LA Crime Analysis – End-to-End Project Report

1. Project Overview

This project implements a complete end-to-end data engineering and analytics solution using the *Los Angeles Crime Dataset (2020–Present)* from the LA Open Data Portal.
The objective is to design a dimensional model, build a Lakehouse pipeline (Bronze → Silver → Gold), and produce analytics-ready datasets for dashboarding.

All work was performed using:

- **Databricks** (Bronze/Silver/Gold)
- **Databricks DLT** for pipelines
- **Power BI/Tableau** (visualization)
- **Navicat / ERD tool** for modeling

2. Data Profiling

Data profiling was performed on the raw dataset to understand:

- **Data quality**
- **Missing values**
- **Invalid values**
- **Category distributions**
- **Schema correctness**
- **Date/time ranges**

Profiling Findings

Row & Column Overview

- The dataset contains **~1 million crime records**.

- Columns cover incident details, time, location, victim demographics, crime codes, weapons, and case statuses.

Null & Missing Values

- `Vict Age`: contains nulls and some invalid entries.
- `Vict Sex`: contains blanks or unknown values.
- `LAT` / `LON`: include rows with `0` or missing coordinates.
- Most core identifiers (`DR_NO`, `Crm Cd`, `AREA`) contain no nulls.

Invalid Value Summary

- **Invalid ages (<0 or >120)**: cleaned in Silver layer.
- **Invalid times (<0000 or >2359)**: corrected during time parsing.
- **LAT/LON = 0**: treated as missing (set to NULL).
- **Date formats** are consistent and parse correctly.

Category Distributions

- **Victim Sex**: majority `M` and `F`, some `X` (unknown).
- **Status Codes**: most frequent = `IC – Investigation Continuing`.
- **Crime Types**: dominated by theft, burglary, simple assault, vehicle break-ins.
- **Weapons**: frequent categories include “UNKNOWN WEAPON”, “HAND GUN”, “VERBAL THREAT”.
- **Areas**: high-volume divisions include **77th Street**, **Central**, **Hollywood**, and **Pacific**.

Date Range Validation

- `DATE OCC` spans **2020** → **Present**, matching the dataset’s purpose.
- No abnormal date outliers.

Conclusion of Profiling

Key issues were identified (invalid ages, invalid times, missing geo, inconsistent victim attributes) and addressed in the Silver layer.
The dataset is suitable for analytics after cleaning.

3. Dimensional Model (ERD)

A **Star Schema** was designed and approved by the professor.
Grain: **One row per crime incident (DR_NO)**.

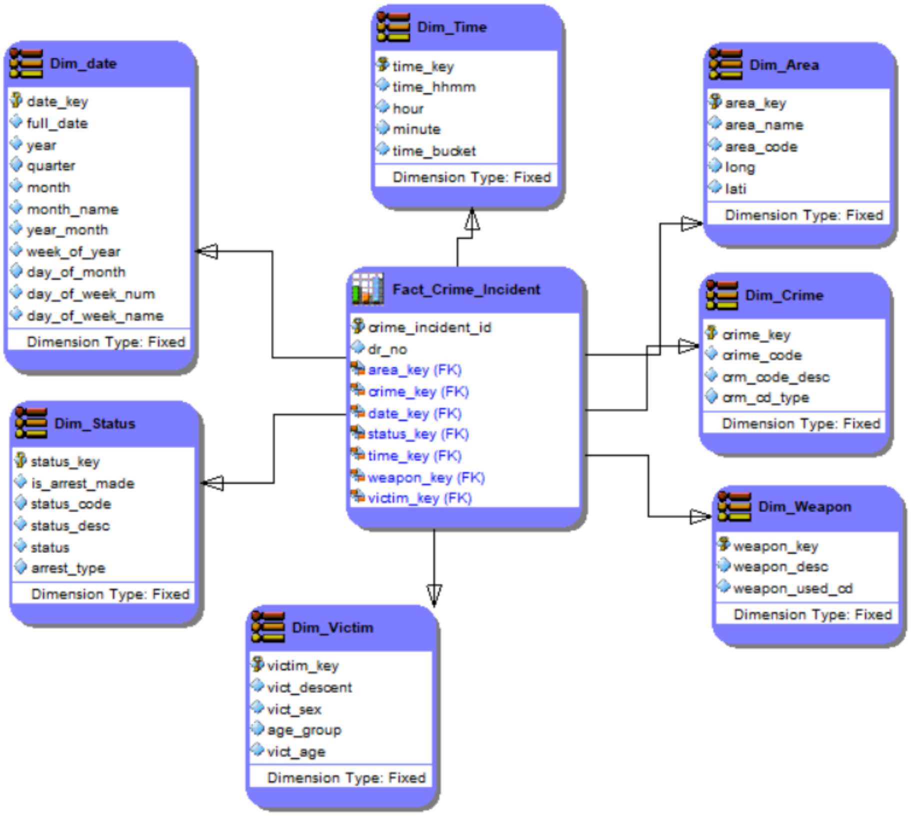
Fact Table

FACT_CRIME_INCIDENT_LA
Contains foreign keys to all dimensions + measures:

- date_key
- time_key
- area_key
- crime_key
- weapon_key
- status_key
- victim_age, victim_sex, victim_descent
- lat / lon
- incident_count = 1

Dimension Tables

Dimension	Description
dim_date_la	Year, quarter, month, weekday
dim_time_la	Hour, minute, time bucket
dim_area_la	LAPD area, name, representative lat/lon
dim_crime_la	Crime code + description
dim_status_la	Arrest type, status code, is_arrest_made
dim_weapon_la	Weapon used in incident
dim_victim_la	Age group, sex, descent



4. Bronze Layer (Raw Ingestion)

The Bronze layer stores the raw dataset as-is.

Implementation

- Loaded dataset using `spark.read` with schema inference.
- Stored as **bronze_la_crime** Delta table.
- No transformations applied (raw zone requirement).

Purpose

- Serves as the foundational raw source for all downstream cleaning.

5. Silver Layer (Data Cleaning & Standardization)

Silver layer performs all necessary cleaning and standardization.

Key Transformations

- **Date Parsing:**
  Correct format `yyyy MMM dd hh:mm:ss a` used to create timestamps + date_key.

- **Time Parsing:**
  Converted TIME OCC into hour/minute.
  Derived `time_bucket`: *Morning, Afternoon, Evening, Night*.

- **Age Cleaning:**
  Ages <0 or >120 → set to NULL.
  Created `age_group` bucket.

- **Victim Sex Standardization:**
  Only `M`, `F`, and `X` allowed.

- **Geo Cleaning:**
  LAT/LON = 0 → set to NULL.

Silver Output

Cleaned dataset stored as **silver_base_la**, used to generate all dimensions and the fact table.

6. Gold Layer (Analytical Tables)

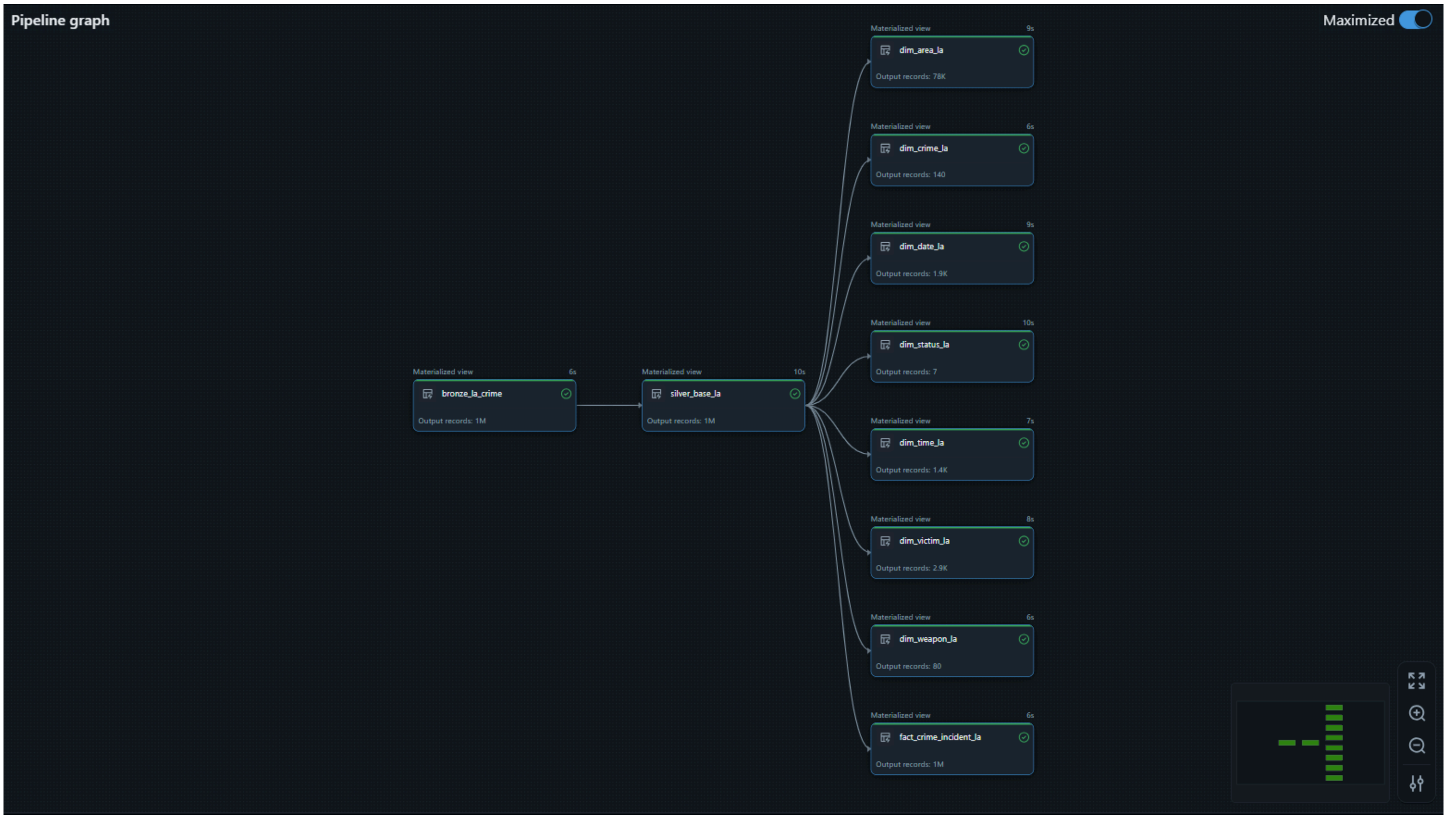
Gold layer contains final Star Schema tables powering the visualization layer.

Final Tables

- **dim_date_la**
- **dim_time_la**
- **dim_area_la**
- **dim_crime_la**
- **dim_status_la**
- **dim_weapon_la**
- **dim_victim_la**
- **fact_crime_incident_la**

Purpose

- Fully cleaned, conformed, analytics-ready dataset.
- Directly supports answering business questions.
- Used by Power BI / Tableau dashboards.



7. Business Requirements Coverage

Requirement	Satisfied By
Crime trends (year/month/quarter)	Fact + dim_date_la
Day-of-week patterns	Fact + dim_date_la
Time-of-day analysis	Fact + dim_time_la
High-crime areas	Fact + dim_area_la

Hotspots (area-based)	fact_crime_incident_la.lat/lon or area-level aggregation
Crime type insights	dim_crime_la
Weapon usage	dim_weapon_la
Age patterns	dim_victim_la
Gender patterns	dim_victim_la
Adult vs Juvenile Arrest Ratio	dim_status_la

All business questions can be answered with the final Gold dataset.

8. Conclusion

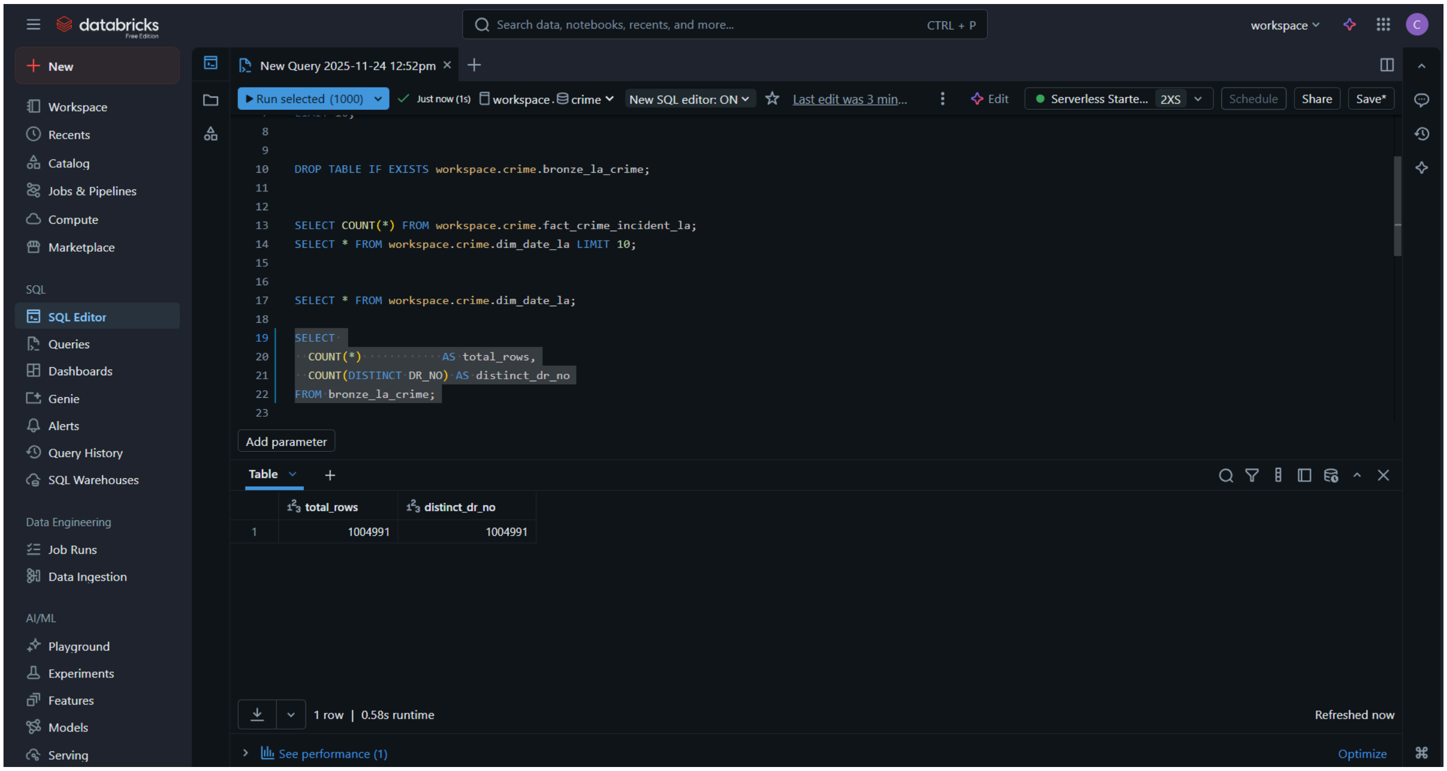
This project was successfully implemented:

- A complete data pipeline (Bronze → Silver → Gold)
- A validated and cleaned dataset
- A professionally designed Star Schema
- High-quality data suitable for analytical dashboards
- Full alignment with assignment deliverables

Some Profiling Validations

1. Structural & Uniqueness Validations

Rule: Each incident ID (`DR_NO`) should be unique.



```sql
DROP TABLE IF EXISTS workspace.crime.bronze_la_crime;


SELECT COUNT(*) FROM workspace.crime.fact_crime_incident_la;
SELECT * FROM workspace.crime.dim_date_la LIMIT 10;


SELECT * FROM workspace.crime.dim_date_la;

SELECT
  COUNT(*)            AS total_rows,
  COUNT(DISTINCT DR_NO) AS distinct_dr_no
FROM bronze_la_crime;
```

	total_rows	distinct_dr_no
1	1004991	1004991

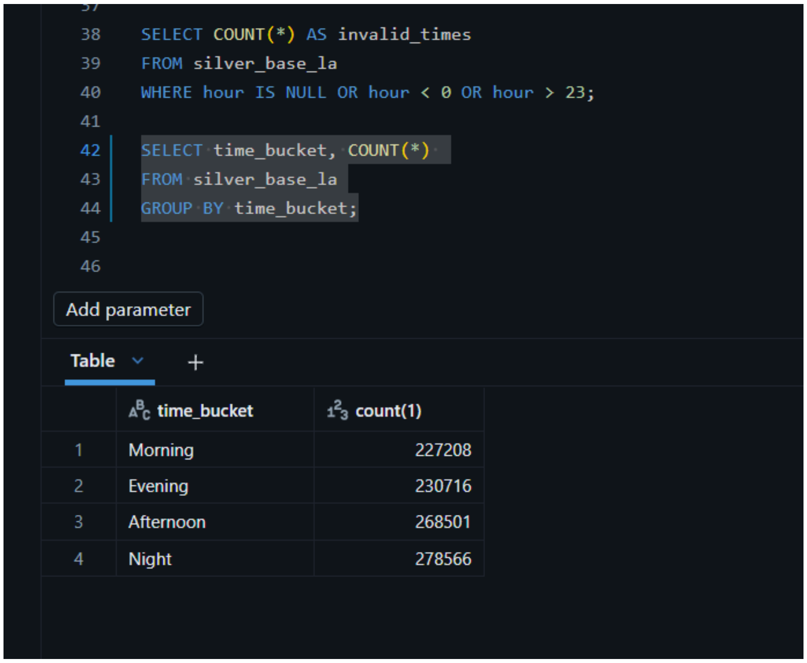
1 row | 0.58s runtime

Validation:

`total_rows = distinct_dr_no` → passes.

If not, list how many duplicates and how you handle them.

2. Status & Arrest Validations



```sql
SELECT COUNT(*) AS invalid_times
FROM silver_base_la
WHERE hour IS NULL OR hour < 0 OR hour > 23;

SELECT time_bucket, COUNT(*)
FROM silver_base_la
GROUP BY time_bucket;
```

	time_bucket	count(1)
1	Morning	227208
2	Evening	230716
3	Afternoon	268501
4	Night	278566

Rule: No invalid time values in Silver.

Validation:

`invalid_times = 0` after Silver cleaning.

Time buckets cover all rows

Rule: Every non-null time should fall into a `time_bucket`.

Validation:

No NULL `time_bucket` for valid times; counts > 0 in each bucket (Morning/Afternoon/Evening/Night).