# Customer Lifetime Value (CLV) Prediction

## Overview

Customer Lifetime Value (CLV) predicts the total revenue a business can expect from a customer during their relationship. This metric is crucial for businesses to understand customer profitability and optimize marketing strategies.

## Steps in the Code

### 1. Importing Libraries

Essential Python libraries are imported for data manipulation, visualization, and modeling:  
- pandas and numpy: For data manipulation and numerical calculations.  
- matplotlib and seaborn: For data visualization.  
- sklearn: For data preprocessing, splitting, and machine learning modeling.

### 2. Loading the Dataset

The dataset, Online Retail.xlsx, is loaded using pandas. The head() method shows the first five rows, providing a snapshot of the data, including fields like:  
- InvoiceNo: Unique identifier for transactions.  
- Description: Product name.  
- Quantity: Items purchased in a transaction.  
- UnitPrice: Price per unit.  
- CustomerID: Unique identifier for customers.  
- InvoiceDate: Transaction timestamp.

### 3. Data Cleaning

To ensure reliable predictions:  
- Rows with missing CustomerID or essential fields are removed.  
- Negative quantities (canceled orders) are excluded.  
- InvoiceDate is converted to datetime format.  
- A new column, TotalSpend, is calculated as Quantity × UnitPrice.

### 4. Feature Engineering (RFM Calculation)

The Recency, Frequency, and Monetary (RFM) features are derived:  
- Recency: Days since the last purchase.  
- Frequency: Number of transactions per customer.  
- Monetary: Total spending per customer.  
  
These features are computed by grouping the data by CustomerID.

### 5. Preparing Data for Modeling

Features (X): Recency, Frequency, and Monetary.  
Target Variable (y): Total spending (Monetary) is assumed to represent CLV.  
Data is split into training and test sets (80/20 split).  
Features are standardized using StandardScaler.

### 6. Training Models

Two machine learning models are trained:  
1. Linear Regression: A simple model assuming a linear relationship between features and target.  
2. Random Forest Regressor: An ensemble method that captures non-linear relationships.

### 7. Model Evaluation

Performance is evaluated using the test set:  
- Metrics:  
 - Mean Absolute Error (MAE): Average error magnitude.  
 - Mean Squared Error (MSE): Penalizes larger errors.  
 - R-squared (R²): Explains variance captured by the model.  
  
Results:  
- Linear Regression: Perfect fit (R² = 1.0), though likely due to overfitting.  
- Random Forest: Excellent performance (R² ~ 0.99) with slightly higher errors than Linear Regression.

### 8. Visualization

Scatter plots compare Actual vs Predicted CLV for both models:  
- A red dashed line represents a perfect prediction.  
- The closer the points are to this line, the better the model's predictions.