1. Comparison of Execution Time and Convergence
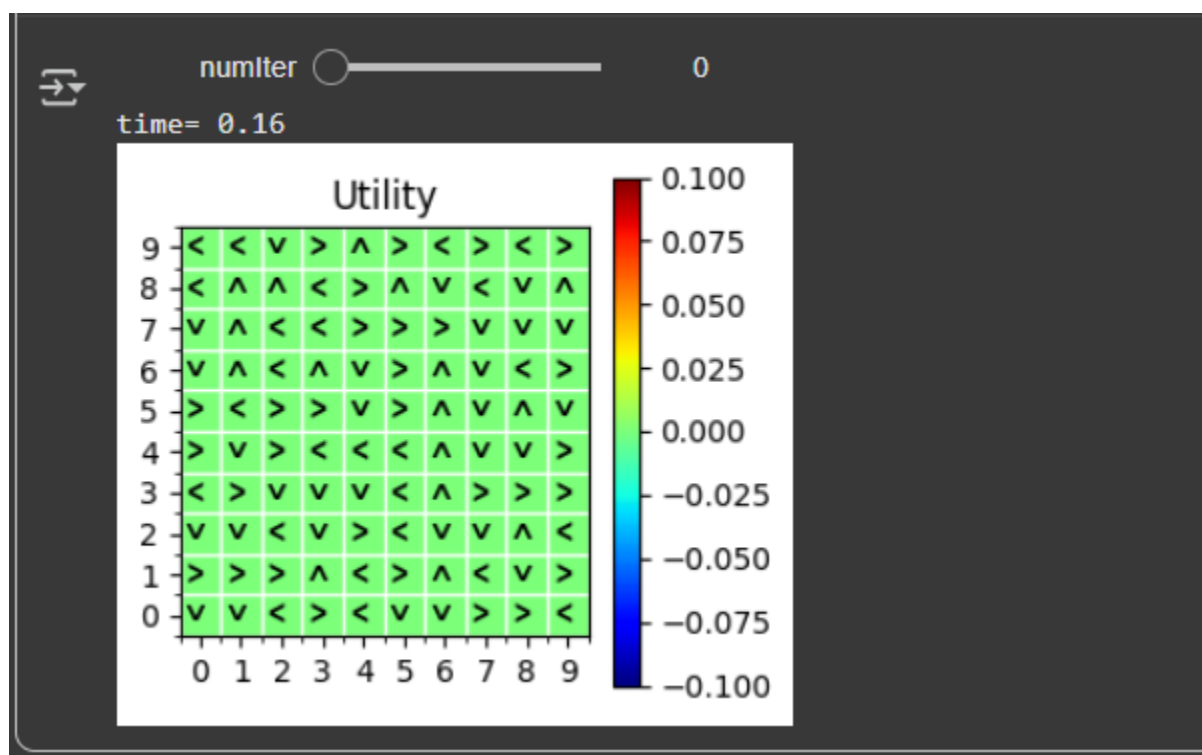
- **Model-Based (Policy Iteration / Value Iteration):**
    - Execution Time: These methods can be computationally expensive per iteration because they require sweeping through all states and actions and using the full transition probability model.
    - Convergence: They generally converge faster (fewer iterations) since they exploit the exact environment dynamics (state transition probabilities and rewards).
    - Overall: High per-iteration cost but fewer iterations → efficient when the model is known and small/medium state space.
- **Model-Free (Q-Learning):**
    - Execution Time: Each step is relatively cheap since it only updates one state-action pair based on sampled experience.
    - Convergence: Convergence is slower and depends on exploration strategy, learning rate, and number of episodes. It requires significantly more interactions with the environment to approximate the optimal policy.
    - Overall: Low per-step cost but requires many more steps → efficient in large/unknown environments but slower to converge.

---

2. Difference Between Model-Based and Model-Free Algorithms

- **Model-Based:**
    - The agent has or learns a model of the environment (transition probabilities + rewards).
    - Uses this model to plan ahead (simulate outcomes) and derive optimal policies.
    - Example: Value Iteration, Policy Iteration.
    - Strength: Faster convergence when the model is accurate.
    - Weakness: Requires complete knowledge of the environment, which is often unrealistic.
- **Model-Free:**
    - The agent does not use an explicit model of the environment.
    - Learns policies or value functions directly from experience through trial and error.
    - Example: Q-Learning, SARSA.
    - Strength: Works even when the environment's dynamics are unknown.
    - Weakness: Slower convergence, depends heavily on exploration.

z

numIter ⊙———————————— 0

time= 0.16

Utility

	0	1	2	3	4	5	6	7	8	9
9	<	<	v	>	∧	>	<	>	<	>
8	<	∧	∧	<	>	∧	v	<	v	∧
7	v	∧	<	<	>	>	>	v	v	v
6	v	∧	<	∧	v	>	∧	v	<	>
5	>	<	>	>	v	>	∧	v	∧	v
4	>	v	>	<	<	<	∧	v	v	>
3	<	>	v	v	v	<	∧	>	>	>
2	v	v	<	v	>	<	v	v	∧	<
1	>	>	>	∧	<	>	∧	<	v	>
0	v	v	<	>	<	v	v	>	>	<

```
                     -1.2573789120002998
        RIGHT
                     -1.2008501840271
    (15, 2)
        UP
                     -1.2005316246233002
        DOWN
                     -1.1974406759823002
        LEFT
                     -1.2289827653896002
        RIGHT
                     -1.2019512662436
    (15, 3)
        UP
                     -1.2012050957015998
        DOWN
                     -1.1939245216572
        LEFT
                     -1.2090570178755002
        RIGHT
```