Q2

Model Structure

My model consists of 4 linear layers that first increase in size and then decrease in size to the output size. Relu activation functions were used in the intermediary layers.

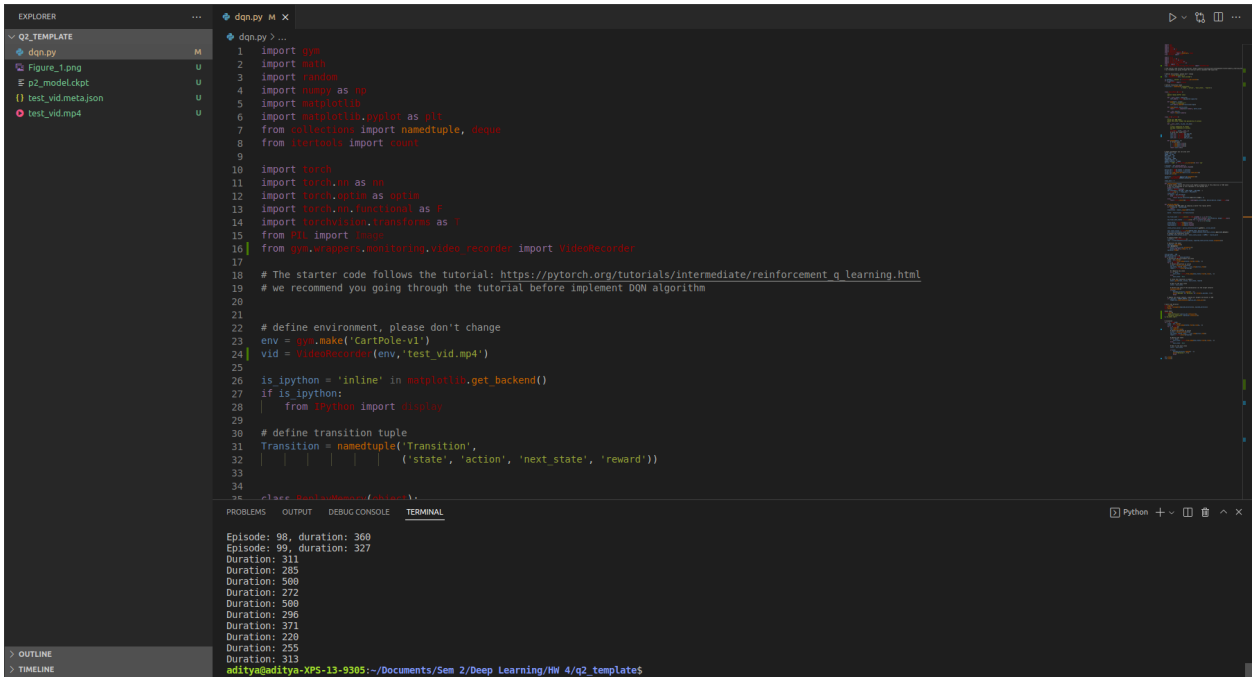
```python
class DQN(nn.Module):
    """
    build your DQN model:
    given the state, output the possiblity of actions
    """
    def __init__(self, in_dim, out_dim):
        """
        in_dim: dimension of states
        out_dim: dimension of actions
        """
        super(DQN, self).__init__()
        # build your model here
        self.fc1 = nn.Linear(in_dim,64)
        self.fc2 = nn.Linear(64,256)
        self.fc3 = nn.Linear(256,64)
        self.fc4 = nn.Linear(64,out_dim)

    def forward(self, x):
        # forward pass
        x = F.relu(self.fc1(x))
        x = F.relu(self.fc2(x))
        x = F.relu(self.fc3(x))
        return self.fc4(x)
```

Hyperparameters

The hyperparameters used for the DQN model were as follows
Batch size = 128
Gamma = 0.999
Eps start = 0.9
Eps end = 0.05
Eps decay = 2000
Target update = 10
Memory capacity = 10000
Number of train episodes = 100
Optimizer = Adam

Performance on test episodes

The model performed well on the test episodes and the mean duration of the 10 episodes was 332.3



Training Process