

Gym



Download Python installer from:

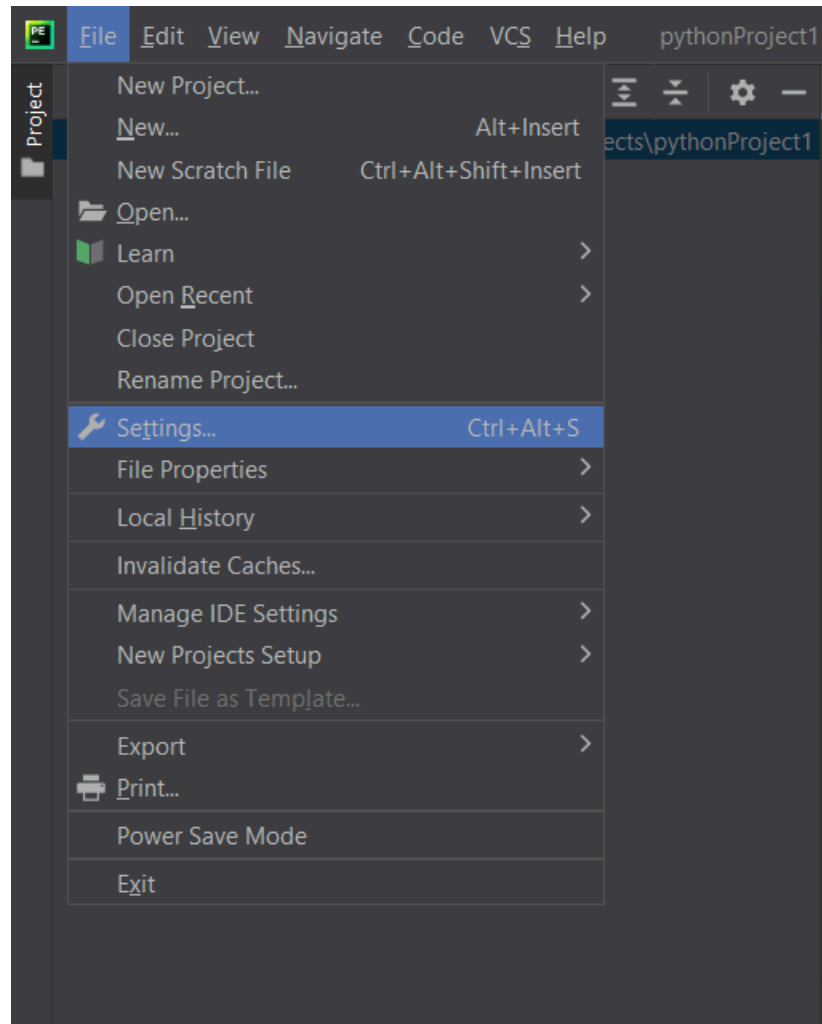
<https://www.python.org/downloads/windows>

Dwonload PyCharm from:

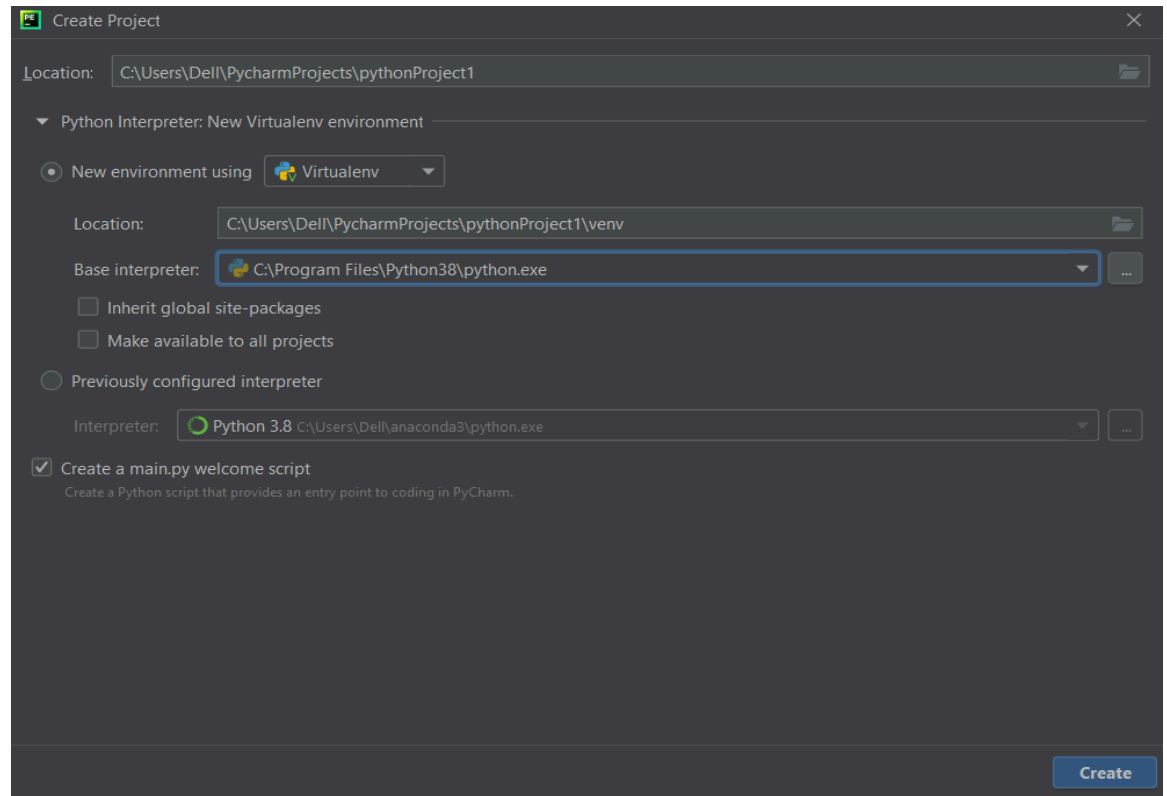
<https://www.jetbrains.com/pycharm/download/#section=windows>

✓ Creating a project:

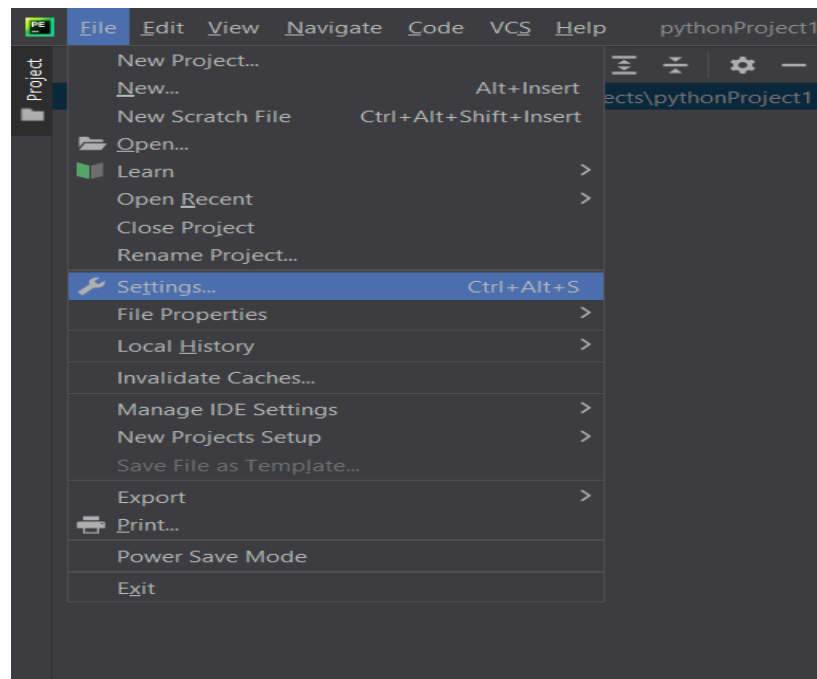
- o Step1 “creating new project”



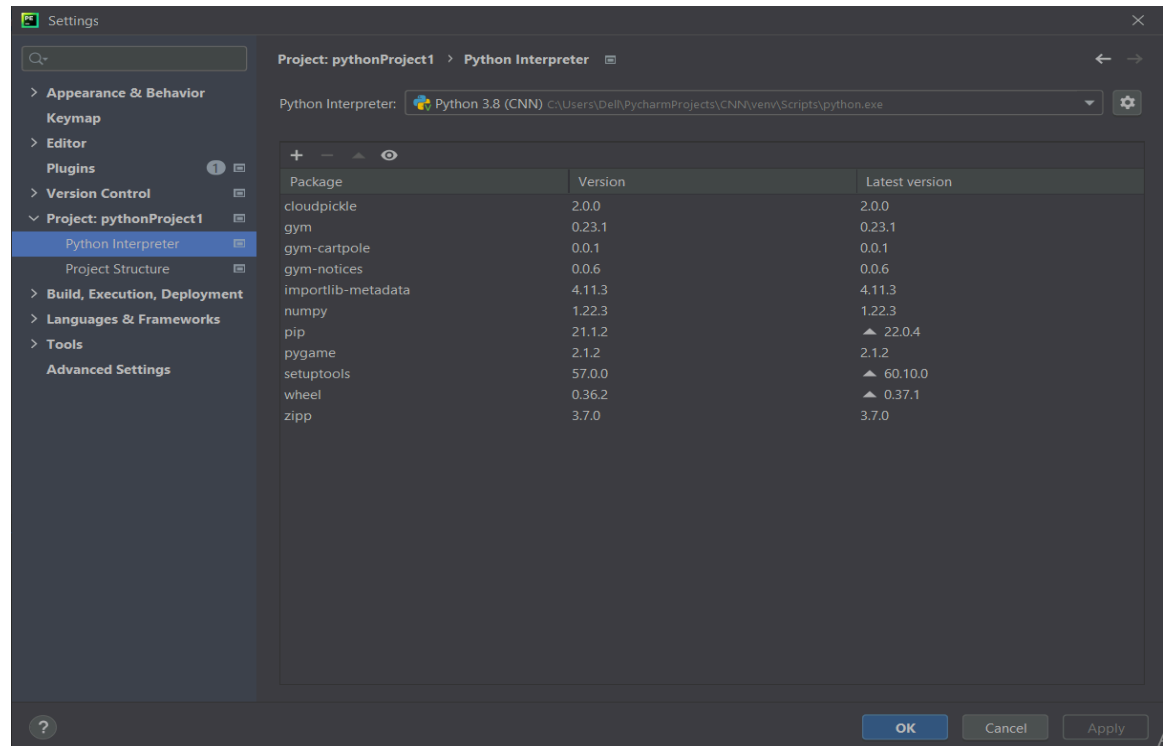
- o Step 2"set name and environment for the new project"



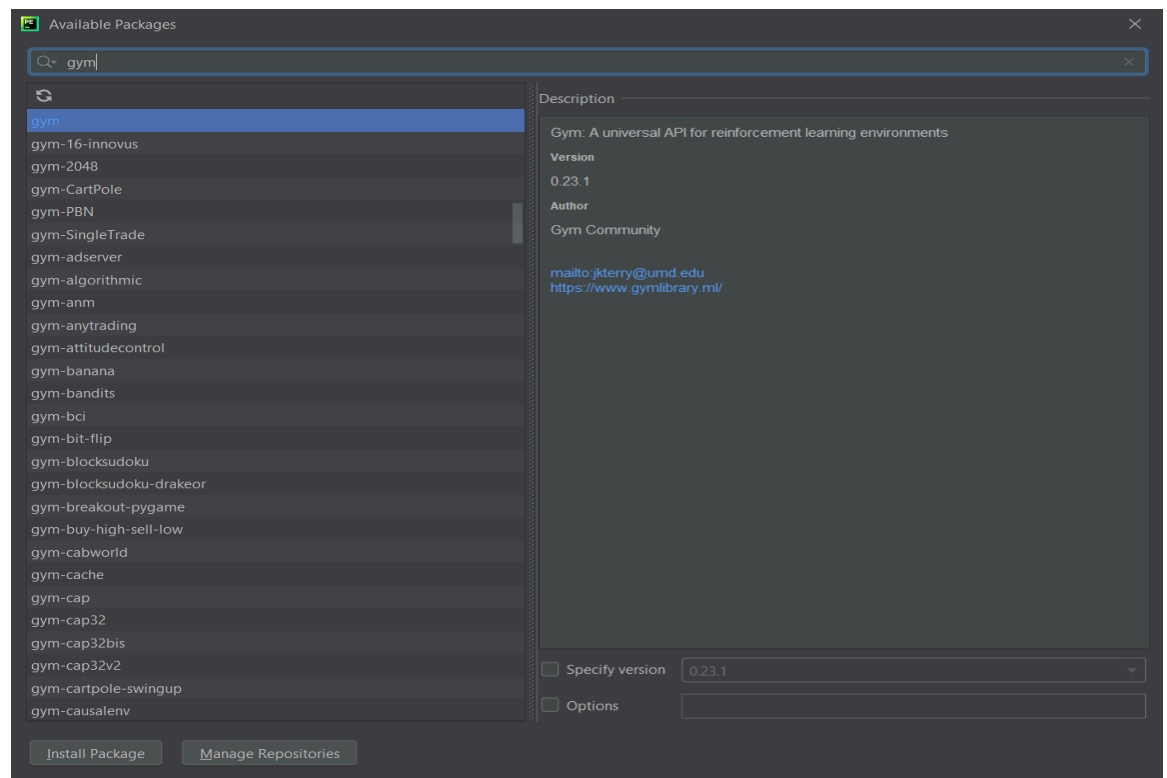
o Step 3



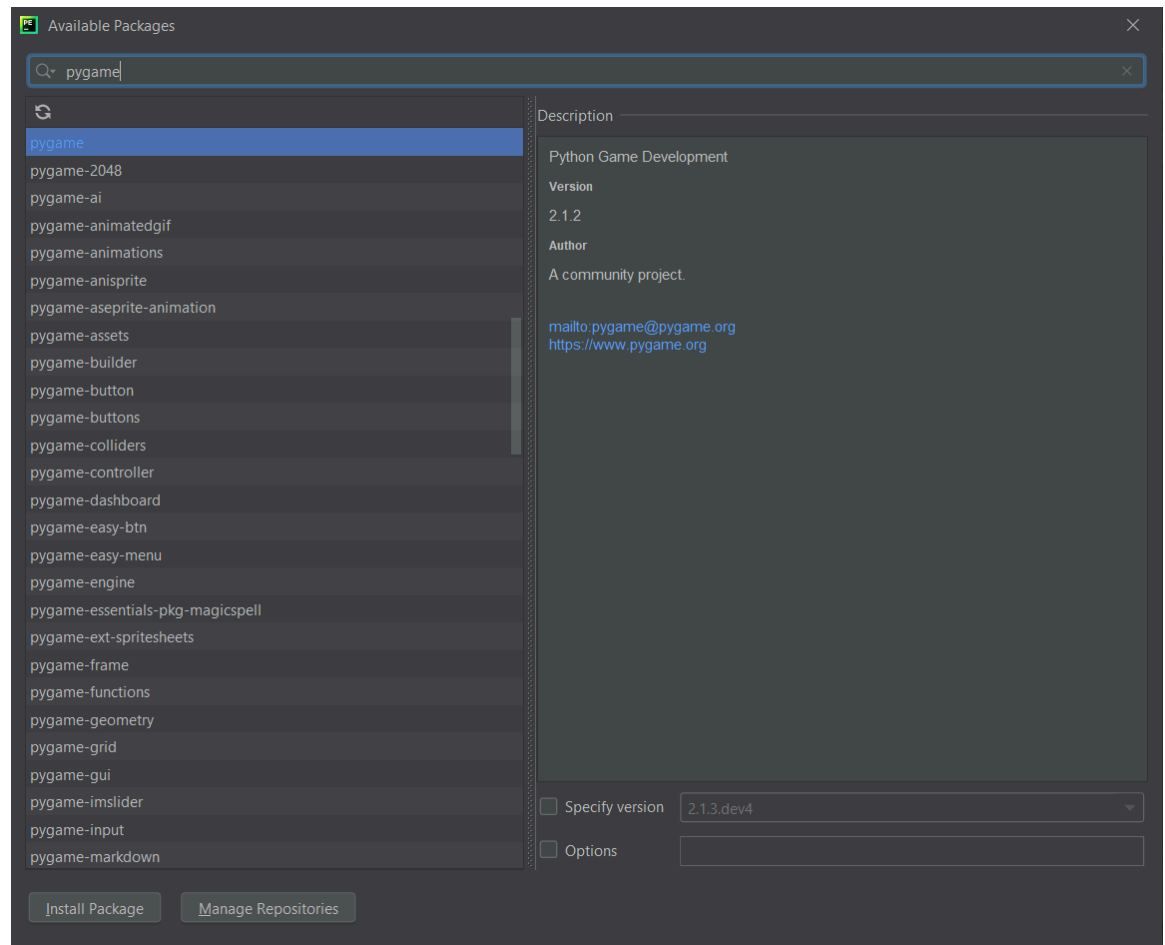
o Step 4 "choosing the interpreter"



o Step 5”installing gym”



o Step 6 “installing pygame”

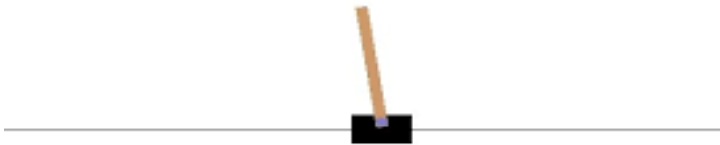


- o Step 7 “install gymnasium”
Use the same way “I think you don’t need screenshot for that”.

✓ What is OpenAI Gymnasium?

<https://gymnasium.farama.org/>

✓ Our first RL task: Cartpole



● Description

A pole is attached by an un-actuated joint to a cart, which moves along a frictionless track. The pendulum starts upright, and the goal is to prevent it from falling over by increasing and reducing the cart's velocity.

● Source

This environment corresponds to the version of the cart-pole problem described by Barto, Sutton, and Anderson

- **Environment**

- **Observation**

Type: Box (4)

Nu m	Observation	Min	Max
0	Cart Position	-2.4	2.4
1	Cart Velocity	-Inf	Inf
2	Pole Angle	~ -0.418 rad (-24°)	~ 0.418 rad (24°)
3	Pole Velocity At Tip	-Inf	Inf

- **Actions**

Type: Discrete (2)

Nu m	Action
0	Push cart to the left
1	Push cart to the right

Note: The amount the velocity is reduced or increased is not fixed as it depends on the angle the pole is pointing. This is because the center of gravity of the pole increases the amount of energy needed to move the cart underneath it

o Reward

Reward is 1 for every step taken, including the termination step. The threshold is 475 for v1.

o Starting State

All observations are assigned a uniform random value between ± 0.05 .

o Episode Termination

1. Pole Angle is more than $\pm 12^\circ$
2. Cart Position is more than ± 2.4 (center of the cart reaches the edge of the display)
3. Episode length is greater than 200 (500 for v1).

● Solved Requirements

Considered solved when the average reward is greater than or equal to 195.0 over 100 consecutive trials.

✓ Cartpole Implementation

1. Import gym

```
import gymnasium as gym
```

2. Use the make() function to set up the simulation environment.

```
env = gym.make('CartPole-v1', render_mode="human")
```

3. Use the reset() function to reset the environment to its initial state.

```
env.reset()
```

4. Visually inspect the environment using render() function.


```
env.render()
```

5. Sending an action to gym with step() function.

```
for _ in range(50):
    obs, reward, terminated, truncated, info = env.step(0)
    env.render()
    if terminated or truncated:
        break

env.close()
```

6. Calculating the reward.

```
for _ in range(50):
    print(f"Pole angle at step start: {np.degrees(obs[2])}", end="\n")
    obs, reward, terminated, truncated, info = env.step(0)
    print(f"Pole angle at step ends: {np.degrees(obs[2])}", end="\n")
    print(f"reward in this step: {reward}", end="\n")
    env.render()
    if terminated or truncated:
        break

env.close()
```

7. Making different episodes

```
for ep in range(5):
    print(f"Episode number is {ep+1}")
    obs, inf = env.reset()
    while True:
        print(f"Pole angle at step start: {np.degrees(obs[2])}", end="\n")
        obs, reward, terminated, truncated, info = env.step(0)
        print(f"Pole angle at step ends: {np.degrees(obs[2])}", end="\n")
        print(f"reward in this step: {reward}", end="\n")
        env.render()
        if terminated or truncated:
            break
env.close()
```