Lab 2

Meaning of "truncate" in your case:

When it's said that the **staging table will be repeatedly loaded without truncating**, it means:

✅ The existing data in the **staging table** is **not cleared** before each new load.

In your scenario, you're using the **OnPreExecute event handler** to **automatically truncate the staging table before the data flow task runs**. Here's how this works and why it's useful

---

Excel

Roll-up Operation

Roll-up aggregates data moving from lower to higher levels in a dimension hierarchy.

Implementation Steps:

1. Create a PivotTable by going to **Insert** > **PivotTable**
2. Add a dimension (e.g., Time) to rows with multiple hierarchy levels (Year > Quarter > Month)
3. Add a measure (e.g., Sales Amount) to values
4. Collapse lower levels to see aggregated data at higher levels

Drill-Down Operation

Drill-down navigates from higher to lower levels in a dimension hierarchy to see more detailed data.

Your implementation now shows:

1. **A complete three-level hierarchy (Country > City > Street Address)**
2. **The ability to drill down from the highest level (US with 9684 reserved rooms)**
3. **Expanded cities showing their street-level details**
4. **Consistent aggregation of values at each level of the hierarchy**

Slice Operation

Slicing extracts a specific portion of the cube by filtering on one dimension.

Implementation Steps:

1. In your PivotTable, add a dimension to the filter area (e.g., Product Category)
2. Select a specific value from the filter dropdown (e.g., "Electronics")
3. The PivotTable now shows data only for that specific slice of the cube

Dice Operation

Dicing extracts a subcube by filtering on multiple dimensions simultaneously.

Implementation Steps:

1. Add multiple dimensions to the filter area (e.g., Product Category, Region, Time Period)
2. Select specific values for each filter (e.g., "Electronics", "West Region", "Q1 2024")
3. The resulting view shows data for that specific subcube defined by multiple criteria

Pivot Operation in OLAP

The pivot operation in OLAP allows you to rotate or rearrange the data view by switching rows and columns, effectively changing the perspective of your analysis without altering the underlying data.

What is a Pivot Operation?

A pivot operation reorganizes your data visualization by:

- Rotating the dimensional structure of a cube
- Swapping rows and columns
- Creating alternative views of the same underlying data
- Enabling different analytical perspectives

Power bi

Report 3: Drill-Down

- **Purpose**: Explore data hierarchically **within the same visual** (e.g., from city → address).
- **Behavior**: Expands/collapses levels using ▶ **buttons** (e.g., click "Chicago" to see all its addresses).
- **Scope**: Stays on the **same report page**; no navigation.
- **Example**:

Copy

Download

▶ Chicago (Total)

 ├– 20 W Kinzie St: 1,299 rooms

└– 660 N State St: 408 rooms

Report 4: Drill-Through

- **Purpose**: Jump to a **dedicated details page** about a specific data point.
- **Behavior**: Right-click → **"Drill Through"** to a pre-built page (e.g., click "Chicago" → opens Page 9 with address details).
- **Scope**: Navigates to a **new page** focused on the selected item.
- **Example**:
    - Main page: Bar chart of cities.
    - Drill-through page: Table/map of Chicago's addresses, filtered to only Chicago data.

Key Contrast

Feature	**Drill-Down**	**Drill-Through**
Navigation	Same page	New page
Interaction	Expand/collapse hierarchy	Right-click → Jump to details
Use Case	Explore levels (e.g., time periods)	Deep-dive into one entity (e.g., a city)

Visual analogy:

- Drill-down = **Zooming in/out** on a map.
- Drill-through = **Opening a street view** for one location.

Questions

1. **Where get data**
2. **How partition the tables from data source**
3. **Available files**

Kaggle .had to partition the csv files to 5 csv files and 1 text file .hotel

Hotel category

Review,user,accomadation transaction complete time. And location txt

4. **Why star schema?**

primarily because you have a single fact table (FactHotel) connected to several dimension tables and to connect dimension tabole with fact table using ids (DimDate, DimReview, DimLocation, DimHotelCategory, DimUser).

5. **Run the fact table to see the availability of data**

Fact tables hold quantitative data (numbers) that businesses analyze, such.Fact tables link to dimension tables (like DimUser, DimLocation) via foreign keys.

6. **Derived column**
   When creating dim location
   To add modified dates

   When creating fact tables

1. First Derived Column (Date Formatting)

Purpose: Clean up timestamp strings (likely from JSON/API sources).

- Input: Raw dates like 2024-05-15T12:00:00Z (ISO format with "T" and "Z").

Output:Cleaned strings (Unicode) ready for database insertion.

Why?

Databases often prefer plain datetime formats (YYYY-MM-DD HH:MM:SS) over ISO formats

2. Second Derived Column (Audit Timestamps & Calculations)

Purpose: Add system-generated metadata and placeholders.

- **Output**:
  - New columns added to the data flow for loading into FactHotel.

Why?

- Tracks when records are inserted (create_time).
- Reserves space for future ETL steps to populate complete_time and process_time.

Process time column in fact table

Remember total amount is calculated for a 1 day

7. **Generated table**
   Dim tables, fact tables and staging table
8. **Show staging,updating values in transform**
9. **Show cube.**
10. **Why Slowly changing dimension**

Slowly changing values: values that change over time, but not frequently.
In here location data can be change in the future.
handles changes in your dimension data (like DimLocation) over time by deciding:

1. Should we update the existing record? (Type 1: Overwrite)
   - Example: Fixing a typo in City name ("New Yrok" → "New York").
2. Should we keep history? (Type 2: Create a new record)
   - Example: Tracking when a hotel moves from "Paris" to "London" by saving both versions**.**

11.Derived columns.
In fact table modified date and inserted dates for location

12.What are lookups. Why using lookup

Lookups are critical for connecting raw source data (staging tables) to dimension tables before loading into the fact table.**Match business keys** (e.g., UserID, ReviewID from staging) to **surrogate keys** (UserKey, ReviewKey from dimensions

In fact table. After sdc derived column is to store the modified date of changed locations.also another derived column to store the insert date

13.Surrogate key?

Surrogate keys are system-generated unique identifiers (usually integers) used as the primary key in dimension tables. they are used to replace business keys as the primary identifiers.

14.Why Event handling? Preexecution and truncate?

When stag tables loaded repeatedly the same data loaded to the table multiple times and cause redundancy. To avoid that we use onpreexecution event handles. So it truncate the staging table before run the task

15.Types of slowly changing values

1. Type 1 SCD: Overwrite (No History)

- What it does: Updates existing records without tracking changes.no history
- Use case: Fixing errors or unimportant attributes.
- Example: Correcting a typo in DimLocation (e.g., "New Yrok" → "New York").
- Impact: Historical data is lost—only the current state is kept.

---

2. Type 2 SCD: Track History (New Row)

- What it does: Creates a new record with a new surrogate key when data changes.
- Use case: Critical attributes where history matters (e.g., customer address changes).
- Example:

    - Original: LocationSK=1, City="Paris", ValidFrom=2023-01-01, ValidTo=NULL
    - After change:
        - Old: LocationSK=1, City="Paris", ValidTo=2024-05-01
        - New: LocationSK=2, City="London", ValidFrom=2024-05-01, ValidTo=NULL
- Key fields added:
    - ValidFrom/ValidTo dates
    - Version flags (e.g., IsCurrent)

3. Type 3 SCD: Track Limited History (Current + Previous Value)

- What it does: Stores only the current + one previous value in the same row.
- Use case: Rarely used; for tracking a single prior state (e.g., "Previous City").
- Example:
    - LocationSK=1, CurrentCity="London", PreviousCity="Paris", LastChangeDate=2024-05-01

Hotel_DW_updating

To update the complete time of transaction and update the calculated value of transaction hours.(for calculate dhours there is a procedure to run)