Introduction to SQL for Data Science

To understand the role of SQL in data science, we must know where SQL fits in, from collecting data to deploying models. In data science, jobs are mostly divided into four categories:

- Data engineers who collect and clean data
- Data analysts who extract valuable insights
- Data scientists who find trends and patterns and create machine learning models
- Machine learning engineers who deploy these models.

At the beginning stage of data science projects, SQL is used to fetch data from databases. Without SQL, data would be unusable and take up unnecessary space. So this blog aims to highlight two things:

- Importance of SQL in data science
- Important SQL commands for Data Science

What is SQL?

SQL (Structured Query Language) is a programming language for managing and manipulating relational database management systems (RDBMS) like Oracle, MySQL, Microsoft SQL Server, and PostgreSQL. Let’s understand the key terms used in this definition.

- Relational database: Database which stores structured data (data in the form of tables) are termed relational databases.
- Structured language: SQL is a structured language because it deals with structured data. On another side, NoSQL deals with structured, semi-structured, and non-structured data.
- Query: Queries are nothing but various types of operations we perform on the database. It is like a question whose answer is in the database, and we are trying to find that answer.

SQL allows users (data analysts, data scientists, etc.) to perform CURD operations (create, read, update and delete data in a database), create and modify tables and indexes, enforce data integrity rules, and control access to the database.

Why SQL for Data Science?

Search results for “what are the three important programming languages for a data scientist?” list python, R, and SQL. Being a 50-year-old language, SQL is constantly maintaining its place in the top three languages. There are three prominent reasons for that.

1. SQL lies at the core of every company

Most companies use relational databases to store their data and SQL to access, update, and delete it. Tech giants such as Microsoft, Dell, Accenture, Cognizant, and StackOverflow all use SQL to manipulate data. In addition to this, SQL has been standardized by both the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO).

2. SQL integrates well with R and Python

The major advantage of using SQL is that SQL queries can be run in python by setting up connections to the database and using the cursor method. Here is a piece of code to show how we connect to the database using python:

```
def connect_to_database(host, user, password, database_name):
    connection = None
    try:
        connection = mysql.connector.connect(
            host=host,
            user=user,
            passwd=password,
            database=database_name
        )
```

```
        print("Successfully Connected to Database")
    except Error as err:
        print("Error: '{err}'")

    return connection
```

Now, after connecting to the database, we can write queries (the same as we write in SQL) and use the cursor method to execute them.

```
def ExecuteQuery(connection, query):
    cursor = connection.cursor()
    try:
        cursor.execute(query)
        connection.commit()
        print("Successfully executed")
    except Error as err:
        print("Error: '{err}'")

Data_query = """
UPDATE STORES SET STORE_NAME = 'Gurgaon' WHERE STORE_NAME= 'Delhi';
 """

connection = connect_to_database("localhost", "root", pw, db) # Connect to
ExecuteQuery(connection, Data_query) # Execute defined query
```

3. Demand at entry level

Reports show that more than 40% of all data science jobs and 60% of data analyst jobs list SQL as an essential skill. That is why it is necessary to know SQL if one plans to make a career in data science.

Which SQL skills are essential for data science?

Before moving forward, we should understand why we are learning SQL, where we can apply it, and what we need to learn. This section provides a roadmap of the essential topics to focus on when learning SQL.

Relational Database Management Systems(RDBMS)

RDBMS is a set of software tools used to access, update and manipulate data stored in a relational database. This topic covers entity-relationship modeling, joins, ACID property, Normalization, views, database designing, etc.

Knowledge of Data Cleaning

We can't assume that the data we collected or got is perfect, so data cleaning is crucial before any analysis. SQL comes with useful commands like UNIQUE and NOT NULL, allowing the user to store unique and not null values in a table. But what if we have to access a given database containing duplicate and null values?

Null values: By null value, we can say that no value is present in the table for a particular attribute. Our analysis will be inaccurate if we don't take care of them. For example, in a dataset of COVID-19, some of the patient's oxygen level is missing. What should be done in this case? We either delete that record or estimate the missing value or understand the data cleaning steps in detail.

Mismatched datatypes: It's a common problem encountered by data engineers. Let's understand this with an example: Ranking of various cryptocurrencies like Bitcoin, Ethereum, BNB, and Binance should be of integer datatype, but what if it is 3.4 (float)? We can use ALTER command to change the data type of the ranking column.

To perform these data cleaning steps, we must know these commands and basic concepts of SQL.

SQL commands

Some standard SQL commands are SELECT, COUNT, UPDATE, DELETE, INSERT, etc. Knowing these commands is of utmost importance for a data science engineer. SQL is a straightforward language, and if one knows English, then one can learn SQL quickly. For example, let's read and try to

decode this query “SELECT f*name, l*name FROM STUDENT WHERE age > 15”. This query asks to retrieve data of students’ first and last names older than 15 years. Pretty straightforward, right?

Important SQL Commands to Perform Query in Data Science

The first step to mastering any coding language is to get familiar with its syntax and basic concepts. And in SQL, the syntax is not a barrier. So what more is needed to fill the gap between knowing SQL and getting comfortable handling large datasets? It’s practice and a formula that is, before writing any query, study the dataset, find correlations between tables, and visualize the result.

Let’s start by setting up the environment for SQL and then installing a sample dataset. Then we will learn about some important SQL commands and perform queries using them.

Setting Up the environment for SQL

Many relational database systems exist, like MySQL, PostgreSQL, Oracle, and Microsoft SQL. After installing one, we can connect to the database through the GUI or their separate workbenches. We will use Oracle Apex, an online database management system with some inbuilt databases for training and learning purposes. Let’s see how to use it.

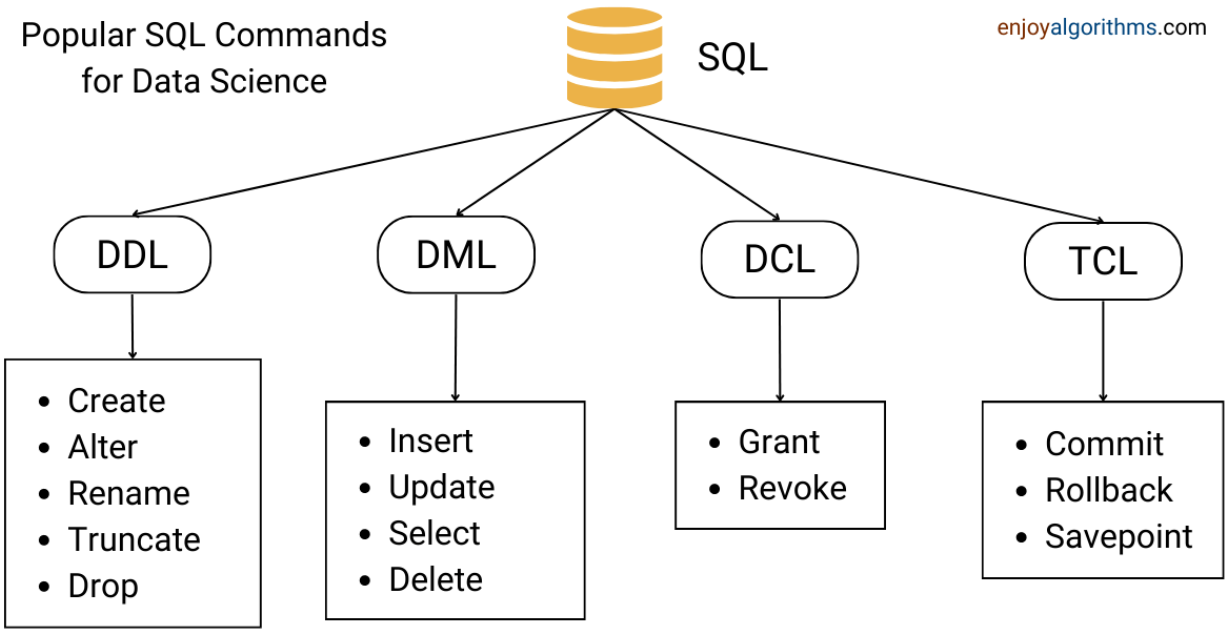
1. Go to oracle apex.
2. Click on the Request a workspace block.
3. Enter your details and wait for 5 minutes. You will receive an email to sign up.
4. Click on SQL workshop and install the customer’s order dataset from sample databases in the utility section.
5. Now you are ready to write SQL queries in the SQL command section.
6. Click on find tables and check for these tables CUSTOMERS, ORDERS, ORDER_ITEMS, PRODUCTS, STORES.

Now we are ready to write SQL queries, but before that, we should know the types of SQL statements.

Types of SQL Commands

SQL commands are mainly grouped into four categories.

1. Data Definition Language(DDL)
2. Data Manipulation Language(DML)
3. Data Control language(DCL)
4. Transactional Control Language(TCL)



Data Definition Language(DDL)

DDL includes commands which are used to define the structure of database objects. These are used to create, modify and delete database structures but not the data itself. Let's see them one by one.

CREATE: Used to create and define datatypes of columns of a table.

```
CREATE TABLE Sale_record(
  sale_id INT PRIMARY KEY,
  DateTime DATE,
```

```
  order_id INT,
  No_sales INT
  );
```

Run these queries in oracle apex's SQL workshop to practice.

ALTER: Used to modify the structure of the table. Sometimes we have to change the datatype of a column, or while feature selection, we find that some more features should be added, so to incorporate these changes, we can use ALTER statement.

```
-- This query alters the INT datatype of No_sales column to FLOAT
ALTER TABLE Sale_record
MODIFY No_sales FLOAT;

-- This query adds a column to table Sale_record
ALTER TABLE Sale_record
ADD (Units INT);
```

RENAME: As the name suggests, it changes the name of the table or column. This command is mainly used to shorten the column names or to change them if more than one table has the same named columns.

```
-- This query changes the column name from No_sales to Sales_no
ALTER TABLE Overall_record
RENAME COLUMN No_sales TO Sales_no;
```

TRUNCATE: Used to delete all the rows of the table. An intern was asked to record some data; by mistake, he/she recorded it all wrong, but one thing he/she did right was used perfect features/attributes/columns. Instead of deleting the whole table, we may use the TRUNCATE command to delete all the rows but preserve the table's structure.

```
TRUNCATE TABLE Overall_record;
```

DROP: This command deletes the table and frees up the space. Continuing the above scenario, now think if the intern could not select the right features and now the data and structure of the table are both incorrect. In this case, we can use the DROP command to delete that table.

```
DROP TABLE Overall_record;
```

These statements were limited to the structure of data. Now we will see commands capable of modifying the data of the table.

Data Manipulation Language(DML)

DML includes commands which are responsible for the manipulation of data. These commands can insert, delete and update data in the database.

INSERT: This command is used to insert data into tables in the form of rows.

```
INSERT INTO STORES
(STORE_ID, STORE_NAME, WEB_ADDRESS, PHYSICAL_ADDRESS, LATITUDE, LONGITUDE,
 LOGO, LOGO_MIME_TYPE, LOGO_FILENAME, LOGO_CHARSET, LOGO_LAST_UPDATED)
VALUES (24,'Delhi','https://www.usabistore.com','Flat 210, street 43,
         vigyan nagar, delhi','','','','','','','');
```

UPDATE: This allows us to modify the data of the column. We all have identity cards, and if your name is difficult to spell, it becomes a headache to get it updated. But the UPDATE command does the same in milliseconds.

```
-- This query updates the name of store to gurgaon from delhi
UPDATE STORES SET STORE_NAME = 'Gurgaon' WHERE STORE_NAME= 'Delhi';
```

SELECT: This command retrieves or selects data from the database. It’s

like the front end of SQL.

```
-- asterisk is used to print whole data of the table
SELECT * FROM PRODUCTS;

-- We can also select columns of our choice whose data you want to retrieve
SELECT PRODUCT_ID NUMBER, PRODUCT_NAME VARCHAR2, UNIT_PRICE FROM PRODUCTS;
```

DELETE: It is used to delete rows from the table. Data deletion is an integral part of data cleaning. Take the case of NULL records, outliers, and flawed data. Removing these is important for better aggregation and visualization.

```
-- This query deletes the row from stores table where store name is delhi.
DELETE FROM STORES WHERE STORE_NAME='Delhi';
```

Now one can retrieve data, remove discrepancies and explore more using these commands. But database management is not only limited to this. We need to provide some permissions and impose restrictions on the users accessing the database.

Data Control Language(DCL)

Commands which deals with granting and revoking access to database comes in this category. These commands ensure the security of data.

GRANT: This is used to give access to various commands to users. Permission to manipulate data should only be given to the management team, not users.

```
GRANT SELECT, UPDATE, DELETE ON PRODUCTS TO 'Nimit'@'localhost;

-- We can also use SHOW command to know what privilages are granted to a us
SHOW GRANTS FOR 'Nimit'@'localhost;
```

REVOKE: It takes back the permissions granted to a user.

```
REVOKE SELECT, UPDATE, DELETE ON PRODUCTS FROM 'Nimit'@'localhost;
```

Transactional Control Language(TCL)

These commands are used to undo or save changes made by DML statements. These ensure that if any failure occurs, we can trace it back to the savepoints.

COMMIT: This command saves all the changes/transactions made to the database since the last commit or rollback.

ROLLBACK is used to undo changes since the last commit or rollback.

SAVEPOINT: These are like those vice-city savepoints where we used to save our mission progress so that we can start our game from these points.

Do you remember the steps mentioned to master SQL? Now it's time to apply them and solve some queries listed below.

Practice Queries

Let's explore the dataset more and answer some questions using SQL queries.

1. How many orders have order status cancelled?
2. Fetch the email address of those customers whose orders get cancelled.
3. Find out the product id and name of those products which got cancelled.

We will perform these queries on the customer dataset downloaded from sample databases on Oracle Apex.

We need to know about the COUNT and WHERE commands to answer

the first question. COUNT gives us the count of records returned by a query. WHERE has the same meaning in English as “to represent conditions.” We have to tell the number of orders not successfully placed.

```
SELECT COUNT(*) FROM ORDERS WHERE ORDER_STATUS='CANCELLED';
```

A store must know how many orders are cancelled and ensure that this will not happen again. To retain the customers, assurance should be made to them. In our following query, we are fetching the email addresses of those customers whose orders got cancelled.

Here we need to fetch information from two tables. Email addresses are present in the CUSTOMERS table, but order status is in the ORDERS table. But these two tables are linked through customer_id, and we will use this to retrieve common records using the WHERE command.

```
SELECT CUSTOMERS.email_address from CUSTOMERS, ORDERS
where CUSTOMERS.customer_id = ORDERS.customer_id
AND order_status='CANCELLED';
```

It is a good practice to use column names with their respective tables (CUSTOMERS.email_address). It will reduce confusion and helps in cases where tables have the same column names.

Now, after sending an assurance mail to customers, what is next? The store owner must know the reason behind the order cancellation. Right? In our following query, we will find which product items need to be shipped correctly.

In this query, we have to fetch information from three tables. Product names and product ids are in the PRODUCTS table, linked with the ORDER*ITEMS table through the product*id and which is further linked with the ORDERS table through order_id.

```
SELECT PRODUCTS.product_id,PRODUCTS.product_name
```

```
from PRODUCTS,ORDER_ITEMS, ORDERS
where ORDER_ITEMS.order_id = ORDERS.order_id
AND ORDER_ITEMS.product_id = PRODUCTS.product_id
AND order_status='CANCELLED';
```

Now you have understood how these SQL commands are used to fetch information from databases.

Conclusion

SQL is a widely used language in data science, and it’s essential to understand it as a tool for analysis, not just querying. On other side, despite technological advancements, relational databases remain at the core of the software industry. Tech giants are constantly modifying SQL to develop their database systems. Microsoft SQL, PostgreSQL, Azure SQL database ledger, and Dune Analytics are some products using SQL as a base.

We hope you enjoyed the blog. Please share your feedback or insights in the message below. Enjoy learning, Enjoy data science! Content moderator: Shubham Gautam.