1.a) When we set epsilon to zero, we are eliminating randomness when choosing an action at a given state. All of the actions are based on the policy. In WorldWithoutThief, the reward values when epsilon is zero is zero, except for the reward for the first episode, which is zero.

b.) Changing epsilon to 0.1 gives 10% randomness. The first 10 - 15 rewards values are negative. The majority of reward values are positive hovering in the range of 100 - 140. Our policy isn't necessarily the optimal solution; therefore, it makes sense to add randomness to allow for better actions allowing our Agent to learn a better policy.

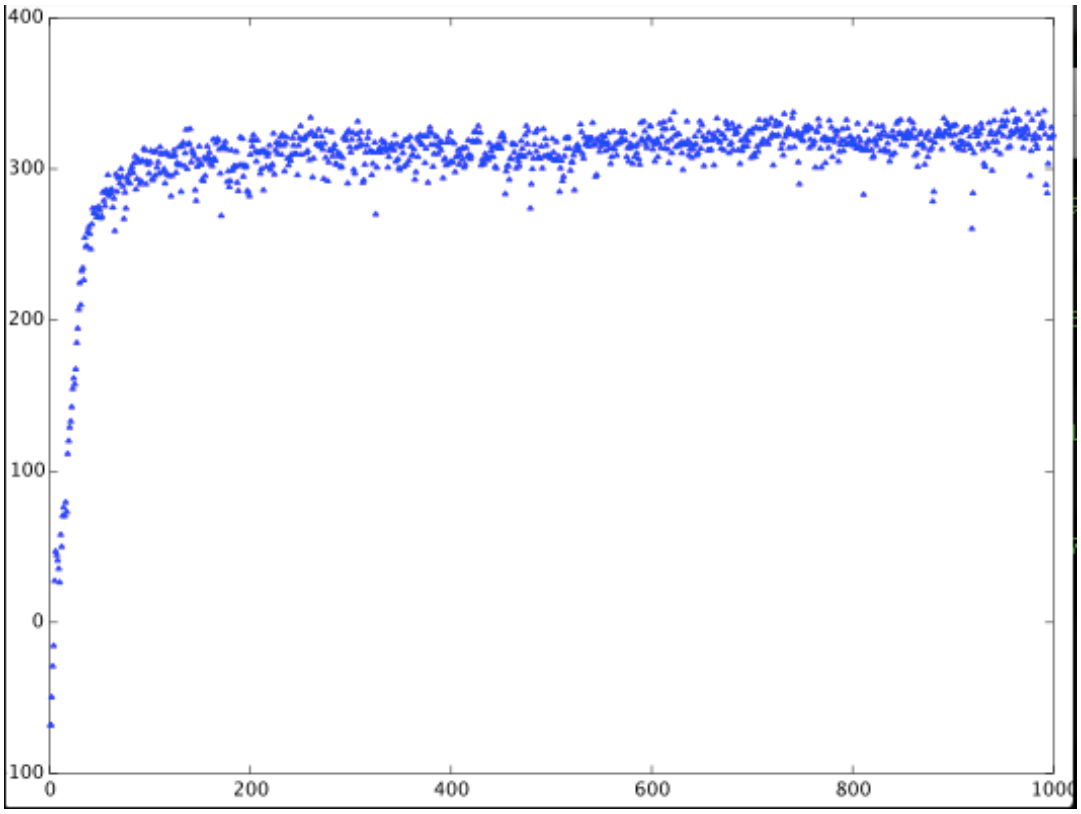
c.) Increasing epsilon to a high value such as 0.5, increases random decision making. The majority of our reward values are negative, which makes sense because we a making a lot of random decisions rather than making the majority of our decision from policy.

2.a) In WorldWithThief where the agent does not know where the thief is, one would expect poor performance because the agent has not learned to avoid the thief. This is proven in the results because the majority of the rewards are negative.

b.) Allowing the agent to know where the thief is made a huge increase in reward values as expected because the agent is actively trying to avoid the thief. The positive reward range is between 250 - 300.

c.) The best value for learning rate is 0.22 and for epsilon is 0.01. Since epsilon controls the probability that I choose a random value versus an optimal value from policy, I wanted to choose a low value. The learning rate is a fixed value that determines the extent of the new found information overriding the current q-value. We don't want our algorithm to completely rely on random predictions therefore we want a low value; however, we still want our agent to learn so we don't want it to be zero.

3.a)



b.) Averaging the rewards over 10 trials causes the rewards to converge to a range of about 300 - 320.