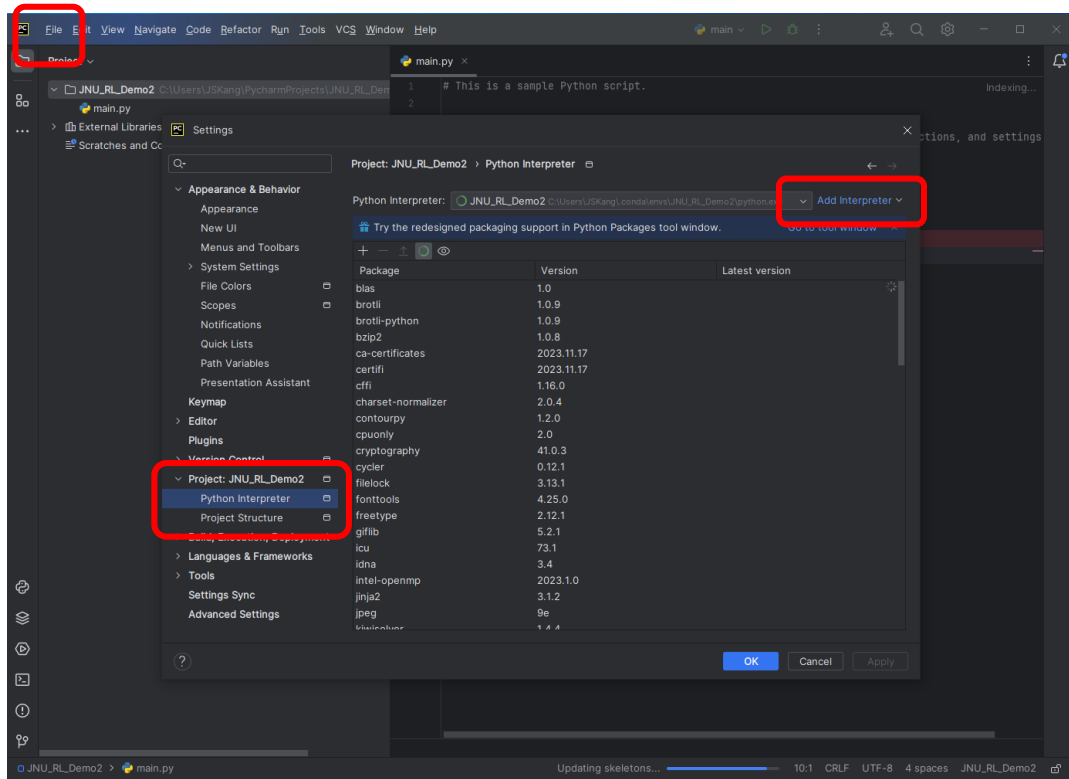


Development Env. Setup Guide – Pycharm

- Pycharm Project Setting

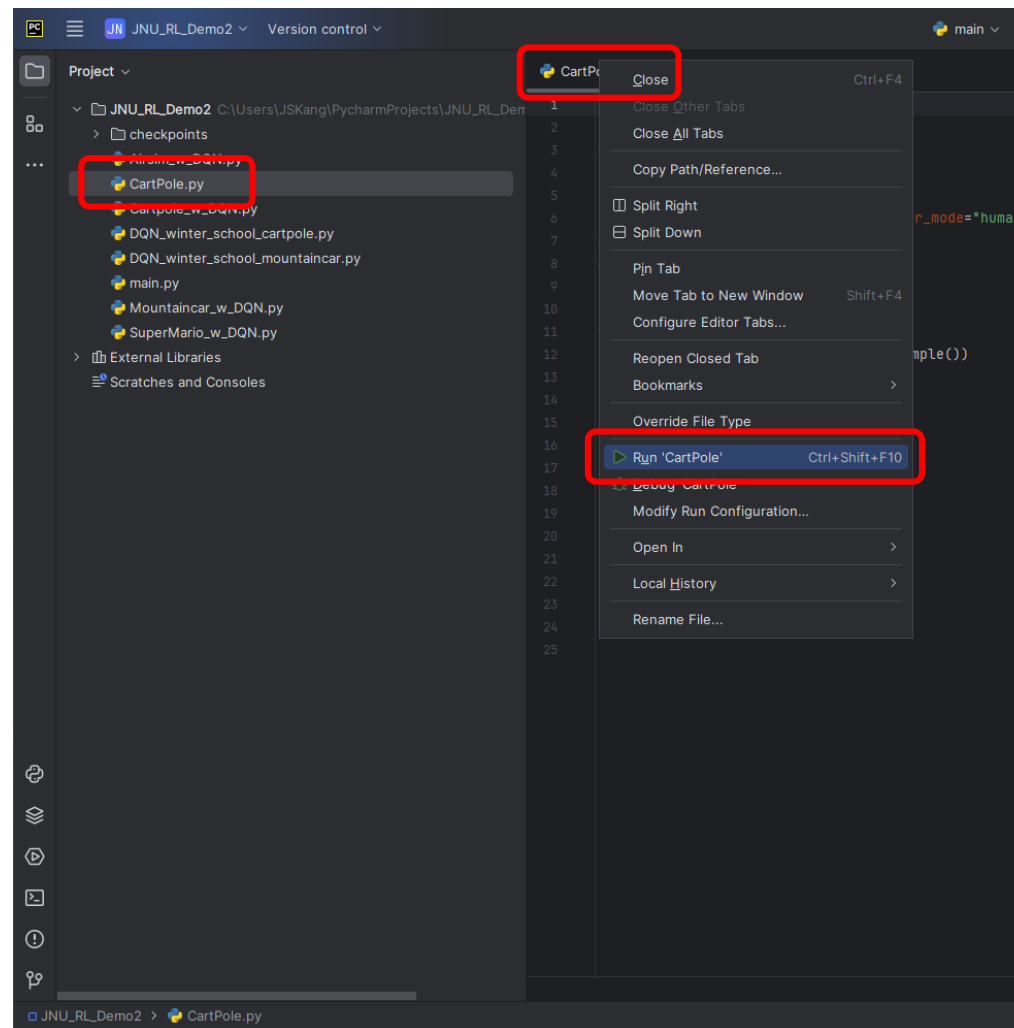
- Project Interpreter 설정방법

- File >> Settings >> Project >> Project Interpreter >> Add Interpreter >> Add local interpreter >> Conda Environment >> Select conda env. >> ok >> ok



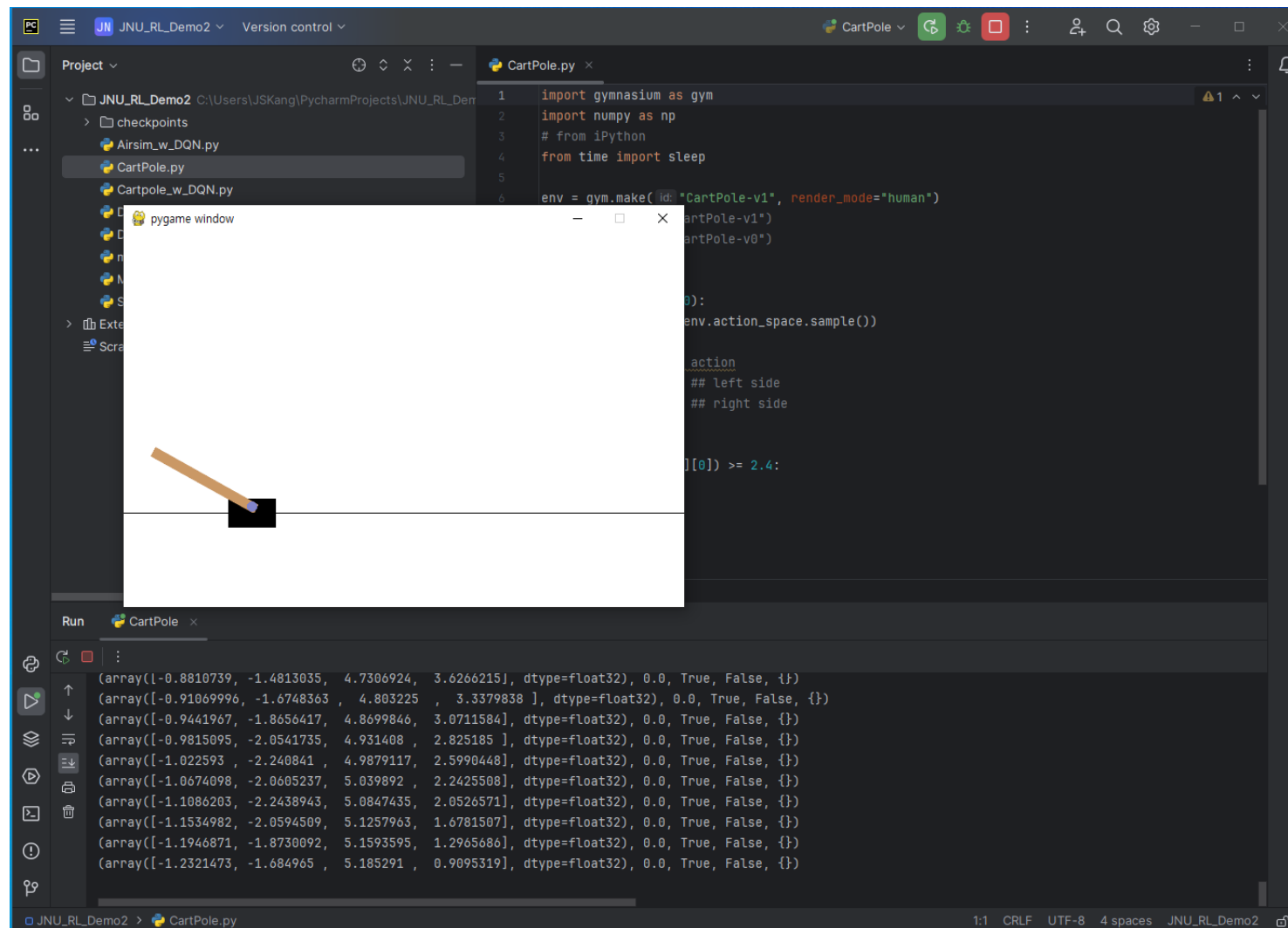
Development Env. Setup Guide – Pycharm Test code

- **Pycharm Setup test code**
 - **Copy all files to project location**
 - C:\Users\WJSKang\PycharmProjects\WJNU_RL_Demo
 - **'CartPole.py' file**
 - Right click >> Run 'CartPole'



Development Env. Setup Guide – Pycharm Test code

- Pycharm Setup test code
 - Setup Done!
 - Enjoy your RL



Development Env. Setup Guide – Pycharm Test code

- **Test Sample Code**
 - 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 ± 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