Distributed AI Model Training (Simplified Simulation)

➢ **Objective:**

Implement a mini simulation of Federated Learning where multiple nodes train partial models on subsets of data and aggregate results.

➢ **Requirements:**

• Dataset: Use the Iris dataset (from scikit-learn). Split into 3–4 subsets to simulate different nodes.
• Model Training: Each node trains a logistic regression model (use only scikit-learn).
• Aggregation: Implement a simple Federated Averaging (FedAvg) function to combine model weights.
• Simulation Features: Randomly drop one node in some rounds (skip its update). When the node recovers, include it again in aggregation.
• Comparison: Train a centralized logistic regression (on full dataset) and compare accuracy vs distributed training.
• Logging: Print logs for each round: node updates, aggregation, node failures/recoveries.

➢ **Optional (Bonus):**

• Plot accuracy over training rounds.
• Simulate different data per node (non-iid split).

➢ **Expected Deliverables:**

• Python code simulating distributed nodes.
• FedAvg aggregation implementation.
• Console logs showing training + aggregation + node failures.
• Accuracy comparison (centralized vs distributed).

➢ **Evaluation Criteria:**

• Correctness of distributed simulation.
• Code clarity and readability.
• Handling of node failures.
• Accuracy comparison.