**Financial Data Analysis for Banking Insights**

**Project Instructions for Interns**

1. **Follow the Medallion Architecture** for structured data processing.
2. **Apply consistent naming conventions** for all target files and SQL Server database tables.
3. **Ensure clean and structured coding practices** in Python scripts and SQL queries, especially for the Gold Layer.
4. **Automate the entire data pipeline**, including raw data ingestion, transformation, and final reporting.
5. **Implement auditing mechanisms** at each pipeline stage to track processing, failures, and transformations.
6. **Document the entire workflow**, including transformation steps, pipeline execution, and business logic.
7. **Use Git for version control**, storing all **code, SQL scripts, configurations, and relevant files**.
8. **Provide the Git repository URL** in the final documentation.

**Project Overview**

This project focuses on **analyzing financial data** to generate **insights for banking operations**, such as:

* **Transaction trends**
* **Customer spending behavior**
* **Risk assessment**

By leveraging **PySpark for large-scale data processing** and **Microsoft Fabric for efficient orchestration**, the project ensures **data quality, faster processing, and accurate reporting** through Power BI dashboards.

**Business Problem**

The current **manual data processing and reporting in Excel** is:

* **Time-consuming**
* **Prone to errors**
* **Lacking scalability for large datasets**

**Objective**

Build an **automated pipeline** to generate **structured input for Power BI**, allowing users to create **interactive, insightful visualizations efficiently**.

**Scope of Work (SOW)**

**1. Data Ingestion & Storage**

* Extract **financial data from Excel files (local system)**.
* Store data in **SQL Server**, which serves as the **primary structured storage system**.
* Use **Microsoft Fabric Data Factory** to move data from **SQL Server** into **OneLake storage**.

**2. Data Processing Using Medallion Architecture**

* **Bronze Layer:**
  + Store **raw data in its original format** in the Bronze Lakehouse.
  + Perform **basic data validation** to ensure completeness.
* **Silver Layer:**
  + **Cleanse, standardize, and transform data** to remove inconsistencies.
  + Perform **data quality checks** before proceeding to the Gold Layer.
* **Gold Layer:**
  + **Refine data** based on business requirements.
  + Store **final production-ready data as Delta tables in OneLake storage** for reporting.

**3. Data Visualization & Reporting**

* Establish a **direct connection from Power BI to the Gold Layer**.
* Develop **interactive dashboards** with at least **four reporting tabs** for analysis.
* Optimize **queries and table structures** for **faster Power BI performance**.

**4. Code & Documentation Standards**

* Maintain **clear documentation** explaining the **pipeline flow, transformations, and business logic**.
* Use **Git for version control**, ensuring all scripts, configurations, and SQL queries are stored properly.
* Follow **consistent naming conventions** for **SQL Server tables, scripts, and files**.

**Architecture**

1. **Data Ingestion**
   * Load **raw data from Excel into SQL Server**.
   * Use **Microsoft Fabric Data Factory** to move data from **SQL Server to OneLake storage**.
2. **Data Processing** (Medallion Architecture in Microsoft Fabric)
   * **Bronze Layer:** Store raw data.
   * **Silver Layer:** Clean and transform data.
   * **Gold Layer:** Refined data stored as **Delta tables in OneLake**.
3. **Data Visualization**
   * **Connect Power BI** to the **Gold Layer** and build interactive reports.

**Services / Key Requirements**

**Excel Data:** Financial datasets will be used.  
**SQL Server:** Used as the **structured storage system** before moving data to **Microsoft Fabric**.  
**Microsoft Fabric Subscription:** Access to **Fabric Data Factory and OneLake storage** for data orchestration.  
 **Power BI:** Desktop version required for visualization and dashboard creation.

**Value Addition**

**Automated pipeline triggering** (scheduled or event-based) for Power BI input.  
 **Power BI dashboard** with at least **four interactive reporting tabs**.

**Learning Outcomes**

**Hands-on experience with PySpark** for large-scale data processing.  
 **Practical skills in data loading, cleaning, and transformation** using PySpark.  
 **Experience implementing Medallion Architecture** for structured data processing.  
 **Ability to derive insights from large financial datasets.**