The Assignment defines a class `QTable` which is used to implement Q-learning, a form of reinforcement learning. The `QTable` is designed to interact with an environment (`env`) which is passed as a parameter when an instance of the class is created. This environment defines the grid space through `x_size` and `y_size`, and the types of states in each grid cell (`' +-'` or spaces). The QTable itself is initialised as a dictionary that maps each possible combination of position (`x`, `y`) and action to a Q-value, which represents the expected utility of performing a given action from a given state. Only valid grid positions (not blocked or obstacles) are initialised to a Q-value of 0.0. The class methods include functions to get or set the Q-values for specific states and actions, retrieve a row of Q-values for a given state, and perform a learning episode where the agent updates Q-values based on the rewards received and the estimated future rewards.

The learning process in `QTable` is driven by the methods `learn_episode` and `learn`. In `learn_episode`, the method performs a single episode of Q-learning starting from a random state. It continues to choose actions randomly from legal actions available until it reaches a terminal state or runs out of legal actions. For each action taken, it updates the Q-value based on the reward received and the maximum future Q-value, using the learning rate (`alpha`) and discount factor (`gamma`). This update formula adjusts the old Q-value towards a new value that accounts for the immediate reward and the discounted maximum future rewards. The `learn` method simply runs multiple learning episodes (`episodes` times), thus allowing the agent to gradually improve its policy by learning from the outcomes of its actions. The class also includes a `__str__` method to return a formatted string representation of the Q-table, which can be useful for debugging and visualisation purposes.

```
(base) surendrapothuri@SURENDRAs-MacBook-Pro Code % sh run.sh learn
UP
-----   -----   -----   -----   -----   -----   -----
2.02    -----   -----   -----   4.25    3.83    -----
1.70    -----   -----   -----   5.38    5.60    -----
1.44    -----   0.49    -----   -----   -----   -----
1.20    -----   0.57    -----   -----   -----   -----

RIGHT
2.45    2.96    3.66    4.64    4.42    2.00    -----
-----   -----   -----   -----   6.32    -9.82   -----
-----   -----   -----   -----   8.01    9.58    -----
-----   -----   -----   -----   -----   -----   -----
0.79    0.67    0.55    0.42    0.31    0.26    -----

DOWN
1.64    -----   -----   -----   5.72    5.60    -9.66
1.37    -----   -----   -----   6.66    8.05    -----
1.21    -----   0.59    -----   -----   -----   -----
0.95    -----   0.68    -----   -----   -----   -----
-----   -----   -----   -----   -----   -----   -----

LEFT
-----   2.03    2.28    2.64    3.14    3.91    3.12
-----   -----   -----   -----   -----   4.07    -----
-----   -----   -----   -----   -----   6.29    -----
-----   -----   -----   -----   -----   -----   -----
-----   0.97    0.81    0.67    0.54    0.40    0.32
```