**Step by Step guide to learn SQL**

1. **Understand the fundamentals:**

Understanding what is SQL, why is it used and where is it used can help you to better understand your purpose of learning SQL. Get familiar with the following terms:

* + What is Data, Database and RDBMS
  + What is SQL and what is the purpose of using it?
  + How data is stored in RDBMS?

***EXERCISE: Install IDE Microsoft SQL Server, MySQL, Oracle etc. – Will share the videos to install SQL.***

1. **SQL Server Installation**
   1. <https://www.youtube.com/watch?v=iaUXjTL_F9U>
2. **Learn the SQL commands:**
   1. **DDL commands**
      1. CREATE
      2. ALTER
      3. DROP
      4. TRUNCATE
      5. **Data type**
         1. VARCHAR
         2. NVARCHAR
         3. INT
         4. DATE
         5. FLOAT
         6. BOOLEAN
      6. **Constraints**
         1. PRIMARY KEY
         2. FOREIGN KEY — Will explain while explaining joins
         3. UNIQUE
         4. CHECK
         5. NULL
         6. Default
         7. Identity
         8. NOT NULL
   2. **DML commands**
      1. INSERT
      2. UPDATE – We will explain clearly after where clause
      3. DELETE -- We will explain clearly after where clause
      4. MERGE command (Which we can use rarely will cover in last).
   3. **TCL commands**
      1. COMMIT
      2. ROLLBACK

***EXERCISE: Check out how table data reacts when using commit and rollback after performing DML operation on your sample tables.***

* 1. **DCL commands -- Optional**
     1. GRANT
     2. REVOKE
  2. **DQL commands** 
     1. SELECT
     2. FROM
     3. WHERE

Once you have learnt the different SQL commands and have practiced creating some sample tables and loaded some data into them, it is time to write SQL Queries.

Start with some basic SQL Queries using the below concepts:

* 1. **SIMPLE SQL QURY** 
     1. SELECT COULMN\_NAMES FROM TABLE\_NAME
  2. **COMPLEX SQL QURY**
     1. SELECT COULMN\_NAMES

FROM TABLE\_NAME JOIN TABLE\_NAMES

WHERE CONDITION (LHS = RHS)

GROUP BY (non aggregated column names)

HAVING CONDITION (Filterations on aggregations)

ORDER BY column\_name asc/desc

* 1. **SYSTEM DATABASE**
     1. Master
     2. Model
     3. MsDB
     4. TempDB
  2. **Aggregate functions** 
     1. MAX
     2. MIN
     3. SUM
     4. AVG
     5. COUNT
  3. **Practice writing some SQL Queries using some of the below operators:**
     1. Comparison Operators like =, <>, != , >, <, >=, ≤
     2. Arithmetic Operators like +, -, \*, /, %
     3. Logical Operators like Operators – AND, OR, IN (NOT IN), BETWEEN (NOT BETWEEN), LIKE (NOT LIKE), IS NULL (IS NOT NULL)
  4. **Set Operators** 
     1. UNION
     2. UNION ALL
     3. INTERSECT
     4. MINUS
  5. **CASE statement**
  6. **DISTINCT and LIMIT/TOP clause**
  7. **ORDER BY**
  8. **Subqueries/Nested Subqueries**
     1. Scalar subquery
     2. Multi row subquery
     3. Correlated subquery
  9. **WITH Clause or CTE Table**
     1. Non-Recursive
     2. Recurring CTE
  10. **GROUP BY, HAVING clause**
  11. **Use aggregate functions with and without group by.**
      1. It is not mandatory to use aggregate functions with GROUP BY always so check out the different queries where you may need to use group by and where you may not need it.
  12. **Different JOINS.**
      1. Inner Join
      2. LEFT Outer Join
      3. RIGHT Outer Join
      4. FULL Outer Join
      5. Self-Join
      6. Cross Join.
      7. Handling Data Retrieval with EXISTS and NOT EXISTS Operators
  13. **Inbuilt Date conversion functions**
      1. GETDATE()
      2. DateDiff
      3. DateAdd
      4. DatePart
      5. Datesub
      6. DATENAME
      7. Write queries using inbuilt date conversion functions like to\_date or date\_format or cast etc.
  14. **Inbuilt String conversion functions**
      1. SubStr
      2. Charindex
      3. Instr
      4. Write queries using inbuilt string functions to fetch part of a text from data such as substring, position, instr etc.
  15. **There would be no of functions for different SQL operations.**

>> Like to add lead zero’s

* + 1. Lpad
    2. Rpad
    3. Trim
    4. Ltrim
    5. Rtrim
    6. …..
  1. **Window functions/ Analytical Functions – Over clause**
     1. row\_number()
     2. rank()
     3. dense\_rank()
     4. first\_value()
     5. last\_value()
     6. lead()
     7. lag()
     8. nth\_value()
     9. ntile()
     10. percent\_rank()
     11. cume\_dist()
     12. Probably one of the best features to solve complex queries are window functions so definitely spend some time to explore the most widely used window functions
     13. Over Clause can be used on aggregate functions also
  2. **Views**
     1. A view can be thought of as either a virtual table or a stored query.
     2. Simple and Complex View

**------------------------------------------ PL SQL --------------------------------------------------**

* 1. **Stored Procedures**
  2. **Functions**
     1. Table variable (Create type table @tablename(col1 datatype, col2 datatype… coln))
  3. **Indexes**
     1. Clustered Index
     2. Non-Clustered Index
  4. **Triggers**
  5. **Cursors**
  6. **Pivot and Unpivot**
  7. **Merge**

**Learning Resources**

* [W3Schools](https://www.w3schools.com/sql/) - for SQL syntax
* [LearnSQL.com](https://learnsql.com/?ref=thoufiqmohammed) - for SQL course
* [StrataScratch](https://www.stratascratch.com/?via=techTFQ) - for practicing basic, intermediate and complex SQL queries
* [LeetCode](https://leetcode.com/) - for practicing basic SQL queries