UDACITY

Logout

CRITERIA	MEETS SPECIFICATIONS
Agent accepts inputs	Student is able to implement the desired interface to the agent that accepts specified inputs.
Produces a valid output	The driving agent produces a valid output (one of None, 'forward', 'left', 'right') in response to the inputs.
Runs in simulator	The driving agent runs in the simulator without errors. Rewards and penalties do not matter - it's okay for the agent to make mistakes.

Identify and update state

CRITERIA	MEETS SPECIFICATIONS
Reasonable states identified	Student has identified states that model the driving agent and environment, along with a sound justification.
Agent updates state	The driving agent updates its state when running, based on current input. The exact state does not matter, and need not be correlated with inputs, but it should change during a run.

Implement Q-Learning

CRITERIA	MEETS SPECIFICATIONS
Agent updates Q-values	The driving agent updates a table/mapping of Q-values correctly, implementing the Q-Learning algorithm.
Picks the best action	Given the current set of Q-values for a state, it picks the best available action.
Changes in behavior explained	Student has reported the changes in behavior observed, and provided a reasonable explanation for them.

Enhance the driving agent

CRITERIA	MEETS SPECIFICATIONS
Agent learns a feasible policy within 100 trials	The driving agent is able to consistently reach the destination within allotted time, with net reward remaining positive.
Improvements reported	Specific improvements made by the student beyond the basic Q-Learning implementation have been reported, including at least one parameter that was tuned along with the values tested. The corresponding results for each value are also reported.
Final agent performance discussed	A description is provided of what an ideal or optimal policy would be. The performance of the final driving agent is discussed and compared to how close it is to learning the stated optimal policy.

English

Student FAQ