**Prompt:**

Implement Iris classification using the CRISP-DM framework with **tf.keras**, **PyTorch**, and **PyTorch Lightning**, integrating **TensorBoard** for visualization.

* **Business Understanding**: Classify Iris species based on four features with high accuracy.
* **Data Understanding**: Use the Iris dataset with three classes and four numeric features.
* **Data Preparation**: Load and standardize data using sklearn.datasets. Convert to formats suitable for each framework.
* **Modeling**:
  1. **tf.keras**: Build a Sequential model with Dense layers, train with fit, and visualize with TensorBoard callbacks.
  2. **PyTorch**: Define nn.Module, train using loss and optimizer, and log metrics with SummaryWriter.
  3. **PyTorch Lightning**: Create LightningModule and LightningDataModule, train with Trainer, and log with TensorBoardLogger.
* **Evaluation**: Compare accuracy and metrics across all implementations.
* **Visualization**: Use TensorBoard to track training dynamics (loss and accuracy) for all models.

**Goal**: Demonstrate framework-specific approaches while maintaining consistency in data handling, training, and evaluation.