What This Code Does:

1. **Imports Required Libraries:**
   - **rdkit for chemical descriptor calculations.**
   - **torch for building and training a neural network.**
   - **matplotlib, numpy, pandas for data processing and visualization.**
2. **Defines Chemical Components:**
   - **Uses SMILES notation to represent Pigment (TiO_2), Extender ($CaCO_3$), and Binder (PVA).**
   - **Computes molecular descriptors (e.g., Molecular Weight, LogP, TPSA) using RDKit.**
3. **Loads Experimental Dataset:**
   - **Contains Pigment, Extender, Binder amounts and their corresponding Viscosity, Density, and pH values.**
4. **Feature Engineering:**
   - **Generates new chemical features by combining molecular descriptors and ingredient proportions.**
   - **Includes cross-component interactions to improve prediction accuracy.**
5. **Data Preprocessing:**
   - **Splits dataset into training (80%) and test (20%) sets.**
   - **Normalizes features using StandardScaler.**
   - **Converts data into PyTorch tensors for model training.**
6. **Builds a Neural Network:**
   - **4-layer architecture: Input → 32 neurons (ReLU) → 16 neurons (ReLU) → 8 neurons (ReLU) → Output (3 neurons for Viscosity, Density, pH).**
   - **Uses Dropout (0.3) to prevent overfitting.**
   - **Optimized with Adam optimizer & MSE loss.**
7. **Trains the Model:**
   - **Runs for up to 5000 epochs, with early stopping if validation loss doesn't improve for 100 epochs.**
8. **Evaluates & Visualizes Results:**
   - **Plots training vs validation loss.**
   - **Compares actual vs predicted values for each target property.**
9. **Feature Importance Analysis:**

  - **Uses permutation importance to determine which features impact predictions the most.**

10. **Predicts Properties for New Data:**

- **Uses the trained model to predict Viscosity, Density, and pH for new formulations.**