Part B: Reinforcement Learning (RL) + Documentation

1. Integration of RL Logic

A Reinforcement Learning (RL) agent, specifically using the Proximal Policy Optimization (PPO) algorithm, was integrated to automate decision-making within the DSS. The `gym` environment provides a simulated interface for the RL agent to learn from the anomaly scores produced by the Part A pipeline.

2. RL Component Breakdown

- **Environment:** The `AnomalyEnv` class simulates the DSS, providing a state and reward to the agent at each step.
- **State Space:** A continuous one-dimensional space (`Box(low=0.0, high=1.0)`) representing the normalized anomaly score from the LSTM autoencoder.
- **Action Space:** A discrete space with three possible actions: ○ **Action 0:** `Ignore` the alert.
    - **Action 1:** `Log` the event (low-priority response).
    - **Action 2:** Initiate a `Cooldown` procedure (high-priority, corrective action).
- **Reward Function:** The reward function encourages the agent to take the correct action.
    - **High Anomaly (`score > 0.05`):** High reward for corrective actions (`+1`), a small reward for logging (`+0.5`), and a penalty for ignoring (`-1`).
    - **Low Anomaly (`score <= 0.05`):** High reward for ignoring (`+1`), and a penalty for taking action (`-0.5`).

3. Proof-of-Concept & Results

The PPO agent was trained on the `AnomalyEnv` for 5000 timesteps. The training was successful, with the agent learning to differentiate between high and low anomalies and select the optimal corresponding action.

The final output, `dss_actions_output.csv`, demonstrates a working proof-of-concept where the PPO agent's decisions are recorded alongside the anomaly scores.

4. Final Performance Observations

The integrated pipeline successfully demonstrates an end-to-end anomaly detection and response system.

- The LSTM-based detector provides a continuous anomaly score, which acts as the foundation for the DSS.
- The PPO agent learns a simple but effective policy, showing that an RL-based system can be trained to make nuanced decisions (e.g., distinguishing between logging and a full corrective action) based on the severity of a detected anomaly.
- The entire system is modular, with clear interfaces between the data, the anomaly detector, and the RL-powered DSS, fulfilling the requirements for both Part A and Part B.