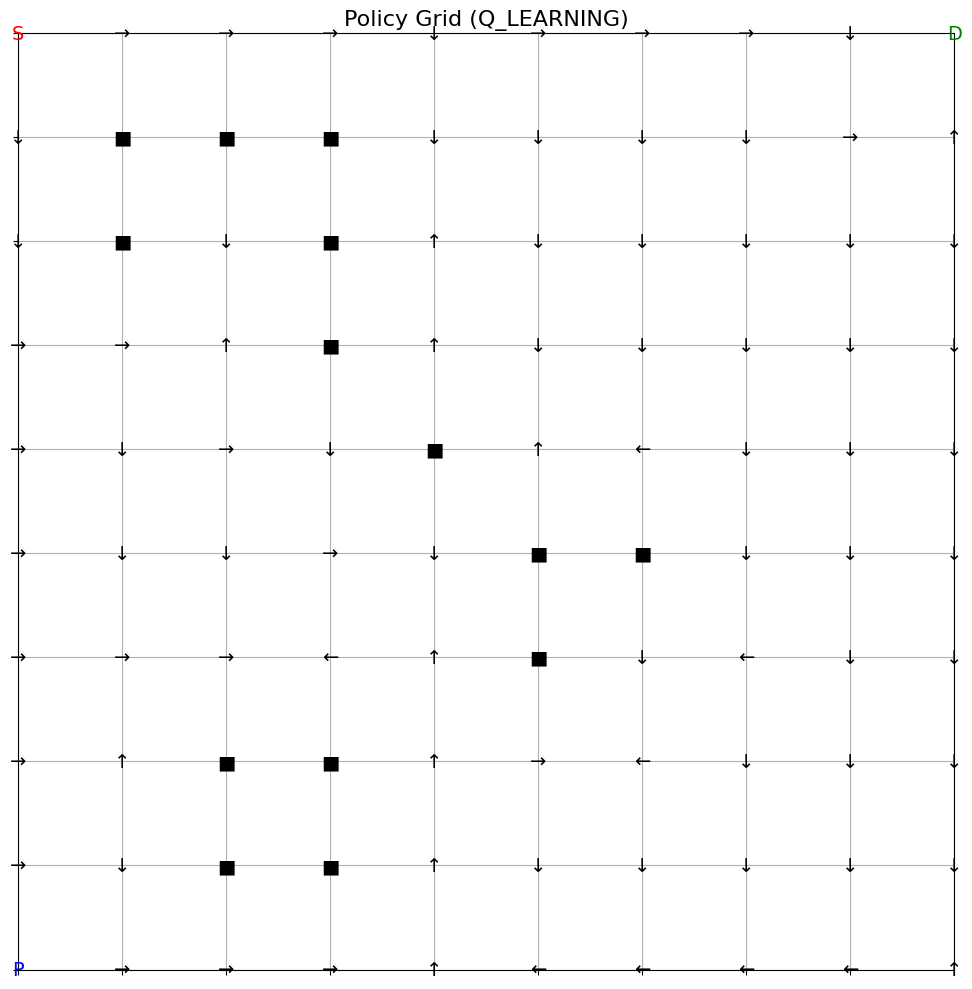
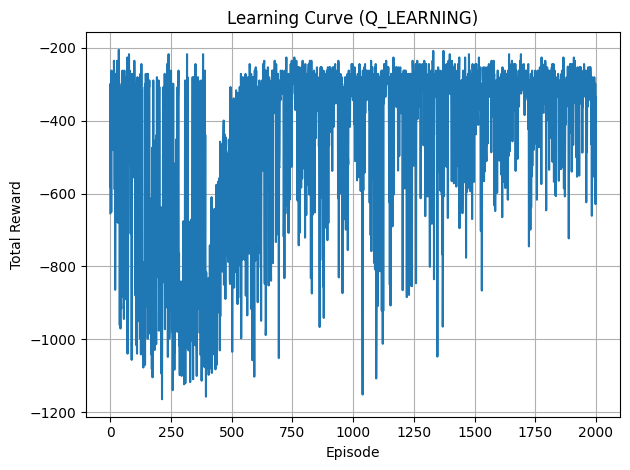
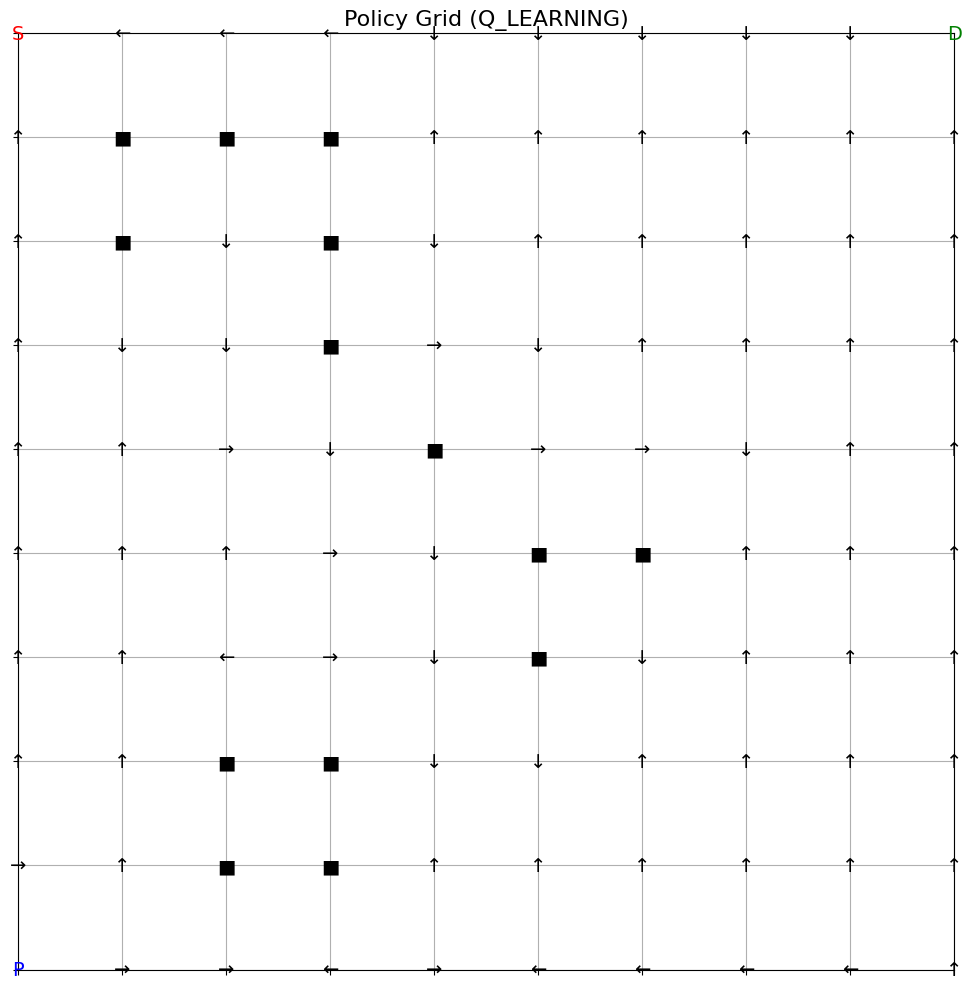
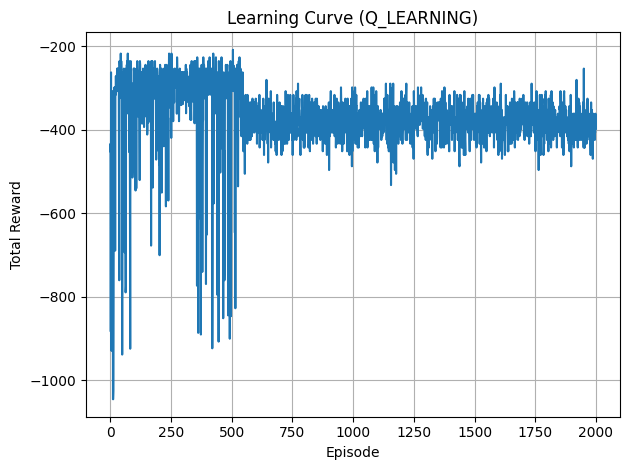
# ML Lab 11

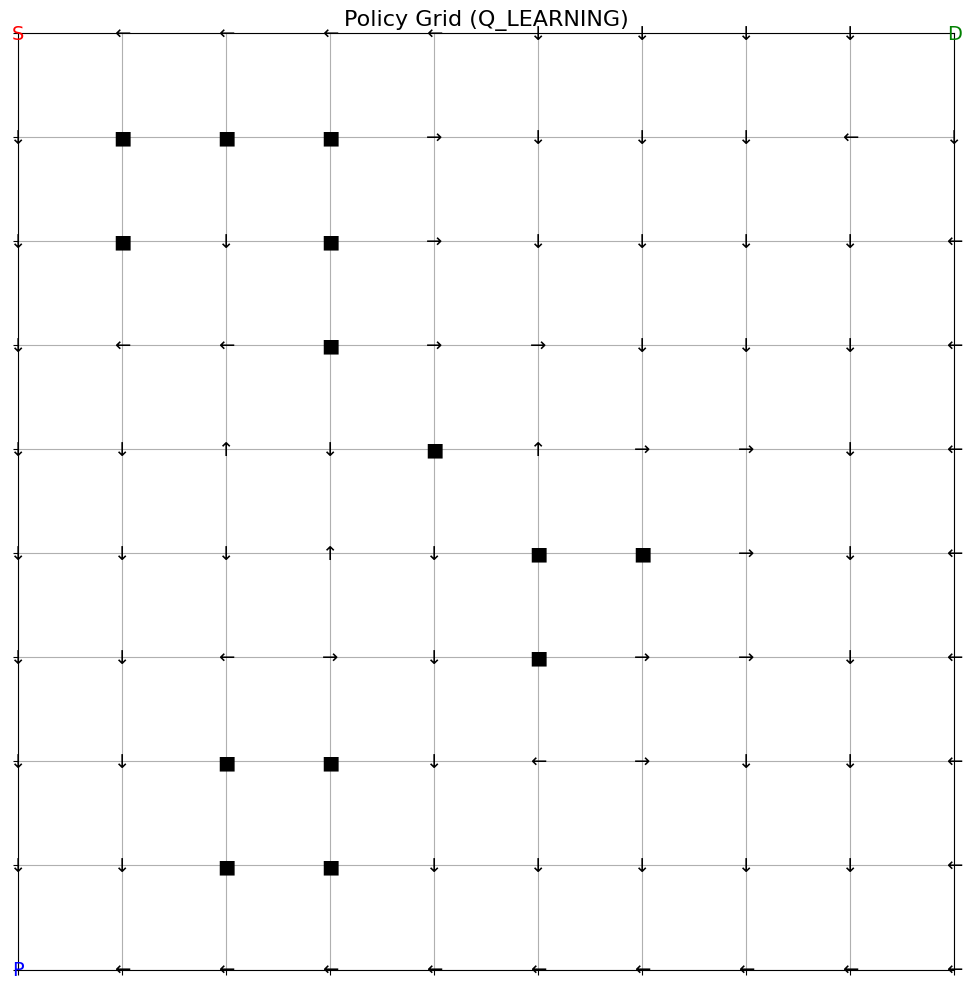
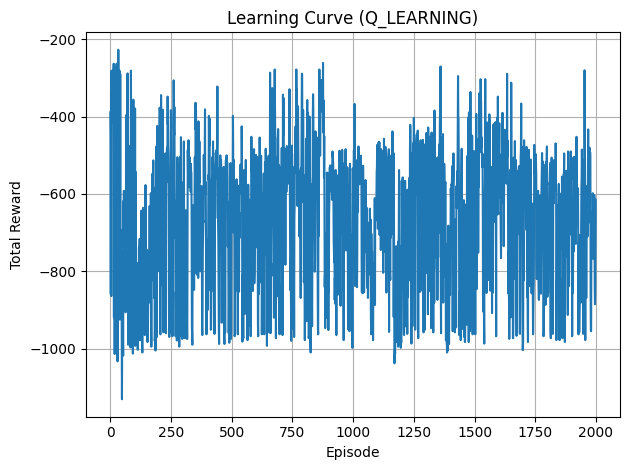
## Investigating Parameter Sensitivity

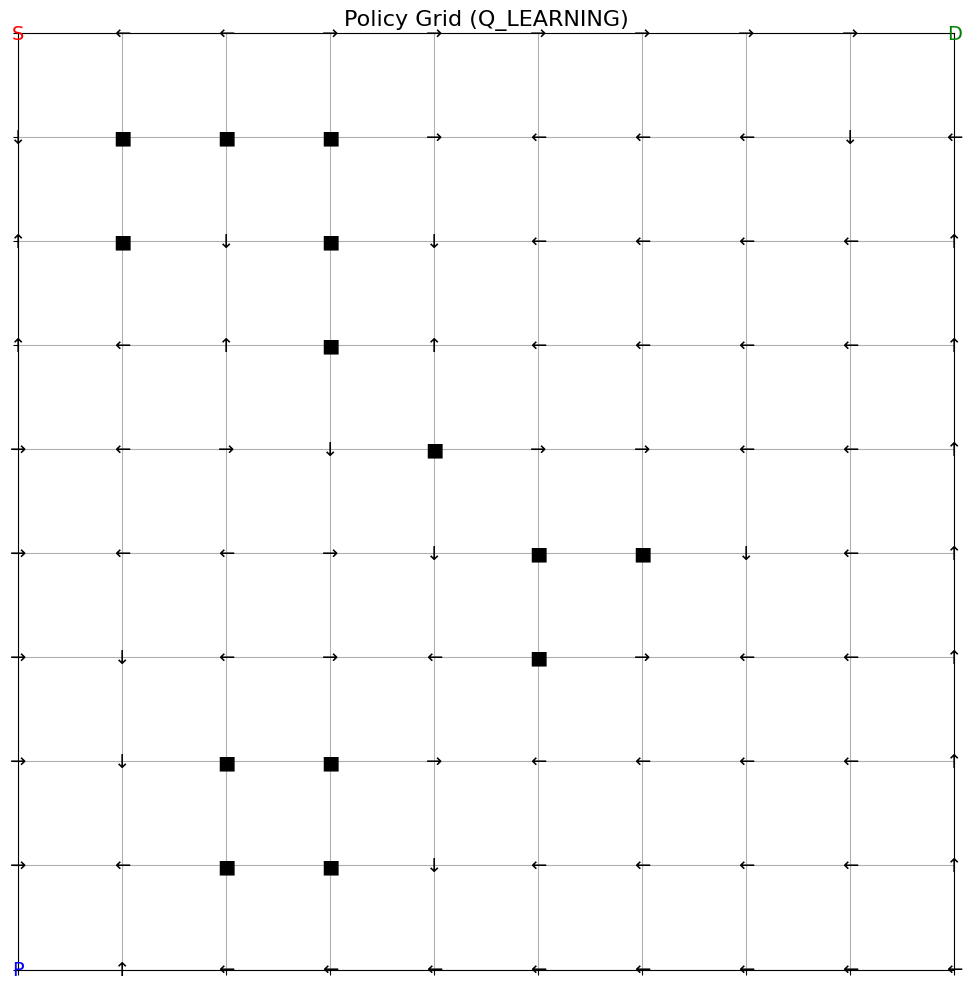
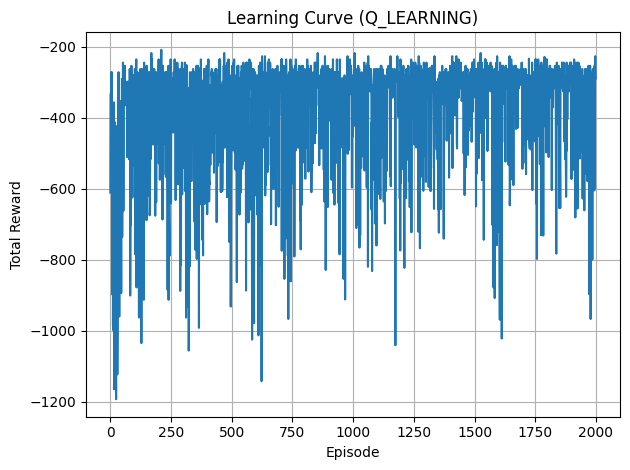
### Learning Rate 0.01 and 0.5



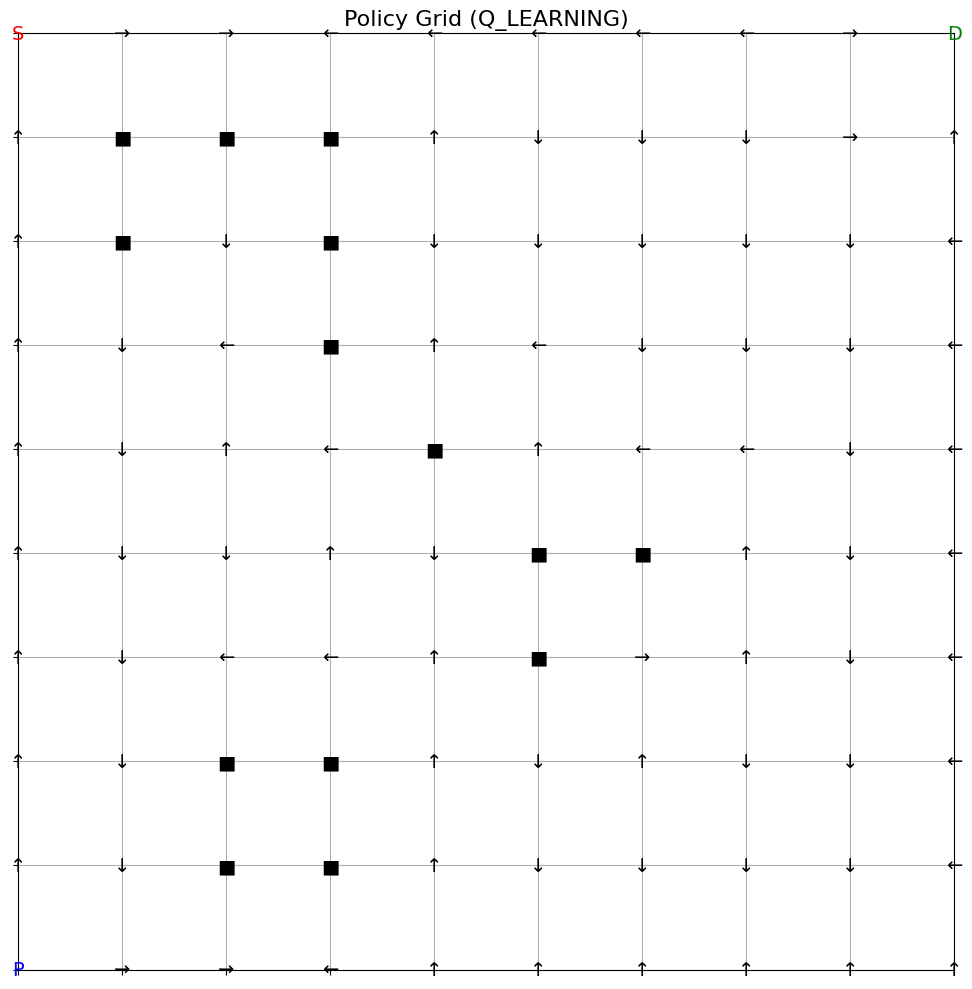
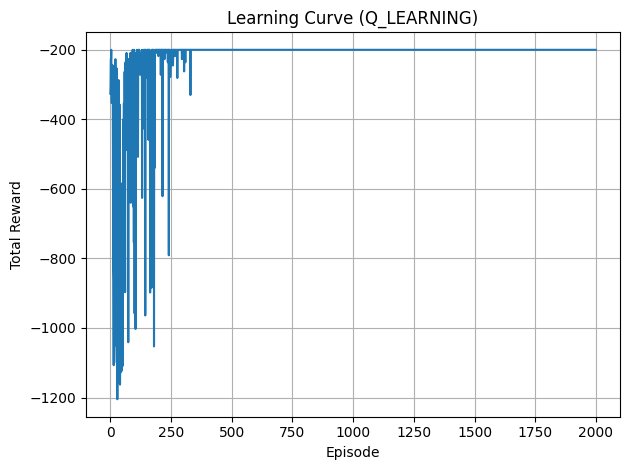


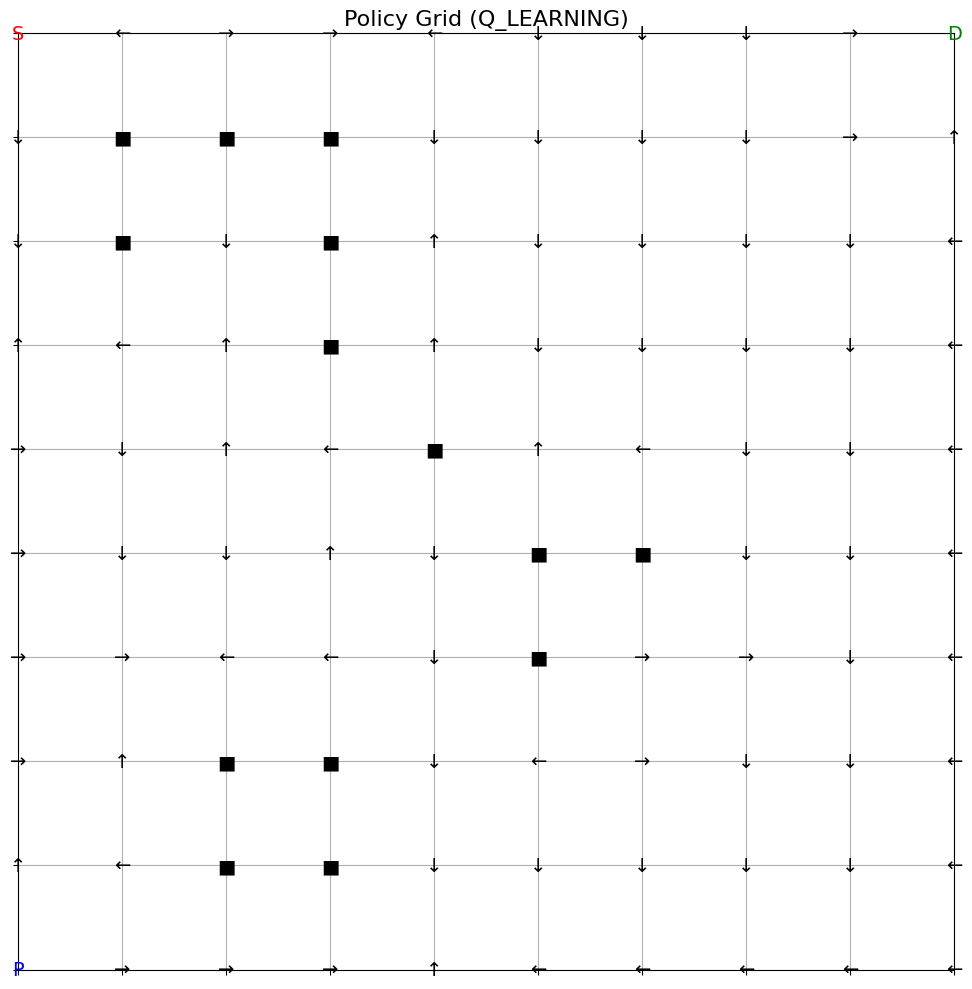
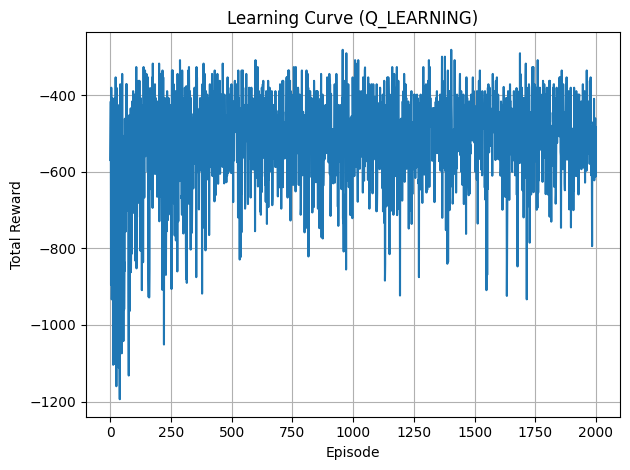
### Discount Factor 0.5 and 0.99



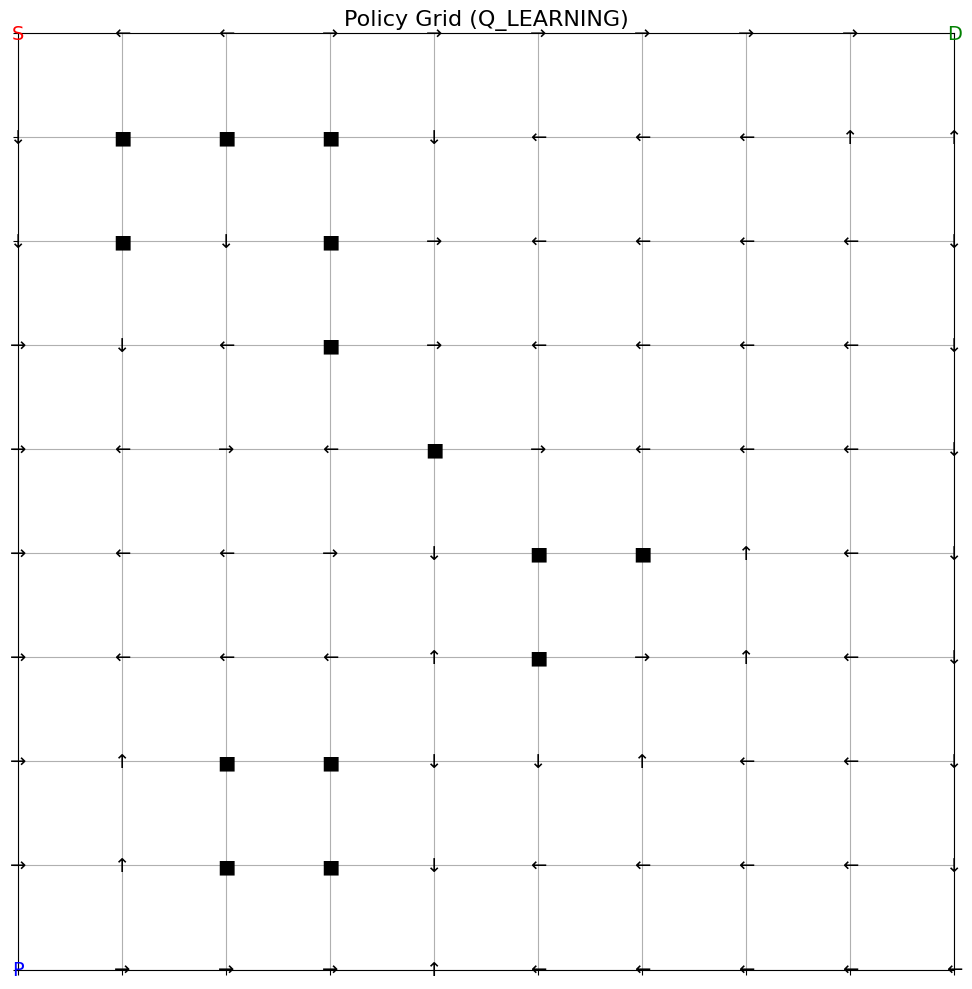
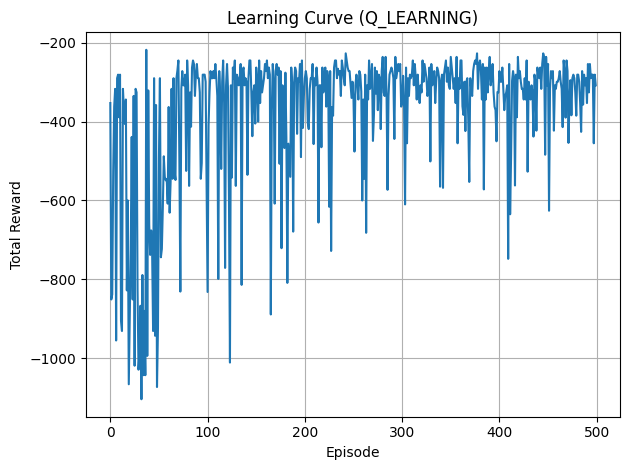


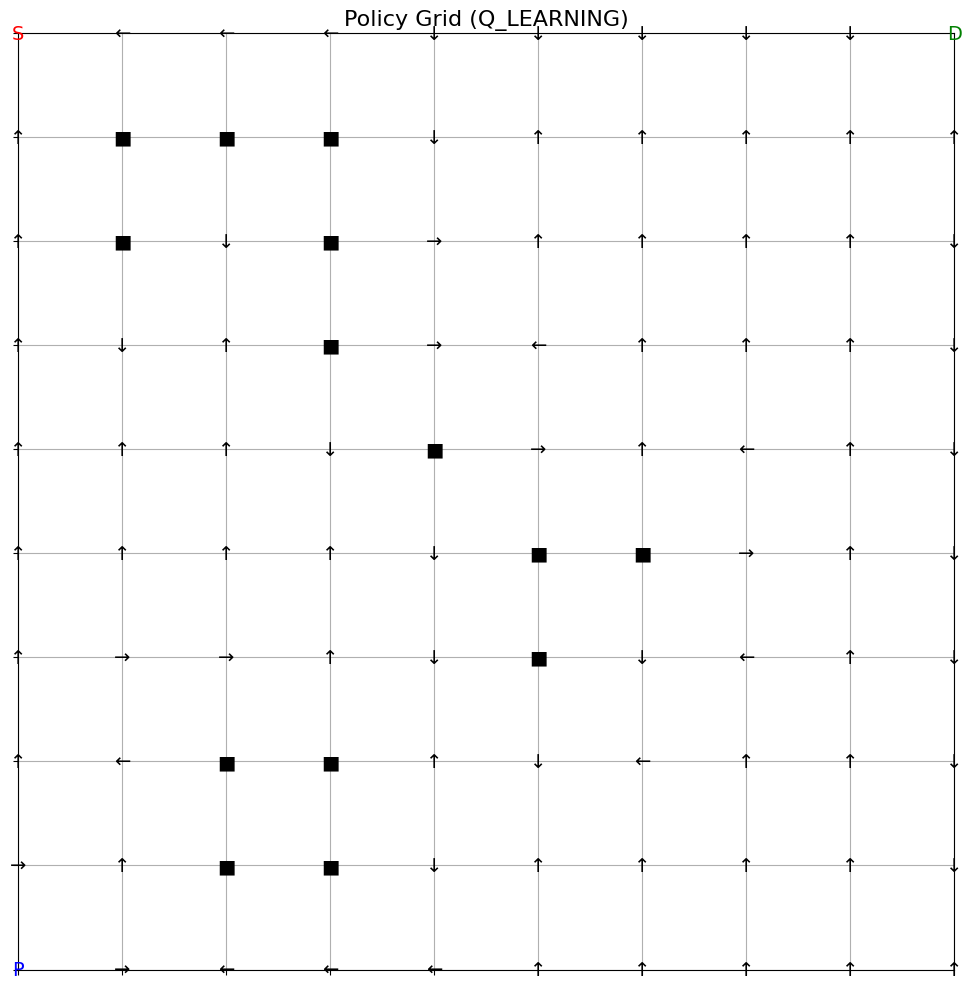
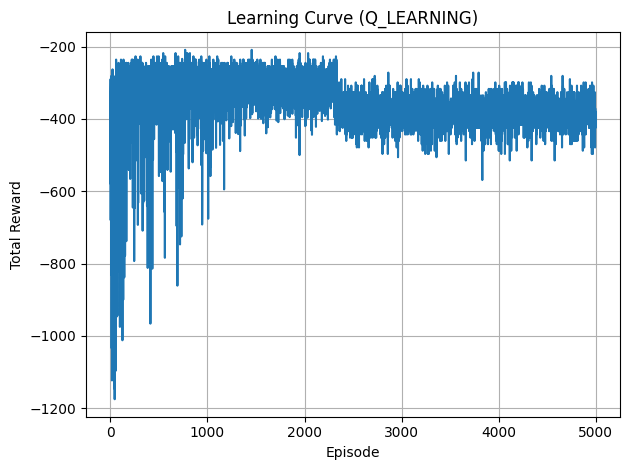
### Exploration Rate 0.0 and 0.5



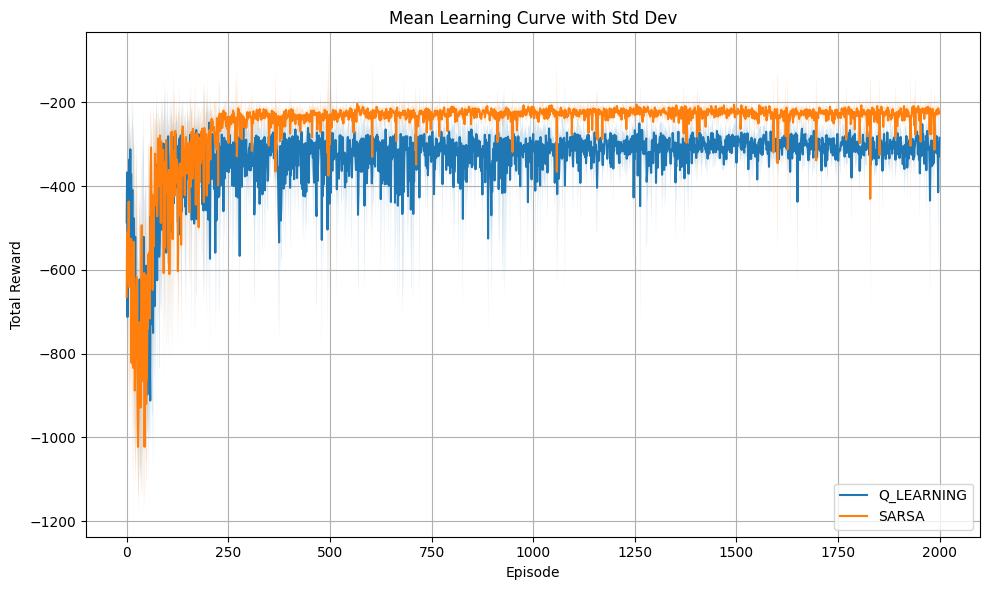


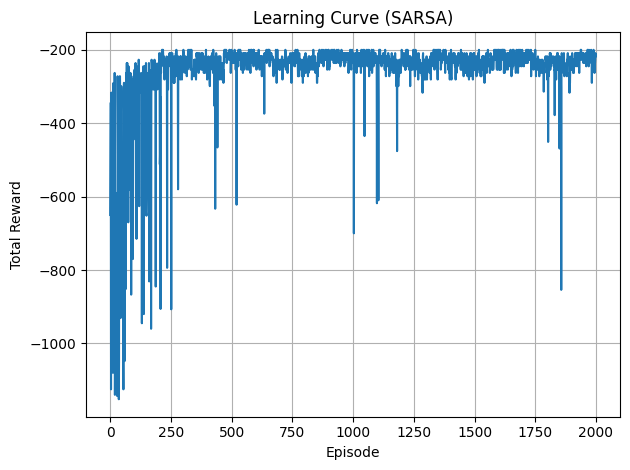
Number of Episodes 500 and 5000





## Compare Q-Learning and SARSA





## Analyze partial observability and state design

### How is the agent’s state represented without a global position?

The agent doesn't have direct access to its absolute position (x, y) on the grid. Instead, it relies on a local view and an additional carrying state. This state abstraction enables the agent to perceive the environment based on local observations without knowing its exact global (x, y) position. The state is thus represented as the tuple.

### Why does Q-learning still apply despite limited observations?

* The robot can still encounter repeated local views over time, even if it doesn't know its exact position. This allows Q-learning to learn reliable state-action values (Q-values) for those views.
* Q-learning optimizes the expected long-term reward from an action, not the immediate reward. Even with partial information, if the agent encounters similar state-action pairs, it can still learn the value of actions, based on future expected rewards.
* Through the exploration mechanism (via ε-greedy), Q-learning can still eventually cover all relevant states and actions enough time to learn optimal behavior. The robot, even with local views, can discover how to navigate the grid and collect/drop items over time.

### How does state abstraction via local views affect learning?

Positive Impacts

* Faster Exploration - The agent doesn't need to learn the exact (x, y) coordinates of every position. Instead, it only learns how to navigate based on its local context. This simplifies the state space and allows the agent to focus on relevant information.
* Generalization - If the agent encounters similar local views multiple times in different areas of the grid, it can generalize its knowledge across those views. This can result in a more robust policy that works across different grid regions.

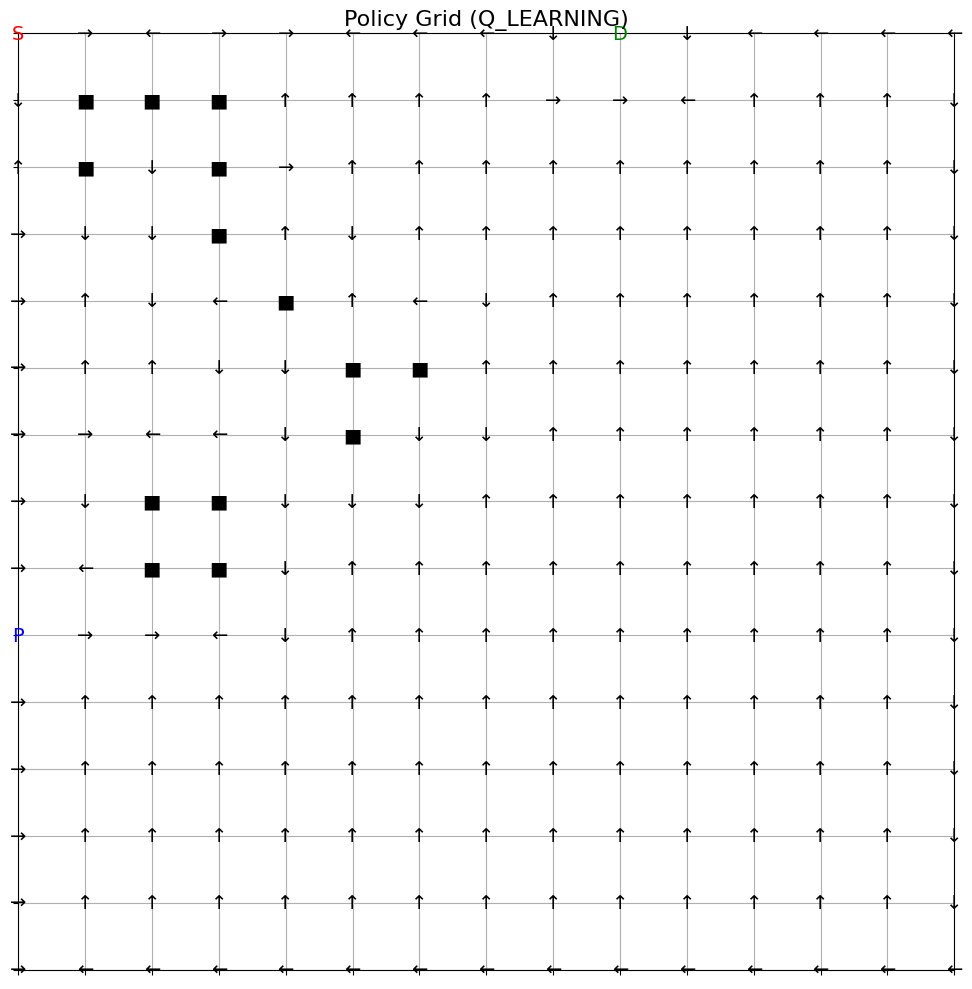
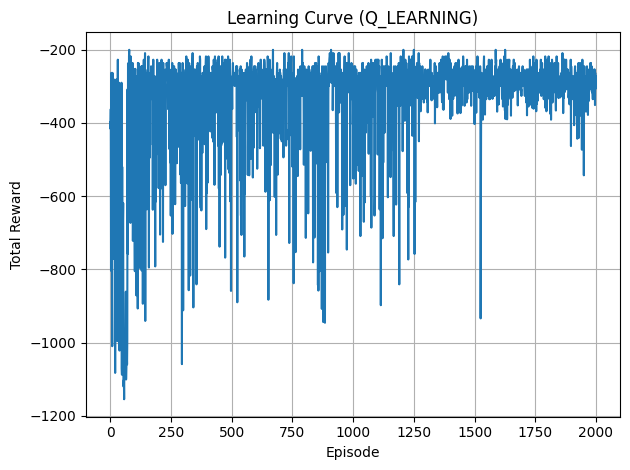
Challenges

* Loss of Spatial Information: The agent no longer has access to its absolute location on the grid. It has to rely entirely on the local view, which can make it harder to understand the global context.
* Increased Complexity in Learning: Since the robot cannot see the full grid, it might take longer to learn optimal actions because it has to figure out how to navigate based only on partial information. For example, if the local view is the same in two different locations, the agent may need additional exploration to figure out whether it's in a safe or dangerous region of the grid.

## Environmental modifications

* The learning speed slowed down with larger grid sizes and more obstacles.
* The stability of the learned policy fluctuated more as the agent has more paths and options to consider.

### Grid Size 15X15



### Grid Size 20X20

