In your report, mention what you see in the agent's behavior. Does it eventually make it to the target location?

When the action is set to pick from random the car moves around at random and would sometimes bump into the target location, but as soon as I changed the action to be the next waypoint, the agent would eventually make it to the target location.

Justify why you picked these set of states, and how they model the agent and its environment.

I used stoplight and the next waypoint. This is a fairly basic representation of the agent's environment because it only takes these two variables into account, excluding what other drivers are doing. However in my trials I have not had an issue with traffic so this is what I'm using.

What changes do you notice in the agent's behavior?

Because I have initialized the Q table arbitrarily, the agent would sometimes miss the deadline early in the trials, but as the Q table builds up the agent would get to the target location without going over the deadline.

Report what changes you made to your basic implementation of Q-Learning to achieve the final version of the agent. How well does it perform?

At the beginning I had initialized the Q values with 0, but the agent would end up stuck in place. I then changed the initial value to be π and it improved the agent to the point where it can "reach the destination within the allotted time, with net reward remaining positive".

Does your agent get close to finding an optimal policy, i.e. reach the destination in the minimum possible time, and not incur any penalties?

Yes. The result for my last 100 trials shows that the agent never missed the deadline and had always obtained a positive total trip reward.