**SQL Notes**

SQL is a structured query language that involves using relational databases, which requires pre-defined schemas.

Table

Description automatically generated

CRUD Operations are most used in SQL. (Create, Read, Update, Delete)

SQL statements are divided into 4 main categories:

1. Data Definition Language (DDL)

* CREATE (Create new database/table)
* ALTER (Modify database/table structure)
* DROP (Delete database or table)
* TRUNCATE (Remove table records)

1. Data Manipulation Language (DML)

* SELECT (Retrieve data from table)
* INSERT (Insert new records into table)
* UPDATE (Update existing records from a table)
* DELETE (Remove existing records from a table)

1. Data Control Language (DCL)

* GRANT (Assign privilege to users for accessing data)
* REVOKE (Remove privilege to users for accessing data)

1. Transaction Control Statement (TCS)

* COMMIT (Permanent work save into database)
* ROLLBACK (Restore database to previous form since last commit)
* SAVEPOINT (Create save point for future rollbacks)
* SET TRANSACTION (Setting transaction to read-write/read only access)

**SQL Query Basic Template**

SELECT <column\_names>

FROM <table\_name>

WHERE <condition on columns from table>

GROUP BY <column\_names>

HAVING <condition on grouped-by columns>

ORDER BY <column\_names>

SQL works in the following order of statement execution:

1. Gathers all data with FROM clause
2. Filters the data with WHERE clause
3. Groups rows together with GROUP BY clause
4. Filters grouped rows with HAVING clause
5. Specifies columns to display with SELECT clause
6. Sorts the results with ORDER BY clause

Types of RDBMS and its database tools:

1. SQLite: DB Browser for SQLite
2. MySQL: MySQL Workbench
3. Oracle: Oracle SQL Developer (PL/SQL – Procedural Language Extension to SQL)
4. PostgreSQL: pgAdmin
5. SQL Server: SQL Server Management Studio (T-SQL – Transact SQL)

ANSI (American National Standards Institute) standard in SQL refers to SQL code that will run in any RDBMS software.

**Identifiers vs Aliases**

Identifiers are name of database objects

Aliases rename column or table temporarily, mostly useful in subqueries.

**Statements vs Clauses**

Statements are blocks of code that starts with a SQL keyword like SELECT and ends with a semicolon.

Clauses are specific sections of the statement that refers to specific SQL keywords like WHERE, FROM etc.

**Single vs Multi-Line Comments**

Single Line Comment:

-- This is a single line comment

Multi Line Comment:

/\* These are

multi line comments \*/

**Single vs Double Quotes**

Single quote: Used for string reference

Double quote: Used for identifier reference

**Wildcard expressions**

%: Represents any n number of characters

\_: Represents any single character

Note that these wildcard expressions are used together with LIKE or NOT LIKE keyword in WHERE clause.

**SQL Data Types**

1. Numeric (INT, DECIMAL, FLOAT)

* INT: Used for values that do not allow for decimals (i.e. 45)
* DECIMAL: Used for fixing number of decimals (i.e. 24.524)
* FLOAT: Store limited number of decimals with power notation (i.e. 2.4524 \* 10^5)

Note: MySQL has the option of setting numeric variables as positive only using UNSIGNED keyword.

1. String (CHAR, VARCHAR, TEXT, NCHAR, NVARCHAR)

* CHAR: Stores fixed length of characters as ASCII data
* VARCHAR: Stores maximum length of specified characters as ASCII data
* TEXT: Used for storing long strings of text like paragraphs
* NCHAR: Stores fixed length of characters as Unicode data
* NVARCHAR: Stores maximum length of specified characters as Unicode data

Note: Unicode data refers to non-ASCII characters (non-English mostly)

1. Date (DATE, TIME, DATETIME, TIMESTAMP, YEAR – For MYSQL)

* DATE: YYYY-MM-DD
* DATETIME: YYYY-MM-DD hh:mm:ss
* TIMESTAMP: YYYY-MM-DD hh:mm:ss UTC
* TIME: hh:mm:ss
* YEAR: YYYY

Note: DATETIME variable type does not store time zone, while TIMESTAMP variable type does store time zone.

1. Boolean (BOOLEAN): FALSE value as 0 and TRUE value as 1

Note: Boolean variable type is currently not supported in Oracle and SQL server.

1. External files like images, documents etc.

Approach 1: Store links to files using VARCHAR variable type

Approach 2: Convert files to binary format and store files using BLOB variable type

**Common SQL Operators**

**Logical Operators**

**AND**: Returns TRUE if both conditions are true or otherwise

**OR**: Returns TRUE if either condition is true or otherwise

**NOT**: Returns TRUE if condition is FALSE or otherwise

**Comparison Operators (Used in conditional statements**)

**=**: Equality

**!=**, **<>**: Inequality

**<**: Less than

**<=**: Less than or equal to

**>**: Greater than

**>=**: Greater than or equal to

**BETWEEN**: Checks if value lies within given range (inclusive)

**EXISTS**: Checks if row exist in subquery (also known as semi-join)

**IN**: Checks if value is contained within a list (Note that having a single null value within the list will always result in FALSE for conditional statements. Thus, recommend using **NOT EXISTS** keyword instead.)

**IS NULL**: Checks if a value is null

**IS NOT NULL**: Checks if a value is not null

**LIKE**: Checks if value matches a simple pattern (Used with wildcard expressions)

**Math Operators**

+: Addition

-: Subtraction

\*: Multiplication

/: Division

%: Modulo (remainder)

&: Bitwise AND

|: Bitwise OR

^: Bitwise XOR

**Aggregate Functions**

Aggregate functions perform calculations on multiple rows to return a single result.

Aggregate functions are mostly used in SELECT and HAVING clause

**COUNT()**: Counts number of values for a given attribute

**SUM()**: Sum of values for a given attribute

**AVG()**: Average of values for a given attribute

**MIN()**: Minimum value for a given attribute

**MAX()**: Maximum value for a given attribute

**LEAST()**: Minimum value from set of attributes for every row

**GREATEST()**: Maximum value from set of attributes for every row

Note that most of these aggregate functions in SQL applies to only non-null values, except for COUNT() function that applies to both null and non-null values.

**Numeric Functions**

Numeric functions are usually applied to attributes with numerical values only.

Math functions:

**ABS(x):** Returns absolute value of x

**SIGN(x):** Returns sign of value of x (-1 for negative, 0 for zero or 1 for positive)

**POWER(x,y):** Returns x to the power of y

**SQRT(x):** Returns square root of x

**EXP(x):** Returns exponent of x

**LOG(y,x):** Returns log of y base x

**LN(x):** Return natural log of x base e

**LOG10(x):** Returns log of x base 10

**MOD(x,y):** Returns remainder of x/y

**RAND():** Returns a random number between 0 and 1

**CEIL(x):** Returns upper integer boundary of value x

**FLOOR(x):** Returns lower integer boundary of value x

**ROUND(x,n):** Rounds x value to n decimal places

**TRUNC(x,n):** Cuts off x value at n decimal places without rounding

**String Functions**

String functions are usually applied to attributes with string data types.

**LENGTH(string):** Returns number of characters of a string

**UPPER(string):** Returns upper case characters of the whole string

**LOWER(string):** Returns lower case characters of the whole string

**TRIM(string):** Removes trailing and leading whitespaces from string

**TRIM(char FROM string):** Removes trailing and leading specified char from string

**LTRIM(string):** Removes leading whitespaces from string

**RTRIM(string):** Removes trailing whitespaces from string

**CONCAT(string1, string2, …):** Concatenate multiple strings into a single string

**SUBSTRING(string, start, length):** Returns a portion of string starting from start index for n length.

**REPLACE(string, old\_substring, new\_substring):** Replace old\_substring from a given string with new\_substring.

**REGEXP ‘Regular\_Expression’:** Search for regular expression pattern

Note: Regular expression syntax is best tested with the following website, instead of memorizing syntax: <https://regex101.com/>

**Datetime Format Specifiers**

|  |  |  |  |
| --- | --- | --- | --- |
| %Y | 4-digit year | %d | Day (1-31) |
| %y | 2-digit year | %h | 12 hours (1 – 12) |
| %m | Numeric month (1-12) | %H | 24 hours (1 – 24) |
| %b | Abbreviated month (Jan – Dec) | %i | Minutes (0-59) |
| %M | Name of month (January – December) | %s | Seconds (0-59) |

Note that datetime functions for different SQL engines have different syntax used. Thus, it is recommended to google it rather than memorizing syntax.

**Null Functions**

Null functions are triggered when there is a missing value from attributes and an alternative output is provided as part of the function.

**COALESCE(attribute, alternative):** Returns “alternative” if null value exist in attribute