# **📘 Assignment: FreshMart Data Pipeline Project**

### **Background**

You are working as a **Data Engineer** at **FreshMart**, a retail store. The company wants to build a simple **data pipeline** that ingests product data, processes it, stores it in a relational database, and ensures that the code is properly version-controlled on GitHub.

You are tasked with building the first version of this pipeline.

### **Tasks**

#### **Part A: Data Ingestion & Transformation with Pandas (30 mins)**

1. *Create a CSV FILE (*freshmart\_products.csv*) with at least 15–20 rows and columns: ProductID, ProductName, Category, Price, StockQuantity). And* load the provided CSV file (freshmart\_products.csv) into a Pandas DataFrame.
2. Perform transformations:  
   * Display the first 5 rows.
   * Clean the data (e.g., handle missing values or duplicates if any).
   * Create a new column StockValue = Price \* StockQuantity.
   * Aggregate the data by Category to compute:  
     + Average Price
     + Total StockQuantity

#### **Part B: Data Storage with PostgreSQL (35 mins)**

1. Establish a PostgreSQL connection using psycopg2.
2. Create a Products table with schema:  
   * ProductID (Primary Key, SERIAL)
   * ProductName (VARCHAR)
   * Category (VARCHAR)
   * Price (DECIMAL)
   * StockQuantity (INT)
3. Write a script to **load processed data from the Pandas DataFrame into the database** (ETL-style insert).
4. Query the database to fetch:  
   * All products in the "Dairy" category
   * Products with StockQuantity < 50
5. Return and display the results in Python.

#### **Part C: Version Control with GitHub (25 mins)**

1. Create a GitHub repository named **FreshMart-DataPipeline**.
2. Push the following:  
   * Your Pandas notebook (FreshMart\_ETL.ipynb)
   * Your PostgreSQL pipeline script (FreshMart\_DBLoad.py)
   * A README.md explaining your pipeline design (data ingestion → transformation → storage).
3. Ensure that your repository has a clear folder structure (e.g., /data, /scripts, /notebooks).

### **Deliverables**

* GitHub Repository Link
* Screenshot of transformed DataFrame (with StockValue column)
* Screenshot of PostgreSQL query results

### **Grading Criteria (100%)**

* Data ingestion & transformation: **30%**
* Database schema design & data loading: **40%**
* Version control & documentation: **30%**