**Task-2**

**Predictive Maintenance System**

**Data Collection**

Predictive maintenance relies on historical data from equipment, which can include:

* **Sensor Data**: Measurements from various sensors installed on the machinery (e.g., temperature, vibration).
* **Maintenance Logs**: Records of past maintenance activities.
* **Failure Records**: Historical data on when and why equipment failures occurred.

### ****Data Preprocessing****

**Data Cleaning**: This involves handling missing values, removing duplicates, and correcting errors in the data.

**Feature Engineering**: This step involves transforming raw data into features that will help the model make predictions. For example, you might calculate rolling averages, differences between sensor readings, or create new features that capture the time since the last maintenance.

### ****Feature Engineering****

**Feature Selection**: Choose which features (columns) to use for the model. In our example, features like sensor1, sensor2, sensor3, and usage\_hours were used directly.

**Creating New Features**: In more complex scenarios, you might derive new features that could improve model performance. For example, if you have multiple sensors, you might calculate their mean or variance.

**Model Selection**

**Choosing a Model**: Various machine learning algorithms can be used for predictive maintenance:

* **Logistic Regression**: Good for binary classification problems.
* **Decision Trees**: Simple and interpretable.
* **Random Forest**: An ensemble method that improves prediction by averaging multiple decision trees.
* **Gradient Boosting Machines (GBM)** and **Neural Networks**: More complex models that can capture intricate patterns in the data.

**Model Training and Evaluation**

**Training**: This involves feeding the model with historical data so it can learn patterns that indicate equipment failure. The model adjusts its parameters to best fit the training data.

**Evaluation**: After training, evaluate the model's performance using a test set (data the model hasn't seen before). Common metrics for evaluation include:

* **Confusion Matrix**: Shows the number of true positives, false positives, true negatives, and false negatives.
* **Classification Report**: Provides precision, recall, and F1-score, which are useful for understanding the model's performance in detail.

### ****Deployment****

**Real-Time Predictions**: Once the model is trained and evaluated, it can be deployed to make predictions on new, real-time data. For example, when new sensor readings come in, the model can predict whether the equipment is likely to fail soon.