Observe what you see with the agent's behavior as it takes random actions.

*Does the **smartcab** eventually make it to the destination?*

- Yes it does!
- Somehow, the smartcab does make it to the destination eventually even though the action selection policy is random.

Are there any other interesting observations to note?

- There was nothing else interesting I could observe
- I was expecting some accidents since the action selection policy was random, but I did not notice any. I am guessing other agents might be smart enough to avoid an accident from occurring.
- Also, there was no change in the agent's state and for some reason the agent's action displayed at the top left corner does not match the agent's action displayed by its side.

*What states have you identified that are appropriate for modeling the **smartcab** and environment?*

- This is a very good question. My answer for this question kept changing as I spent more time modelling the agent. (I wanted to leave my previous answers in here to show the progression of my thinking, but it would too much work for the reviewer).
- Since we are concerned about the time to destination and traffic safety, I decided to go with states that had a bit of both in them.
- Basically, my states were a combination of the direction of the shortest distance to the destination and the traffic light signals (red, green with oncoming vehicles and green with NO oncoming vehicles)

Why do you believe each of these states to be appropriate for this problem?

- Like I noted in my previous answer, having some location information and information for monitoring traffic safety fully describes what could be occurring in the agent's world at each point in time.
- I believe having states that contain both location and traffic information would provide us with enough information at each point in time to make decisions about distance to the destination and safety of the smartcab's driving.

*How many states in total exist for the **smartcab** in this environment?*

- There can be a whole lot of states in the cab environment.
- I went down the path of having a state for each intersection and traffic light and the problem got very complex quickly and the amount of states grew really large!

- With my new state description model, we have a total of 25 states.
- 1 goal state and 24 intermediate states

Does this number seem reasonable given that the goal of Q-Learning is to learn and make informed decisions about each state? Why or why not?

- Having 25 states is not bad at all.
- Remember, our estimate of the Q value converges to the actual Q value as we visit all the state action pairs up to infinity!
- Infinity is a very long time!
- The more states you have the longer infinity becomes!

What changes do you notice in the agent's behavior when compared to the basic driving agent when random actions were always taken?

- The agent's actions were a lot less random after implementing Q Learning
- I could notice the agent getting to the destination faster and incurring the least penalties
- With random actions, it really is just about chance and time.
- The pretty cool thing or frustrating thing is how dumb the agent is at the beginning.
- Initially, it seems random is better since the agent is trying to visit all possible state action pairs to solidify its learning and watching it turn away from the destination broke my heart a couple of times.

Why is this behavior occurring?

- This behavior is occurring because in order for our estimate of the Q value to converge to its actual value, the estimated Q value for all state action pairs needs to be visited and updated infinitely.
- Taking very random actions or non-optimal actions early on, allows our algorithm to recover from local minima issues which I noticed until I added the decaying epsilon and tuned other parameters.

Report the different values for the parameters tuned in your basic implementation of Q-Learning. For which set of parameters does the agent perform best?

discount_rate (gamma)	learning_rate (alpha)	max_epsilon	epsilon_decay	num_trials	rank
0.9	0.9	0.8	0.00005	100	2nd
0.9	0.9	0.7	0.00095	100	3rd
0.9	0.9	0.9	0.005	100	4th
0.9	0.9	0.8	0.00085	200	1st

How well does the final driving agent perform?

- The final driving agent performs the best. The last 5 to 10 trials, the agent performs as optimal as it can.

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, I think it does.

How would you describe an optimal policy for this problem?

- I think the optimal policy for this problem would be to do the following:
  - For the most part stop at a red light
  - Avoid turning left when there is an oncoming traffic
  - Follow the shortest distance in terms of direction to the destination