**End-to-End Process: Extracting SQL Query Results ,converting to JSON and processing using Python**

**Pallepati Pallavi GitHub Link:** [**Sample Task**](https://github.com/Palle-1997/Sample/tree/main/Example%20Task)

**Introduction:**

This document provides a **step-by-step** process on how we have executed to:

* Extract data from an **SQL database** using a query.
* Convert the query results into **JSON format.**
* Process and display the data using **Python & Pandas**.
* Maintain **data lineage** for transparency and traceability.

By following this process, we can efficiently handle **database exports**, structure data into a **machine-readable format**, and ensure **auditability**.

**Step 1: Understanding Data Lineage**

What is Data Lineage?

Data lineage tracks the flow of data from source to transformation and final output. This helps in:

* Understanding data origin (SQL database).
* Ensuring data consistency (through JSON export).
* Tracking modifications (through Python processing).

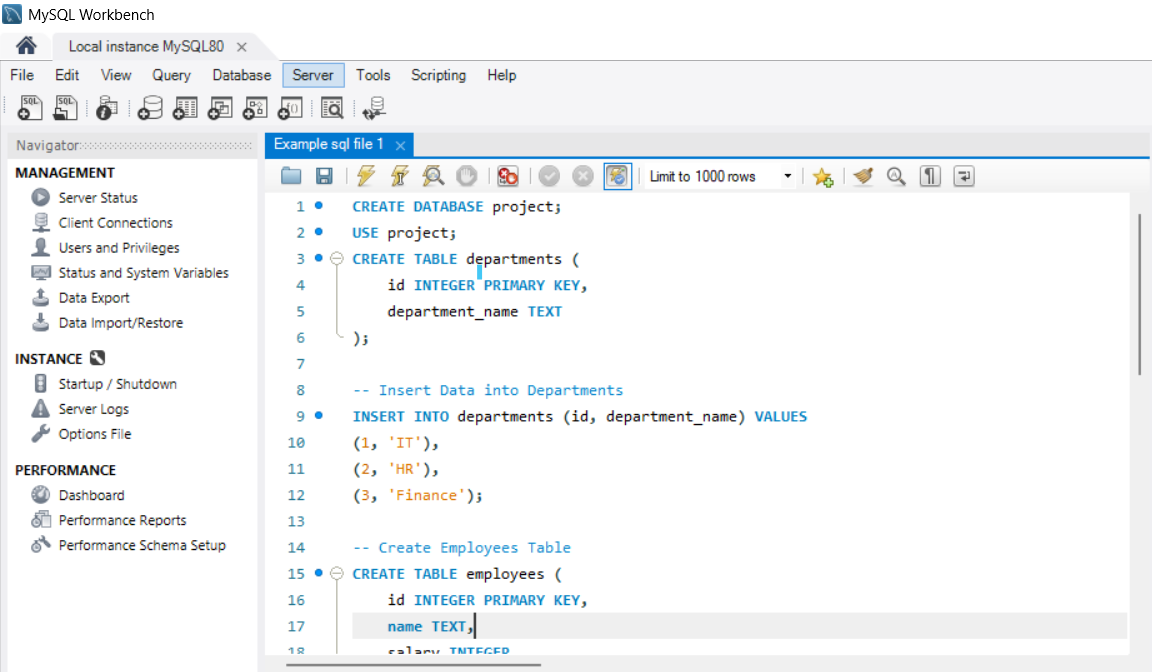
**Data Lineage Flow in This Process**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **Step** |  |  | | --- | --- | --- | | |  | **Action** |  | | --- | --- | --- | | |  |  | **Data Representation** | | --- | --- | --- | |
| |  |  |  | | --- | --- | --- | | **1.Source Data** |  |  | | |  |  |  | | --- | --- | --- | |  | Read data from SQL database |  | | |  |  |  | | --- | --- | --- | |  |  | **Raw Table Data (SQL)** | |
| |  |  |  | | --- | --- | --- | | **2.Transformation** |  |  | | |  |  |  | | --- | --- | --- | |  | Apply SQL query (e.g., SELECT \* FROM employees) |  | | |  |  |  | | --- | --- | --- | |  |  | **Query Result (Tabular Format)** | |
| |  |  |  | | --- | --- | --- | | **3️.Conversion** |  |  | | |  |  |  | | --- | --- | --- | |  | Convert to JSON format |  | | |  |  |  | | --- | --- | --- | |  |  | **JSON File (Structured Data)** | |
| |  |  |  | | --- | --- | --- | | **4️.Processing** |  |  | | |  |  |  | | --- | --- | --- | |  | Load JSON into Python & Pandas |  | | |  |  |  | | --- | --- | --- | |  |  | **Data Frame (Tabular View)** | |
| |  |  |  | | --- | --- | --- | | **5️.Output** |  |  | | |  |  |  | | --- | --- | --- | | Display thestructured data |  |  | | |  |  |  | | --- | --- | --- | |  |  | **Human-Readable Table & JS** | |

**Step 2: Extracting Data from SQL File**

**I have used MySQL Workbench to Create a Query File**

1. Open **MySQL Workbench** or any SQL client.
2. Write and save the SQL query in a .sql file.



Here is the link to the SQL file that we have executed : [**Sample Task**](https://github.com/Palle-1997/Sample/tree/main/Example%20Task)

**Step 3: Export SQL Query Result to JSON**

For this I have just exported the recently executed query into json format, using MySQL Workbench

Run the SQL query inside MySQL Workbench.

Click Export → Select JSON format → Save the file(anywhere on the desktop).

**Step 4: Processing JSON Data in Python and putting it into tabular format**

Python Code to Read and Process JSON

Reads the JSON file (open())  
Prints structured JSON output (Json. Dumps())  
Converts JSON into a table (pandas.DataFrame())

A screenshot of a computer

AI-generated content may be incorrect.

**Summary**

* **Extracted data** query file from SQL using MySQL Workbench.
* **Added variations, Converted SQL query results into JSON** format.
* **Loaded JSON into Python** and displayed it in a table.
* **Ensured data lineage** to track data from source to output.

This document provides a **step-by-step process** on exporting **SQL query results** to **JSON** and processing it using **Python**. The process ensures **data consistency, automation, and easy data manipulation**