SQL in Python

Renato R. Maaliw III, *DIT*
Engineer & Professor 1, College of Engineering
Southern Luzon State University
Lucban, Quezon, Philippines

Why SQL and Pandas are the cornerstone in data extraction?

For a data professional, SQL is often the first step in data extraction and performing basic transformations on data.

Why SQL and Pandas are the cornerstone in data extraction?

Once the data is in Python, Pandas shine in preparing data for machine learning models, statistical analysis and data exploration.

Why SQL and Pandas are the cornerstone in data extraction?

With Pandas, data can be refined further, missing values can be replaced, and exploratory data analysis (EDA) can be performed to get insights that inform downstream machine learning models and analyses.

Complementary Skill Sets

SQL allows you to pull data but its capabilities for data manipulation are limited, especially for statistical or machine learning task.

Pandas can take over where SQL leaves off, offering more advanced manipulation tools, data wrangling, and integrating with other Python libraries.

Basics of SQL (View only)

SELECT (Extract)

The *SELECT* is the used to retrieve data from a source. You specify the columns you want to extract or you can use ***** to select all columns

Basics of SQL (View only)

AS (Alias)

AS allows you to rename a column temporarily in the query output. Aliasing can make your query results more readable

Basics of SQL (View only)

WHERE (Certain Conditions)

(NOT, AND, OR, <, >, =, !=, etc.)

The *WHERE* filters records based on a specified conditions and operators to allow you to refine conditions.

Basics of SQL (View only)

LIKE (Wildcards)

(%, _)

The *LIKE* operator is used for pattern matching in strings.

Basics of SQL (View only)

ORDER BY (Arrangements)

(ASC or DESC)

ORDER BY sorts the results of query by one or more columns. By default, it sorts in ascending order.

Basics of SQL (View only)

LIMIT (restrict number of rows)

LIMIT restricts the number of rows returned by a query.

Basics of SQL (View only)

NOTE:

WHERE is used to filter rows **before** any aggregation occurs.

HAVING is used to filter groups or aggregated results **after** *GROUP BY* has been applied.

•**HAVING** is used to filter groups or aggregated results **after** GROUP BY has been applied.

Cognate/Professional Electives

Basics of SQL (View only)

GROUP BY (Aggregation)

SUM(), AVG(), COUNT(), MAX(), MIN()

The *GROUP BY* group rows that have the same values in specified columns,

HAVING (Filter, when using *GROUP BY***)**

Used with aggregation functions to filter groups based on aggregate conditions

Basics of SQL (View only)

DISTINCT (Unique)

Returns unique values, often used in SELECT to remove duplicates

Basics of SQL (View only)

IN (Shorthand *OR*)

Used to specify multiple possible values for a column in a *WHERE* clause, making it a cleaner alternative to using multiple *OR* conditions.

Basics of SQL (View only)

BETWEEN (Range)

Filters data within a specified range of values (inclusive of the endpoints)

Basics of SQL (View only)

UNION (combine queries)

UNION combines the results of two queries and removes duplicates. Used when pulling similar data.

Thank you very much for listening.