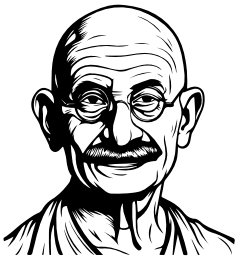
learnwithbhawana

Reinforcement Learning (Gandhiji- Style)



Let’s go **line by line**, code explanation for each part of your code:

- **import random**

  Explanation: Imports Python’s random module to pick actions randomly.

  (“Randomly choose actions like Gandhi trying different ways.”)

- **actions = ["violence", "non_violence"]**

  Explanation: List of possible actions Gandhi can take.

  Gandhi can either choose violence or non-violence.”

- **q_values = {"violence": 0, "non_violence": 0}**

  Explanation**:** Initialize Q-values (learning scores) for each action as 0.

“Start with zero knowledge about which action is better.”

- **Code:**

```
def get_reward(action):
  if action == "non_violence":
    return 10
  else:
    return -10
```

Explanation**:** Reward system: non-violence gives +10, violence gives -10.

“Good actions get points, bad actions lose points.”

- **print("🕊️ Training Gandhi-Style RL Model...\n")**

Explanation: Print a message to show training has started.

- **Code**

```
for episode in range(10):
  action = random.choice(actions)  # try actions randomly (trial and error)
  reward = get_reward(action)      # receive feedback
  q_values[action] = q_values[action] + 0.5 * (reward - q_values[action])  # learn
  print(f"Episode {episode + 1}: Action = {action}, Reward = {reward}")
```

Explanation: For 10 tries: pick an action randomly, get its reward, update knowledge (Q-value) using simple learning rule, and print results.

“Try actions, see if they work, learn a little each time.”

- **Code:**

```
print("\n✅ Final Learned Q-Values:", q_values)
best_action = max(q_values, key=q_values.get)
print(f"\nGandhi-style RL ke hisaab se best policy: '{best_action}'")
```

Explanation: Show the final learned scores and pick the best action (highest Q-value).

“After learning, Gandhi knows the best action is... non-violence!”

💡 **Overall summary in one line:**

“This code teaches Gandhi (with trial and error) which action—violence or non-violence—gives the best reward using a simple reinforcement learning approach.”