Refer:
https://www.learndatasci.com/tutorials/reinforcement-q-learning-scratch-python-openai-gym/

You are given the Taxi-v3 environment, where:

- A taxi agent operates in a 5x5 grid.
- It must pick up a passenger at one location and drop them off at another.
- The agent gets:
    a. **+20** for a successful drop-off,
    b. **-1** for each time step (to encourage faster completion),
    c. **-10** for illegal pick-up/drop-off actions.
- The filled square shall represents the taxi, which is yellow without a passenger and green with a passenger.
- The pipe ("|") represents a wall which the taxi cannot cross.
- R, G, Y, B are the possible pickup and destination locations. The blue letter represents the current passenger pick-up location, and the purple letter is the current destination.
- We have an Action Space of size 6 and a State Space of size 500. As you'll see, our RL algorithm won't need any more information than these two things. All we need is a way to identify a state uniquely by assigning a unique number to every possible state, and RL learns to choose an action number from 0-5 where:
    a. 0 = south
    b. 1 = north
    c. 2 = east
    d. 3 = west
    e. 4 = pickup
    f. 5 = dropoff

Recall that the 500 states correspond to a encoding of the taxi's location, the passenger's location, and the destination location.

Reinforcement Learning will learn a mapping of states to the optimal action to perform in that state by exploration, i.e. the agent explores the environment and takes actions based off rewards defined in the environment.

Your task is to:

- Initialize the Q-table with zeros for all state-action pairs.
- Implement the Q-learning algorithm. Train the agent for a sufficient number of episodes (e.g., 10,000).
- Track performance metrics:
    a. Total timesteps per episode
    b. Total penalties per episode
    c. Total reward per episode
- (Optional) Play around with the hyperparameters and observe how alpha, gamma, and epsilon affect learning.

Deadline : 31st May