Main Exercise: Joining Tables and Creating a Wide Table

Objective: - Join `crime_data` (which is the merged data of `burglary` and `robbery`) and `crimecode` tables using appropriate keys and create a comprehensive wide table.

Task: - Identify the common key(s) between the tables. - Perform a SQL join operation. - Write the resulting table to a new BigQuery table.

SQL Example:

```
CREATE OR REPLACE TABLE your_project_id.your_dataset.new_wide_table AS
SELECT
    a.*,
    b.code_group,
    b.code_group_description
FROM
    your_project_id.your_dataset.crime_data a
JOIN
    your_project_id.your_dataset.crimecode b
ON
    a.OFFENSE_CODE_GROUP = b.code;
```

Note: - Ensure to replace `your_project_id` and `your_dataset` with the actual project ID and dataset name you are using. - Verify the column names to match with your actual schema.

Additional Exercise 1: Time-based Analysis

Objective: - Analyze the crime data based on the time (hour, day, month, year) to identify patterns or trends.

Task: - Group the data by the desired time unit and perform aggregations like COUNT, AVG, etc.

SQL Example:

```
SELECT
    EXTRACT(HOUR FROM OCCURRED_ON_DATE) as hour,
    COUNT(*) as num_incidents
FROM
    your_project_id.your_dataset.crime_data
GROUP BY
    hour
ORDER BY
    hour;
```

Additional Exercise 2: Geographic Analysis

Objective:

- Explore and analyze the geographic distribution of the different types of crimes

Task:

- Group by geographic location and analyze the count of different types of crimes

SQL Example:

```
SELECT
    DISTRICT,
    OFFENSE_CODE_GROUP,
    COUNT(*) as num_incidents
FROM
    your_project_id.your_dataset.crime_data
GROUP BY
    DISTRICT, OFFENSE_CODE_GROUP
ORDER BY
    num_incidents DESC;
```

Additional Exercise 3: Correlation Analysis

Objective:

- Investigate if there is any correlation between different types of crimes

Task:

- Calculate the number of incidents for different crime types and observe if there is any noticeable pattern or correlation

SQL Example:

```
SELECT
    OFFENSE_CODE_GROUP,
    COUNT(*) as num_incidents
FROM
    your_project_id.your_dataset.crime_data
GROUP BY
    OFFENSE_CODE_GROUP
ORDER BY
    num_incidents DESC;
```

Note:

- Ensure to verify the SQL queries with your actual schema and modify them accordingly
- Encourage students to explore further by creating visualizations based on these analyses using tools like Google Data Studio