1. Understand the Goal

The main objective is to **predict potential car sales in India using insights from Japan's market data**.
You'll:

- Train a **classification model** on Japanese data (to predict whether someone will buy a car).
- Apply that model to Indian data to estimate sales potential.
- Validate & explain the model.
- Visualize sales trends using **Tableau**.

2. Step-by-Step Approach

Step 1 — Business Understanding

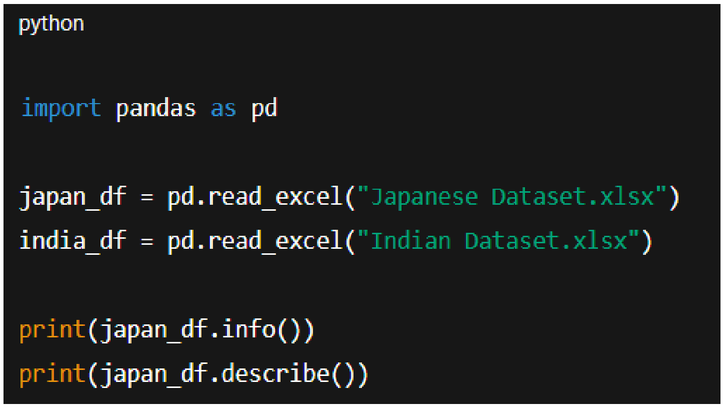
- **Objective:** Help ABG Motors decide if entering the Indian market will be profitable.
- **Target:** Predict **at least 12,000 sales in a year**.
- **Data:** Two datasets — one for Japan, one for India.
- **Prediction Goal:** Whether an individual will buy a car (Yes / No).

Step 2 — Data Understanding & Cleaning

Load both datasets (in Excel/CSV format) into Python (pandas) and:

- Check column names and meanings.
- Identify categorical and numerical variables.
- Check for **missing values**, **duplicates**, and **inconsistencies**.
- Explore **summary statistics** (mean, median, min, max, std).

Example:



```python
import pandas as pd

japan_df = pd.read_excel("Japanese Dataset.xlsx")
india_df = pd.read_excel("Indian Dataset.xlsx")

print(japan_df.info())
print(japan_df.describe())
```

Step 3 — Exploratory Data Analysis (EDA)

- **Visual trends:** (age, income, car ownership, etc. vs. purchase decision).
- **Check correlation** between features and the target.
- For categorical variables, use bar plots; for numerical, use histograms/box plots.

Example:

- In Japan: Maybe **income** and **marital status** are strong predictors.
- In India: Check if the same patterns exist.

Step 4 — Feature Engineering

- Encode categorical features (e.g., **OneHotEncoder** or pd.get_dummies).
- Scale numerical features if needed (StandardScaler / MinMaxScaler).
- Handle missing values (impute mean, median, or mode).
- Define **target variable** (e.g., BuyCar column with Yes=1, No=0).

Step 5 — Build the Classification Model

Since the output is **Yes/No**, you can start with:

- **Logistic Regression** (easy to interpret).
- Or try **Random Forest / Decision Tree** (better for non-linear patterns).

Process:

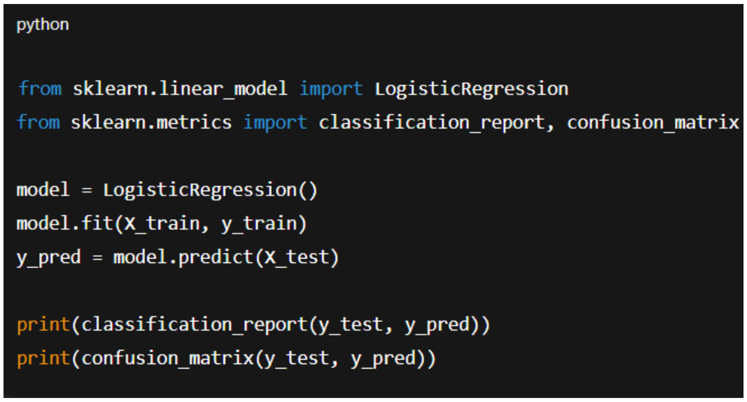
1. **Train-Test Split** on Japanese data (e.g., 70% train, 30% test).
2. Train the model.
3. Check coefficients (for Logistic Regression) or feature importances (for tree-based models).
4. **Justify decisions** — Why Logistic Regression? Why that split ratio? Why chosen features?

Step 6 — Model Validation & Performance

Evaluate with metrics:

- Accuracy
- Precision, Recall, F1-score
- ROC-AUC score
- Confusion Matrix

Example for Logistic Regression:



python

```python
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

model = LogisticRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

print(classification_report(y_test, y_pred))
print(confusion_matrix(y_test, y_pred))
```

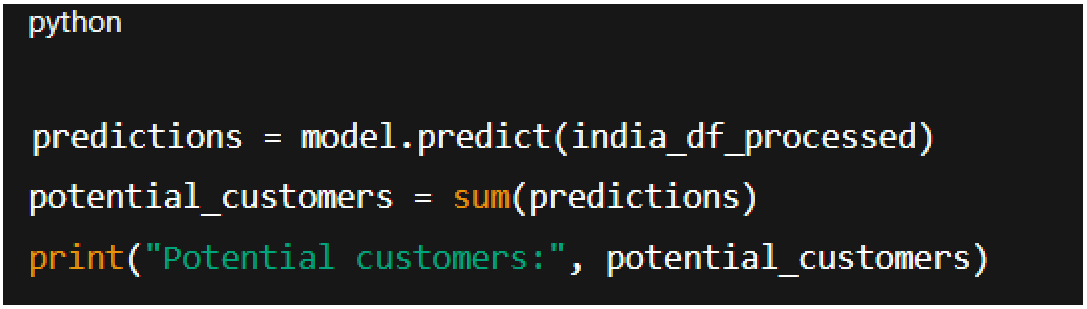
Step 7 — Business Interpretation

For Logistic Regression:

- Coefficients show how each feature affects the probability of buying a car.
    - Example: A coefficient of **+0.5 for income** → higher income increases purchase likelihood.
- Interpret in business terms so ABG Motors knows what factors matter most.

Step 8 — Apply Model to Indian Data

- Preprocess Indian data **the same way** as Japanese.
- Use the trained model to predict which Indian customers are likely to buy.
- Compare with the **12,000 sales target**.
- Count them:



python

```python
predictions = model.predict(india_df_processed)
potential_customers = sum(predictions)
print("Potential customers:", potential_customers)
```

Step 9 — Tableau Visualization

Create:

1. Sales trend comparison (Japan vs India).
2. Age group vs. Purchase likelihood.
3. Income level vs. Car purchase rate.
4. Pie chart of predicted buyers vs. non-buyers for India.

Step 10 — Final Report

Include:

- **Problem Statement** (why the analysis is done).
- **Data Cleaning & Preparation** steps.
- **EDA insights** (with graphs).
- **Model choice & justification**.
- **Performance metrics**.
- **Business interpretation of results**.
- **Recommendation**: Enter market or not.