**RetailCorp Cloud Data Platform on Snowflake & Snowpark**

**1. Problem Statement**

RetailCorp operates multiple retail stores across different cities. Each store maintains separate systems for sales, inventory, and customer information. The company faces challenges including:

* Fragmented and siloed data across multiple formats (CSV, JSON, Parquet, ORC).
* Difficulty in performing real-time analytics and business intelligence.
* Lack of secure collaboration with external partners and role-based access control.
* High operational costs due to inefficient query execution and redundant storage.
* Limited capability to integrate analytics and ML workflows directly on cloud data.

**Objective:** Build a centralized, scalable, and secure cloud data platform using Snowflake and Snowpark to enable unified analytics, automation, and machine learning-ready data pipelines.

**2. Skill Tower for Project Development**

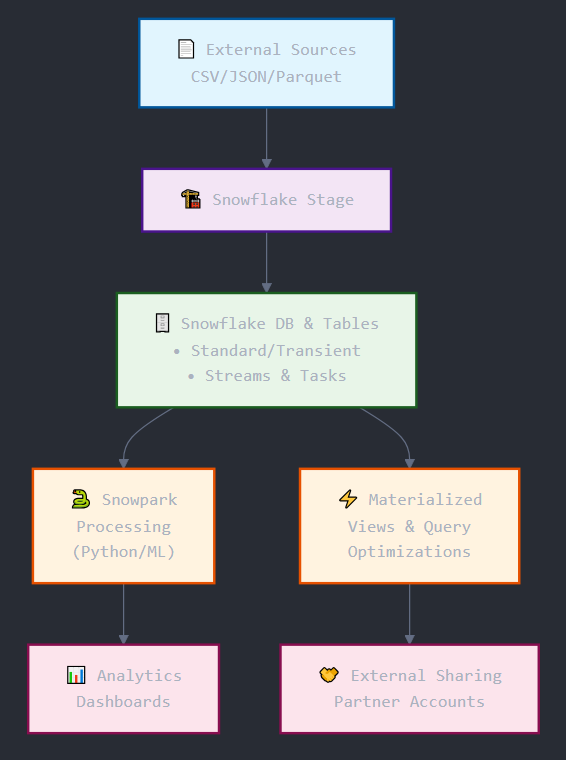
The project requires the following skill sets:

| **Skill Tower** | **Description** |
| --- | --- |
| Cloud Data Platforms | Snowflake setup, databases, schemas, tables, stages, warehouses |
| Data Engineering | ETL/ELT using COPY INTO, streams, tasks, Snowpark pipelines |
| SQL & Query Optimization | Clustering keys, materialized views, query profiling |
| Data Governance | Role-based access control, secure sharing, audit logging |
| Programming & ML | Snowpark Python/Scala/Java, UDFs, integration with ML workflows |
| Analytics & Reporting | Query analysis, dashboards, performance evaluation |
| Project Management | Milestones, documentation, evaluation rubrics |

**3. Use Case / Architecture Diagram**

**Use Case:**  
RetailCorp wants to ingest multi-format sales, inventory, and customer data into Snowflake, perform transformations and analytics, automate updates, share data securely with partners, and integrate ML workflows using Snowpark.

**Architecture Diagram:**



**4. User Stories**

| **ID** | **User Story** | **Priority** |
| --- | --- | --- |
| US01 | As a data engineer, I want to ingest sales, inventory, and customer data in multiple formats into Snowflake. | High |
| US02 | As a data analyst, I want to query large datasets efficiently using clustering and materialized views. | High |
| US03 | As a developer, I want to run Snowpark pipelines to preprocess and integrate ML models. | Medium |
| US04 | As a business manager, I want automated reports and analytics dashboards for sales and inventory. | High |
| US05 | As a security officer, I want role-based access control and secure data sharing for external partners. | High |
| US06 | As a cost manager, I want to optimize warehouse usage and Snowflake pricing. | Medium |

**5. Expected Deliverables**

1. **Snowflake Setup**
   * Databases, schemas, standard & transient tables
   * Stages for raw data
   * Virtual warehouses with cost optimization
2. **Data Pipelines**
   * COPY INTO ingestion scripts
   * Snowflake Streams & Tasks for automation
3. **Performance Optimization**
   * Clustering keys
   * Materialized views
   * Query profiling report
4. **Snowpark Development**
   * DataFrames & transformations
   * UDFs and Stored Procedures
   * Integration with ML workflow
5. **Security & Governance**
   * RBAC implementation
   * Secure data sharing with partners
6. **Reports & Dashboards**
   * SQL query outputs
   * Analytics dashboards showing key metrics
7. **Documentation**
   * Architecture, design decisions, and implementation notes

**6. Implementation Notes**

* **Data Loading:** Use internal Snowflake stages for all source files; COPY INTO commands must specify correct file format.
* **Time Travel & Cloning:** Use zero-copy cloning for dev/test environment.
* **Optimization:** Implement clustering keys and materialized views for frequent queries; monitor warehouse usage for cost optimization.
* **Automation:** Use Streams & Tasks for incremental data processing.
* **Snowpark Integration:** Python-based DataFrame API for transformations and ML pipeline integration.
* **Governance:** Implement RBAC, secure sharing, and audit logging for compliance.
* **Environment:** Development in Snowflake web UI + Python Snowpark environment.