Mastering Data Manipulation with SQL

TOP 5 SQL Skills For Data Science

Dr. Hervé Teguim

Mastering Data Manipulation with SQL

01. Subqueries

Subquery is a query nested within another query to provide intermediate results that the outer query can use.
They can be used in various parts of a SQL statement, such as the SELECT, FROM, WHERE, and HAVING clauses.

Dr. Hervé Teguim

Example: Identify Customers Who Made Purchases Above the Average Order Value

```
SELECT
      customer_id,
      order_id,
      order_value
FROM
     orders
WHERE
   order_value > (SELECT
                        AVG (order_value)
                         FROM orders);
```

02. Case Statements

CASE statements are used to handle conditional logic directly in the SQL queries. They allow to perform different actions based on different conditions, making the queries more dynamic and flexible.

Dr. Hervé Teguim

Example: Categorize Products Based on Stock Levels

```
SELECT
    product_name,
    stock_quantity,
    CASE
        WHEN stock_quantity < 10 THEN 'Out of Stock'
        WHEN stock_quantity < 50 THEN 'Low Stock'
        ELSE 'In Stock'
    END AS stock_status
FROM
products;
```

03. JOINs

JOINs help combine multiple tables based on matching keys. There are different types of JOINs, each serving a specific purpose:

INNER JOIN: Matches only rows with a common value in both tables.
LEFT JOIN: Includes all rows from the left table and matching rows from the right tab.
RIGHT JOIN: Includes all rows from the right table and matching rows from the left tab.
FULL JOIN: Includes all rows from both tables, regardless of match.

Dr. Hervé Teguim

Example: Get Customer Orders with Product Details

```
SELECT
    orders.order_id,
    orders.customer_id,
    products.product_name,
    products.price
FROM
    orders
INNER JOIN
    products
ON
    orders.product_id = products.product_id;
```

Mastering Data Manipulation with SQL

04. Common Table Expressions (CTEs)

CTEs (Common Table Expressions): powerful feature in SQL that allow to create temporary result sets within a query using the WITH clause. This can make complex queries easier to read and maintain.

Dr. Hervé Teguim

Example: Identify Top-Selling Products by Category

```
WITH ProductSales AS (
    SELECT product_id,
        category_id,
        SUM(sales_amount) AS total_sales
    FROM sales
    GROUP BY product_id, category_id
)
SELECT category_id,
            product_id,
            total_sales
FROM ProductSales
WHERE total_sales = (
    SELECT MAX(total_sales)
    FROM ProductSales AS ps
    WHERE ps.category_id = ProductSales.category_id
);
```

Mastering Data Manipulation with SQL

05. Windows Functions

Window Functions perform calculations across a set of table rows related to the current row, without collapsing rows like GROUP BY does. They allow to perform complex calculations such as running totals, moving averages, and rankings within a specified window of rows.

Dr. Hervé Teguim

Example: Calculate Running Total of Sales for Each Salesperson

```
SELECT
    salesperson_id,
    sales_date,
    revenue,
    SUM(revenue)
        OVER (PARTITION BY
        salesperson_id
        ORDER BY sales_date) AS running_total
FROM
    sales_data;
```