```
import gym
import numpy as np
import random

# Initialize the FrozenLake environment
env = gym.make("FrozenLake-v1", is_slippery=False)

# Q-learning parameters
alpha = 0.1  # Learning rate
gamma = 0.99  # Discount factor
epsilon = 1.0  # Exploration rate
epsilon_min = 0.1  # Minimum exploration rate
epsilon_decay = 0.995  # Decay rate for exploration probability

# Initialize the Q-table
q_table = np.zeros((env.observation_space.n, env.action_space.n))

# Training parameters
num_episodes = 1000
max_steps_per_episode = 100

# Q-learning algorithm
for episode in range(num_episodes):
    state = env.reset()
    done = False
    step = 0
    total_reward = 0

    while not done and step < max_steps_per_episode:
        # Exploration-exploitation tradeoff
        if random.uniform(0, 1) < epsilon:
            action = env.action_space.sample()  # Explore
        else:
            action = np.argmax(q_table[state, :])  # Exploit

        # Take the action and observe the outcome
        next_state, reward, done, _ = env.step(action)

        # Update the Q-table
        old_value = q_table[state, action]
        next_max = np.max(q_table[next_state, :])
        new_value = (1 - alpha) * old_value + alpha * (reward + gamma * next_max)
        q_table[state, action] = new_value

        state = next_state
        step += 1
        total_reward += reward

    # Decay the exploration rate
    if epsilon > epsilon_min:
        epsilon *= epsilon_decay

    if (episode + 1) % 100 == 0:
        print(f'Episode {episode + 1}/{num_episodes} - Total reward: {total_reward} - Epsilon: {epsilon}')
        print(f'Q-table snapshot:\n{q_table}')

# Evaluate the agent
num_eval_episodes = 100
total_rewards = 0

for episode in range(num_eval_episodes):
    state = env.reset()
    done = False
    step = 0
    episode_reward = 0

    while not done and step < max_steps_per_episode:
        action = np.argmax(q_table[state, :])  # Always exploit during evaluation
        next_state, reward, done, _ = env.step(action)
        episode_reward += reward
        state = next_state
        step += 1

    total_rewards += episode_reward

average_reward = total_rewards / num_eval_episodes
print(f'Average reward over {num_eval_episodes} evaluation episodes: {average_reward}')

env.close()
```

```
 [3.00984444e-01 4.63133002e-02 9.76380016e-01 0.00000000e+00]
 [8.96077280e-02 9.89357301e-01 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 4.42146309e-02 5.00856645e-01 0.00000000e+00]
 [1.05385835e-01 1.87311993e-01 9.99923823e-01 4.49784855e-01]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]]
Episode 500/1000 - Total reward: 1.0 - Epsilon: 0.0996820918179746
Q-table snapshot:
[[2.57967746e-01 9.50917648e-01 9.59888291e-04 4.25477197e-01]
 [7.78714740e-02 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [2.69194204e-01 9.60574965e-01 0.00000000e+00 3.84676614e-01]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [1.71747383e-01 0.00000000e+00 9.70293865e-01 2.43452026e-01]
 [4.79013518e-01 1.45283853e-01 9.80098982e-01 0.00000000e+00]
 [5.02068648e-01 9.89999839e-01 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 4.42146309e-02 6.33413939e-01 9.70102057e-02]
 [1.84416127e-01 4.04834911e-01 9.99999991e-01 5.02815773e-01]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]]
Episode 600/1000 - Total reward: 1.0 - Epsilon: 0.0996820918179746
Q-table snapshot:
[[0.44319831 0.95099002 0.12310583 0.523514  ]
 [0.48251959 0.         0.         0.        ]
 [0.02395337 0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.45396019 0.960596   0.         0.49046894]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.3855252  0.         0.970299   0.57497446]
 [0.47901352 0.3847749  0.9801     0.        ]
 [0.59103233 0.99       0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.04421463 0.80049555 0.09701021]
 [0.27840679 0.51601628 1.         0.59349976]
 [0.         0.         0.         0.        ]]
Episode 700/1000 - Total reward: 1.0 - Epsilon: 0.0996820918179746
Q-table snapshot:
[[0.53787186 0.95099005 0.30532645 0.66725256]
 [0.67046853 0.         0.         0.        ]
 [0.02395337 0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.50366317 0.96059601 0.         0.57616107]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.58329205 0.         0.970299   0.67687469]
 [0.57051419 0.47573037 0.9801     0.        ]
 [0.663093   0.99       0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
```