Universidad San Francisco de Quito

Data Mining

Proyecto 02

---

1) Resumen

Construir un **data pipeline** que ingesta **TODOS los archivos Parquet de 2015–2025** del dataset **NYC TLC Trip Record Data** (**Yellow** y **Green**), aterriza en **Snowflake** (esquema **`raw`/bronze**), estandariza/depura en **silver**, y modela **hechos y dimensiones** en **gold** usando **dbt** (ejecutado desde **Mage**). Deben practicar **clustering** en Snowflake y operar credenciales mediante **secrets** con una **cuenta de servicio** de menor privilegio.
Referencias del dataset y diccionarios de datos: NYC TLC (página oficial y data dictionaries): https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

---

2) Objetivos de aprendizaje

1. Ingerir datos **Parquet** a gran escala (histórico 2015–2025) y aterrizarlos en **Snowflake** con orquestación de **Mage**.
2. Implementar **arquitectura de medallas** (bronze/silver/gold) con **dbt**: estandarización, limpieza y **modelo en estrella**.
3. Aplicar **clustering** en tablas grandes de Snowflake y evaluar su impacto mediante **pruning** y **Query Profile**.
4. Operar **secretos** y **cuentas de servicio** con permisos mínimos para ingesta y transformaciones.
5. Entregar **documentación, tests y métricas** que garanticen calidad y reproducibilidad.

---

3) Alcance y restricciones

- **Fuente**: NYC TLC Trip Record Data (enlace oficial). Ingerir **todos los meses 2015–2025** disponibles **en Parquet** de **Yellow** y **Green**). NYC.gov

- **Formato**: **únicamente Parquet**. No convertir otros formatos. Si algún mes carece de Parquet, **documentar explícitamente** la ausencia en el README (tabla de cobertura).
- **Destino**: Snowflake, esquema `raw` (bronze), `silver` (curated) y `gold` (marts).
- **Orquestación**: **Mage** (pipelines para backfill masivo y cargas por mes/lote).
- **Transformaciones**: **dbt** ejecutado desde Mage. Mage AI
- **Clustering**: definir y medir **clustering keys** en al menos **1 hecho grande** de gold. Snowflake Docs
- **Seguridad**: **secretos** + **cuenta de servicio** de menor privilegio.

---

4) Requisitos técnicos

- Cuenta en **Snowflake** con **almacén** (warehouse) para cargas y transformaciones.
- **Mage** en Docker con integración **dbt**. Mage AI
- Acceso a los **archivos Parquet**.
- Conocimientos básicos de **dbt**, **modelado dimensional** y **data lakehouse**

---

5) Arquitectura esperada (alto nivel)

- **Capa bronce (`raw`)** en Snowflake: tablas que **reflejan el origen** (sin transformar) + **metadatos de ingesta** (run_id, ventana temporal, lote/mes).
- **Capa silver**: estandarización, limpieza, tipificación, unificación de esquemas **Yellow/Green**, enriquecimiento con **Taxi Zones**.
- **Capa gold**: **modelo en estrella**:
  - **Hechos**: viajes (granularidad: **1 fila = 1 viaje**)
  - **Dimensiones**: fecha, zona, proveedor, ratecode, payment_type, etc.
- **Orquestación**: Mage coordina **ingesta** y **dbt run/build** (bronze→silver→gold). Mage AI
- **Clustering**: aplicar **cluster keys** en el hecho principal y comparar perfiles de consulta antes/después. Snowflake Docs

---

6) Seguridad y acceso (obligatorio)

- **Secrets**: **todas** las credenciales (Snowflake account, user, password/keys, role, warehouse, database, schema) deben residir en **Mage Secrets**, **no** en el repo.
- **Cuenta de servicio** en Snowflake:
  - Crear **usuario técnico** + **rol** dedicado con privilegios mínimos: `USAGE` en **warehouse**, **database**, **schema**

    - **Prohibido usar cuentas personales con permisos amplios.**
- **Evidencia**: capturas con **nombres** de secretos/roles (valores ocultos) y resumen de privilegios.

---

7) Conocer el dataset (antes de modelar)

- **Yellow/Green**: campos típicos incluyen **pickup/dropoff datetime**, **ubicaciones (Taxi Zone IDs)**, **distancia**, **tarifa itemizada**, **pasajeros**, **método de pago**, **RatecodeID**, etc. Usen los **data dictionaries** oficiales para alinear tipos/semántica. NYC.gov
- **Taxi Zone Lookup**: tabla de referencia de zonas y boroughs (para enriquecer dimensiones de ubicación).
- **Granularidad objetivo** (gold): **1 fila = 1 viaje**.

---

8) Ingesta con Mage → Snowflake (Parquet 2015–2025)

Sin código, pasos que deben implementar y evidenciar:

1. **Estrategia de cobertura**: construir una **matriz de meses** 2015–2025 (Yellow/Green) que indique disponibilidad **Parquet** y **estado de carga** (pendiente/ok/fallida).
2. **Pipeline de backfill** (Mage):
    - **Chunking mensual** (por año/mes) para controlar volumen y reintentos.
    - **Metadatos** por lote (run_id, fechas, tamaño/archivos).
    - **Idempotencia**: reejecutar un mes **no** debe duplicar datos en bronce.
3. **Carga a bronze (`raw`)** en Snowflake:
    - Tablas de origen (Yellow y Green) y tabla de **Taxi Zones**.
    - Guardar **metadatos de ingesta** (run_id, ingest_ts, fuente, año/mes, conteos).
4. **Validaciones de cobertura**: conteos por mes y por servicio (Yellow/Green); tabla de auditoría con resultados.

> Nota: si algún mes **no** existe en Parquet, **no** lo conviertan: marquen **brecha** y sigan. La página oficial describe la naturaleza del dataset y sus campos clave.

---

9) Transformaciones con dbt (medallion)

Bronze → Silver → Gold (ejecutado desde Mage). Conceptos de medallas: **mejorar calidad/estructura por capas**. Databricks

- **Bronze (`raw`)**: reflejo del origen, tipos crudos, sin lógicas de negocio.
- **Silver (`stg`/`core`)**:
    - Estandarizar tipos/zonas horarias; normalizar nombres/valores (p. ej., `payment_type` legible).
    - Reglas de calidad mínimas (nulos, rangos de fechas, distancias/tiempos no negativos; outliers razonables).
    - Unificar **Yellow** y **Green** en un **esquema común** (añadir columna `service_type` = 'yellow'/'green').
    - Enriquecer con **Taxi Zones** (PULocationID/DOLocationID → zona, borough).

- **Gold (`marts`)**: **modelo en estrella**

    - **Hecho principal:** `fct_trips` (1 fila = 1 viaje).
    - **Dimensiones conformadas (compartidas):** `dim_date`, `dim_zone`, `dim_vendor`, `dim_rate_code`, `dim_payment_type, dim_service_type, dim_trip_type`.
    - **Relaciones en el hecho (ejemplo):**
        - `dim_date` → `pickup_date_sk`, `dropoff_date_sk`
        - `dim_time` → `pickup_time_sk`, `dropoff_time_sk`
        - `dim_zone` → `pu_zone_sk` (pickup), `do_zone_sk` (dropoff)
        - Etc

  Clustering ejemplo en Snowflake (tabla de hechos): `CLUSTER BY (pickup_date_sk, etc)`; evaluar también por `dropoff_date_sk` según consultas.

---

10) Clustering en Snowflake (práctica obligatoria)

1. **Seleccionar tabla objetivo**: el **hecho `fct_trips`** (gold), por su volumen.
2. **Elegir clustering key(s)** en función de **patrones de consulta**: típicamente por **fecha/hora de pickup** y/o **PULocationID** (y, si aplica, `service_type`).
3. **Medir antes/después**:
    - Capturar **Query Profile** (tiempo, particiones leídas, pruning) **antes** de clusterizar.
    - Aplicar **cluster key(s)**; si habilitan **auto-clustering**, documentar.
    - Re-ejecutar consultas representativas y comparar métricas (tiempo, micro-partitions escaneadas).
4. **Conclusión**: justificar si el clustering aporta, y qué llaves elegirían a largo plazo (evitando sobreclusterizar).
   Conceptos de **micro-partitions**, **pruning** y **cluster keys** en Snowflake.

Tip: diferenciar **clustering** de **Search Optimization Service** (SOS) para búsquedas ultra selectivas; aquí el foco es clustering.

---

11) Calidad y documentación

- **Tests** (dbt): unicidad de llaves naturales, `not_null` en campos clave, `accepted_values` (p. ej., `payment_type`) y relaciones PU/DO con zonas válidas.
- **Diccionario de datos**: describir **columnas finales** en gold y su origen (lineage).
- **Auditoría de cargas**: tabla/reporte con conteos por mes/servicio y % de filas descartadas por reglas de calidad (si descartaron).

---

12) Entregables (GitHub)

1. **README** completo con:
    - Descripción y **diagrama** de arquitectura (bronze/silver/gold) y orquestación en Mage.
    - **Cobertura de meses 2015–2025** (matriz por servicio) y estado de carga (Parquet).
    - Estrategia de pipeline de backfill mensual e **idempotencia**.
    - **Gestión de secretos** (nombres y propósito) y **cuenta de servicio / rol** (permisos mínimos).
    - Diseño de **silver** (reglas de limpieza/estandarización) y **gold** (hechos/dimensiones).
    - **Clustering**: llaves elegidas, métricas antes/después, conclusiones.
    - **Pruebas** (qué validan y cómo interpretar resultados).
    - Troubleshooting (archivos faltantes, fallas de carga, límites, costos).
2. **Docker Compose donde esta el contenedor de orquestación**
    - **Proyecto de Mage** versionado (pipelines de ingesta y de transformaciones con dbt). Mage AI
    - **Proyecto dbt** con capas **bronce/silver/gold**, documentación y tests.
3. **Evidencias** (capturas):
    - Secrets/roles (valores ocultos), matriz de cobertura, ejecuciones en Mage, lineage dbt, Query Profiles de clustering (antes/después).
4. **Notebook con respuestas a 5 preguntas de negocio** (ver Sección 14) basadas **exclusivamente** en la capa gold (indicando tablas y medidas usadas).

---

13) Rúbrica de evaluación (100 pts)

- **Ingesta Parquet 2015–2025 (25 pts)**: cobertura clara (Yellow/Green), idempotencia, metadatos por lote y auditoría de conteos.
- **Arquitectura de medallas (25 pts)**: bronze fiel al origen, silver estandarizado y enriquecido, gold en **estrella** (hechos/dimensiones).
- **Clustering en Snowflake (20 pts)**: selección de keys justificada, comparación antes/después, análisis de pruning. Snowflake Docs
- **Seguridad y operación (15 pts)**: secrets + cuenta de servicio con mínimo privilegio; documentación; orquestación en Mage.
- **Calidad y documentación (15 pts)**: tests dbt, diccionario de datos, README claro, evidencias completas.

---

14) 10 preguntas de negocio (capa gold, obligatorias)

1. **Demanda por zona y mes**: ¿cuáles son las 10 zonas con más viajes por mes? (PU y DO por separado).
2. **Ingresos y propinas**: ¿cómo varían los **ingresos totales** y el **tip %** por **borough** y **mes**?
3. **Velocidad y congestión**: promedio de **mph** por franja horaria y borough (viajes diurnos vs. nocturnos).
4. **Duración del viaje**: percentiles (p50/p90) de duración por **PULocationID** (pickup)
5. **Elasticidad temporal**: distribución de viajes por **día de semana** y **hora**; ¿cuáles son las horas pico?

Todas deben resolverse sobre **gold** indicando **hechos/dimensiones** y **métricas con SQL**. Crear un notebook de **data_analysis.ipynb,** donde se conectan a la capa gold de Snowflake y ahí hacen las queries para responder cada pregunta.

---

15) Checklist de aceptación (copiar al README y marcar)

- ☐ **Cargados todos los meses 2015–2025** (Parquet) de **Yellow** y **Green**; matriz de cobertura en README. NYC.gov
- ☐ **Mage** orquesta backfill mensual con **idempotencia** y metadatos por lote.
- ☐ **Bronze (`raw`)** refleja fielmente el origen; **Silver** unifica/escaliza; **Gold** en estrella con `fct_trips` y dimensiones clave.
- ☐ **Clustering** aplicado a `fct_trips` con evidencia antes/después (Query Profile, pruning). Snowflake Docs

- [ ] **Secrets** y **cuenta de servicio** con permisos mínimos (evidencias sin exponer valores).
- [ ] **Tests dbt** (not_null, unique, accepted_values, relationships) pasan; **docs** y **lineage** generados.
- [ ] Notebook con respuestas a las **5 preguntas de negocio** desde **gold**.