Project Report: Financial Credit Risk Data Warehousing Pipeline

1. Project Overview

The objective of this project was to design and deploy a production-grade ETL (Extract, Transform, Load) pipeline for a massive financial dataset. The system was engineered to handle high-volume data (2.2M+ records) with a focus on data integrity, automated error handling, and performance optimization. It transforms raw, unstructured loan data into a structured **Star Schema** for advanced credit risk analytics.

2. Technologies Used

- **Orchestration:** Bash Scripting (for workflow automation and error exit codes).
- **ETL Engine:** Python 3.12 (Pandas for chunking, SQLAlchemy for ORM).
- **Database:** Oracle Database XE (Stored Procedures, PL/SQL, Indexed Fact Tables).
- **Driver:** Python oracledb (Thin mode) for high-speed database connectivity.

3. Project Workflow

The project follows a "Medallion Architecture" logic, ensuring data is refined at every step:

- **Step 1: Automated Trigger:** A Bash script initializes the pipeline, ensuring environment variables and directories are ready.
- **Step 2: Intelligent Extraction:** Python extracts data in chunks of 50,000 rows to maintain low memory overhead.
- **Step 3: Validation & Rejection (The "Reject Link"):**
    - **Logic:** Records with missing or zero annual_inc are flagged as "Bad Data."
    - **Outcome:** These records are diverted to a dedicated ERR_LOAN_DATA table with a failure reason.

- **Step 4: Transformation:** Interest rates are stripped of special characters and converted to numeric types.
- **Step 5: Staging Load:** Validated data is bulk-loaded into a STG_LOAN_DATA table using explicit Oracle data type mapping to avoid precision errors.
- **Step 6: Warehouse Finalization:** A PL/SQL procedure executes the final move from Staging to the FACT_LOAN_TRANSACTIONS table, assigning surrogate keys and audit timestamps.

4. Quantitative Results

A. Processing Efficiency

- **Total Source Records:** Over 2.2 Million.
- **Batch Processing Performance:** Successfully processed and validated batches of **50,000 records** in approximately **1.4 seconds** per batch.
- **System Integrity:** In a standard test run of 250,000 records, the system identified and rejected **2 invalid records** into the error table while successfully loading **249,998 records** into the warehouse.

B. Data Warehouse Insights

The following metrics were generated from the final **Fact Table**:

Loan Grade	Total Volume	Avg. Loan Amount	Avg. Interest Rate
Grade A	**86,200**	**$14,787.70**	**6.85%**
Grade B	**145,592**	**$14,057.80**	**9.96%**
Grade C	**143,805**	**$14,855.23**	**13.24%**
Grade D	**71,063**	**$16,102.52**	**16.74%**

Loan Grade	Total Volume	Avg. Loan Amount	Avg. Interest Rate
Grade E	**39,902**	**$18,662.15**	**19.14%**
Grade F	**11,146**	**$20,111.17**	**23.38%**
Grade G	**2,290**	**$19,614.52**	**27.50%**

5. Conclusion

The project successfully demonstrates a scalable solution for financial data management. By implementing a **Validation Layer**, the pipeline ensures that only high-quality data reaches the Fact table, preventing downstream analytical errors.

Key Achievements:

- Eliminated "Out of Memory" errors using Python chunking.
- Resolved Oracle-specific binary precision conflicts through explicit schema mapping.
- Established a 100% auditable process via the ETL_AUDIT_LOG table.