Gymnasium

- **Introduction**
  - OpenAI Gym is an open-source platform developed by OpenAI that provides a collection of environments for training and testing reinforcement learning agents.
  - It's a toolkit designed to help researchers and developers in the field of machine learning and artificial intelligence build and experiment with reinforcement learning algorithms.
  - You can visit the official website:
    - https://gymnasium.farama.org/index.html

Gymnasium

- **Installation**
  - Type the command on your prompt:

```
> pip install gymnasium
```

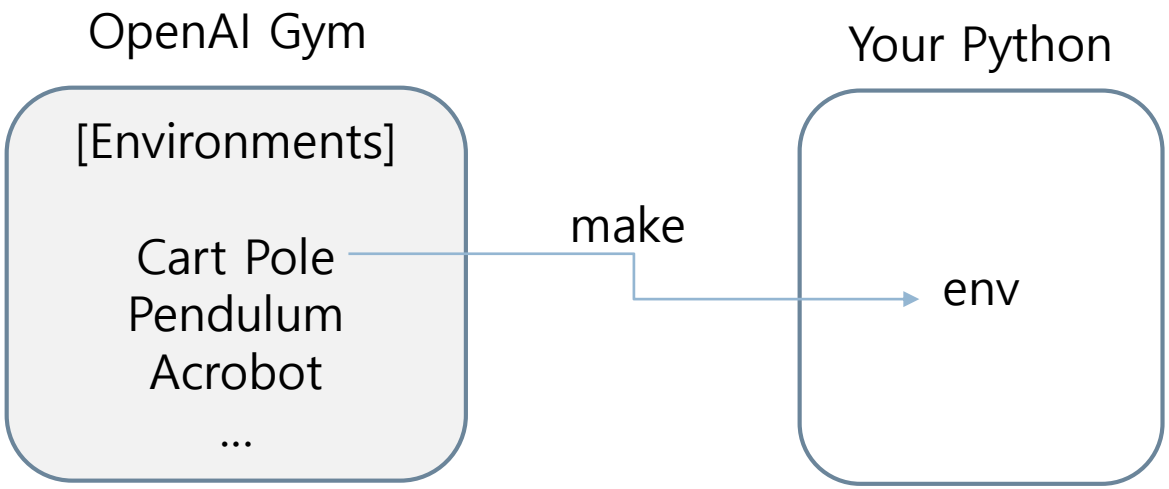
- **4 key functions**
  - make, reset, step and render

Gymnasium

- **1) Make**
  - **Load an environment**

```
import gymnasium as gym
env = gym.make('CartPole-v1')
```

-> Env. Name



Gymnasium

- **2) Reset**
  - **Initialize the environment**
    - **It provides an initial observation (state)**

```
observation, info = env.reset()
```

    - **Example**
      - In Cart Pole example, the observation includes [Cart position, Velocity, Pole angle, Pole angular velocity]
      - info is additional information of environment (if there is no information, it will be Null value)

Gymnasium

- **3) Step**
  - **Interaction between an agent and environment**

```
observation, reward, terminated, truncated, info = env.step(action)
```

Value	Description
Action	(ActType) An action value decided by the agent
Observation	(Object) this will be an element of the environment's
Reward	(Float) The reward returned as a result of taking the action
Terminated	(Bool) End of episode (True) or False
Truncated	(Bool) End of episode by unexpected error (True) or False
Info	(dictionary) Additional information (Null in the typical case)

Gymnasium

- **4) Render (mode)**
  - **Compute the render frames as specified by render_mode**

Option	Description
None (default)	no render is computed
Human	Visualization mode (render return None)
rgb_array	return a single frame representing the current state of the environment

Gymnasium

- **4) Render (mode)**
  - **Example (human model)**

```
import gymnasium as gym
import time

env = gym.make("CartPole-v1", render_mode='human')
state, info = env.reset()

for i in range(300):
    action = 1
    next_state, reward, terminated, truncated, info = env.step(action)

    # Render the env
    env.render()
    time.sleep(0.01)
    state = next_state

    if terminated:
        state, info = env.reset()

env.close()
```