SQL Workshop

A3SR Stats Club

Getting Started

Download DBeaver v5.2.4

Download chinook.db

Load the Data in

Intro

<u>Relational Databases:</u>

A database structured to recognize relations among stored items of information.

ORACLE DATABASE
PostgreSQL
Microsoft SQL Server
TERADATA
MariaDB
MySQL
IBM DB2
SYBASE
H2
Access
Apache Derby
HIVE
SQLite
HyperSQL

<u>Other types:</u>

- Non-relational Databases
- Graph Databases
- Etc.

Why use databases?

Save space:

- Optimized data structure (schema)
- Facts Tables:
    - events
- Dimensions Tables:
    - attributes

Save Time:

- Indexing
    - Filter (subsetting)
    - Aggregates (grouping)
    - Joins (merging)

Terminology

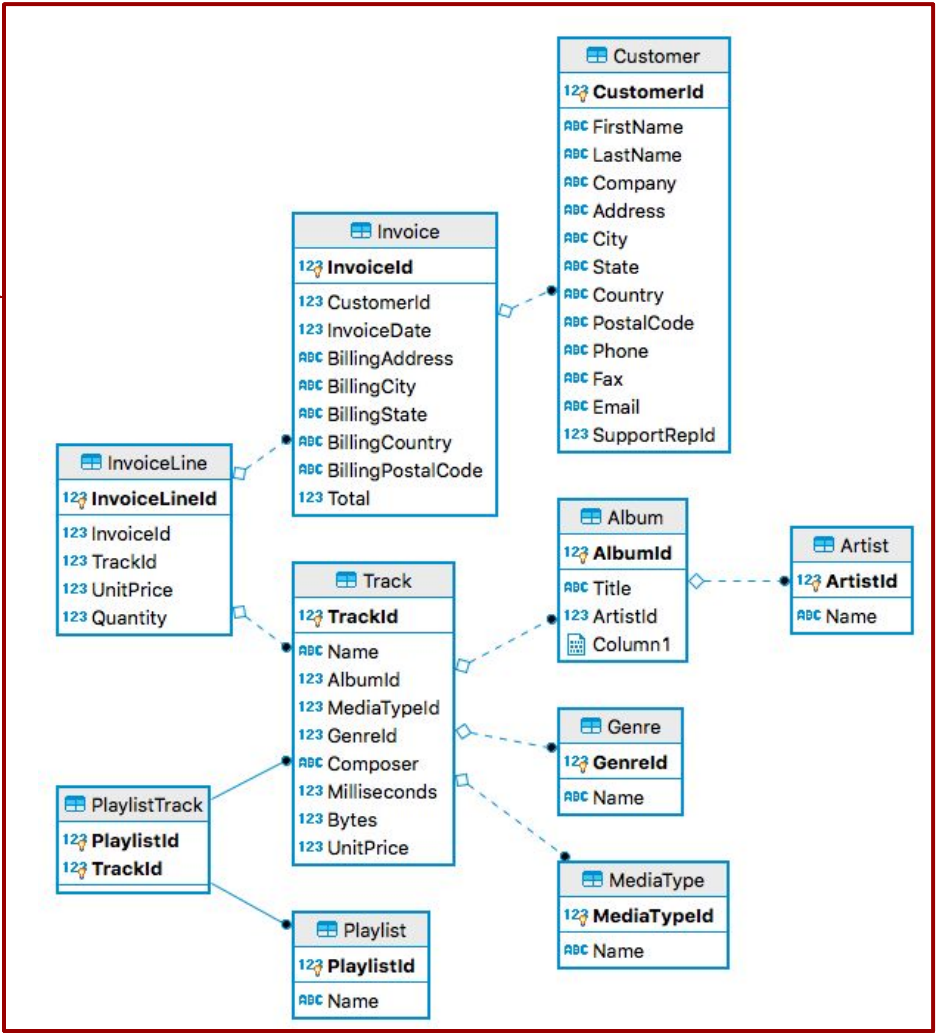
Schema

Tables

Columns

Primary Key (PK)

Foreign Key (FK)



Terminology

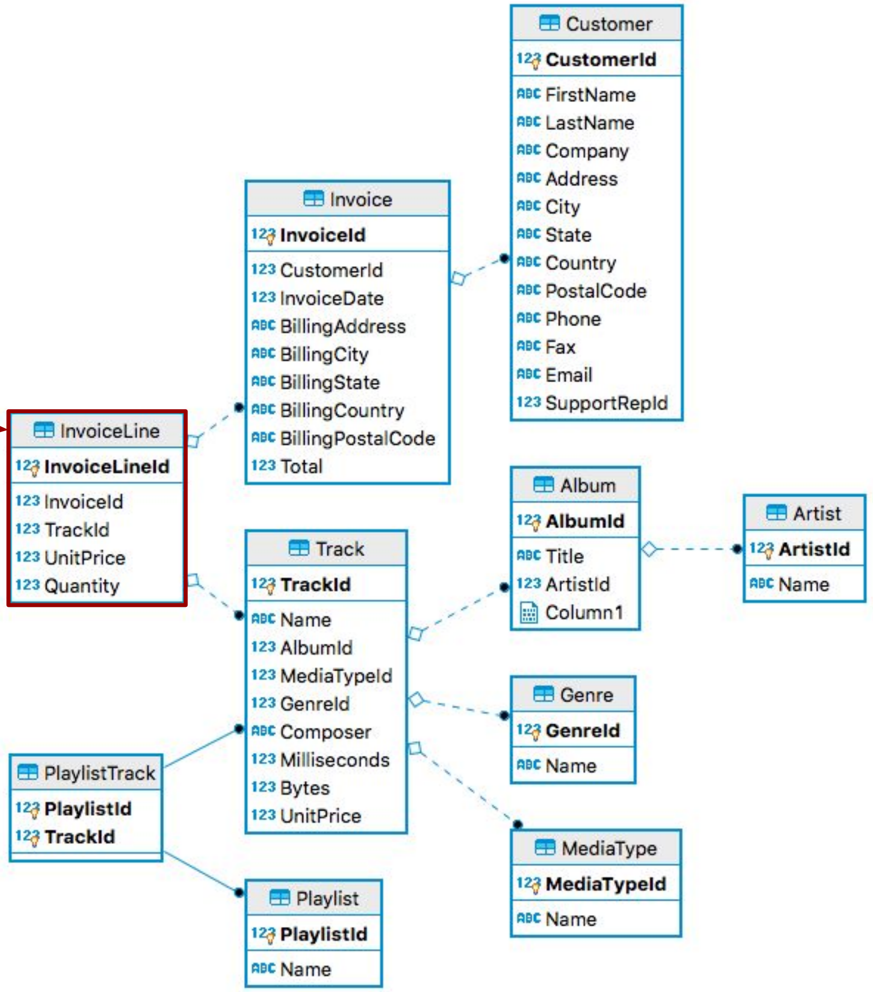
Schema

Tables

Columns

Primary Key (PK)

Foreign Key (FK)



Terminology

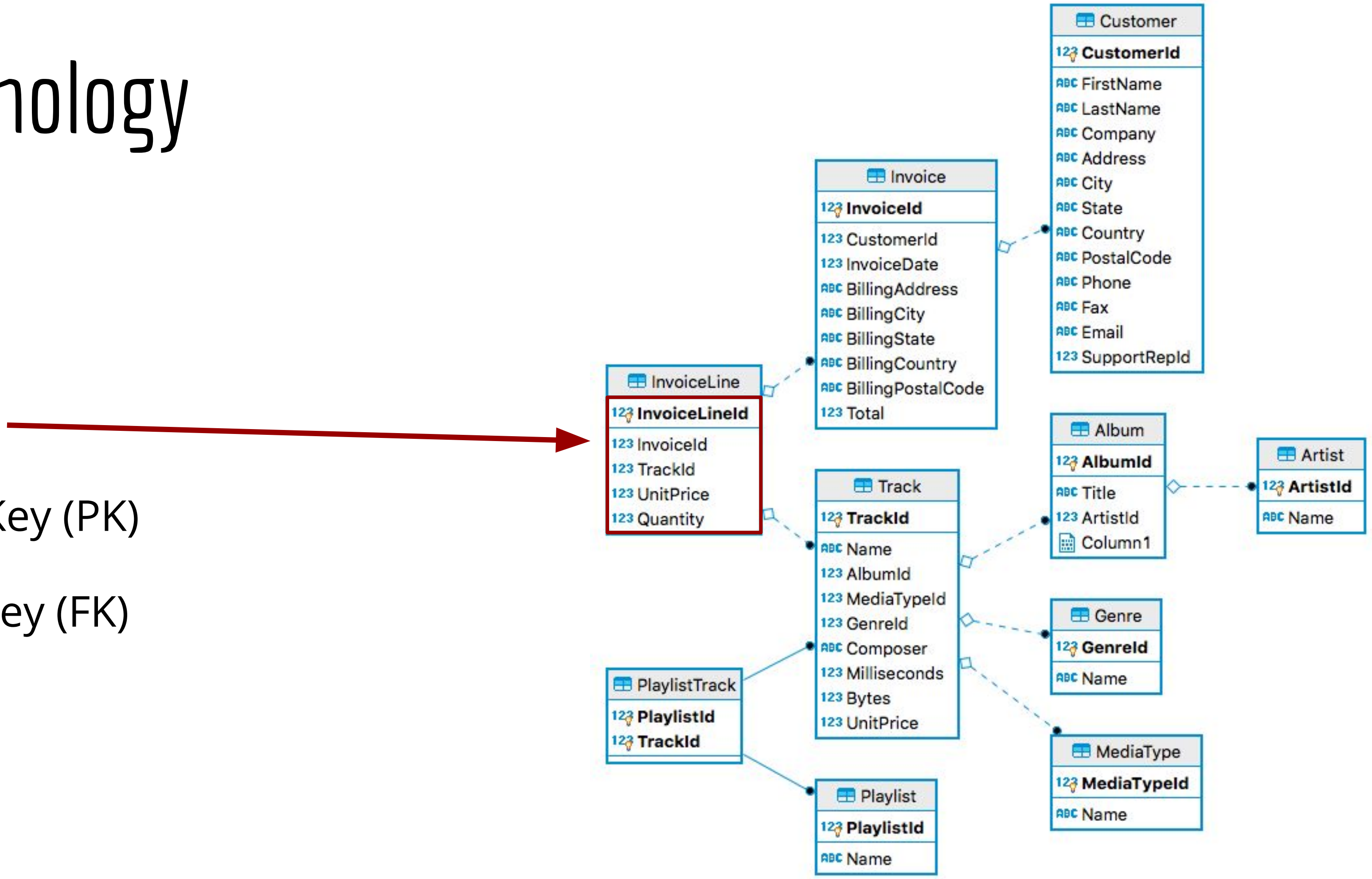
Schema

Tables

Columns

Primary Key (PK)

Foreign Key (FK)



Terminology

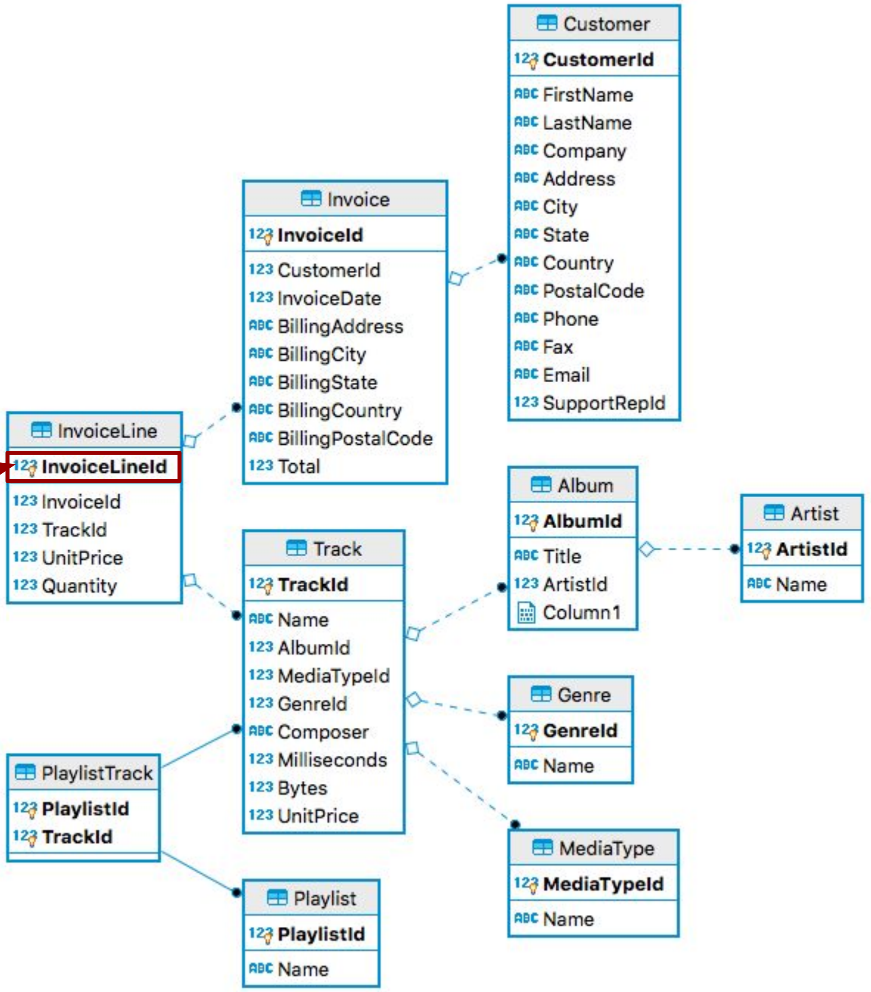
Schema

Tables

Columns

Primary Key (PK)

Foreign Key (FK)



Terminology

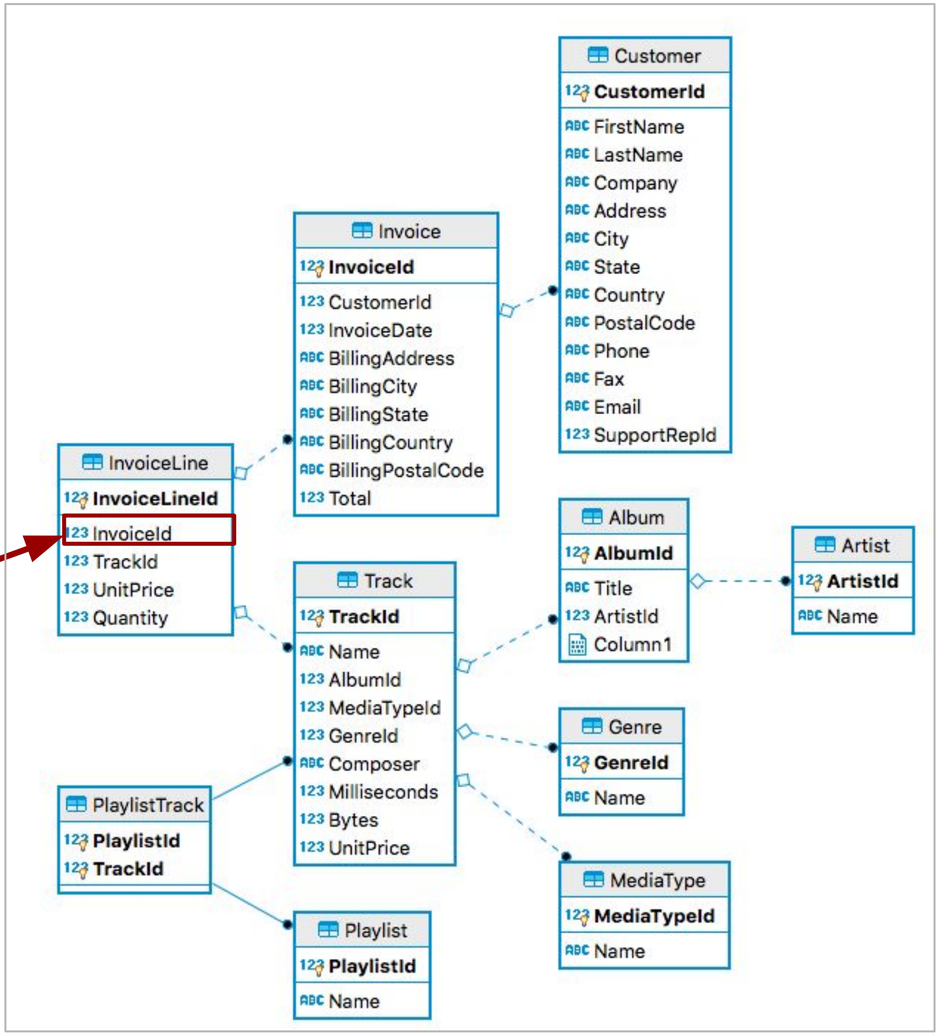
Schema

Tables

Columns

Primary Key (PK)

Foreign Key (FK)



Syntax : ‘Querying’ the tables

SELECT *columns* FROM *table* ;

SELECT * FROM *table* ;

SELECT * FROM *table* t ; (*t: Alias for the query)*

SELECT * FROM *table* LIMIT 5 ;

SELECT * FROM *table* ORDER BY *column ;*

SELECT * FROM *table* ORDER BY *column* DESC *;*

Syntax: ‘Filtering’ the table

SELECT *columns* FROM *table* WHERE *column* = x ;

SELECT *columns* FROM *table* WHERE *column1* = x AND *column2* < y ;

SELECT DISTINCT *column* FROM *table ;*

SELECT columns FROM table WHERE column BETWEEN x AND y ;

<u>Notes:</u>

Conditional Operators: = , != , > , < , >= , <=, etc.

Casting: CAST(*column* as datatype) [eg: INT, CHAR]

Syntax: ‘Aggregating’ the table

SELECT COUNT(*) FROM *table* ;	Counts the rows
SELECT AVG*(column)* FROM *table* ;	Average value
SELECT SUM*(column)* FROM *table* ;	Total value of the rows
SELECT MIN*(column)* FROM *table* ;	Min value in the column
SELECT MAX*(column)* FROM *table* ;	Max value in the column

Syntax: ‘Grouping’ by values in a column

SELECT column, COUNT(*) FROM *table* GROUP BY *column* ;

SELECT column, COUNT(*) FROM *table* GROUP BY *column* HAVING *column* > *x* ;

Exactly the same idea as dplyr!

Order of Operations: Structuring the query

SELECT	*some columns, aggregates*
FROM	*some table*
WHERE	*condition(s) satisfied*
GROUP BY	*variables in a column*
HAVING	*some aggregate condition satisfied*
ORDER BY	*a variable*
LIMIT	*the results*

Exercise-1a

1. How many tables are in your Schema?

2. What are the columns of the InvoiceLine Table?

3. Now looking at the Track table: what are the Primary Keys and Foreign?

4. What are the 3 shortest tracks (in milliseconds)?

5. Let's look at songs longer than 300000 ms. What is the title of the song with the smallest bytes?

Exercise-1b

6. Let's look at the Invoice table. What is the total number of orders?

7. How many orders were made in 2008? Use ‘between’

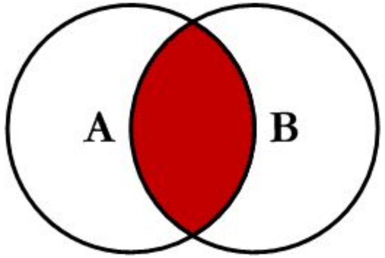
8. Let's look at the Customer table. What is the number of unique countries that customers come from?

9. How many customers come from Canada?

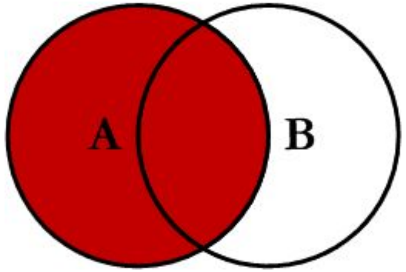
10. From the Tracks table, find the GenreID corresponding to the Genre with the highest average track length accounting only those tracks greater than 30secs. With this information of the GenreID, find the name of the Genre from the Genre table.

Joins

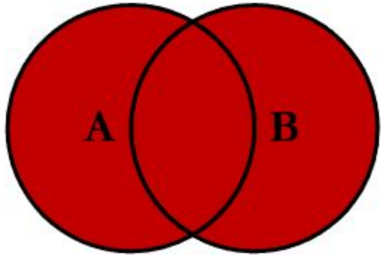
JOIN (INNER JOIN)



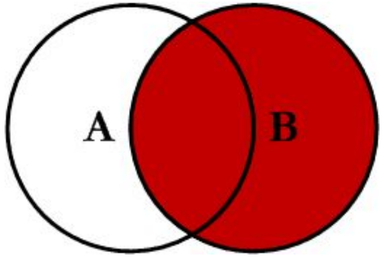
LEFT JOIN



OUTER JOIN



RIGHT JOIN



Syntax: ‘Joining’ multiple tables

SELECT *column_table1, column_table2* FROM *table1* JOIN *table2* USING(*primary_key*);

SELECT *column_table1, column_table2* FROM *table1* JOIN *table2* ON *primary_key_table1 = foreign_key_table2*

SELECT *column_table1, column_table2* FROM *table1* OUTER JOIN *table2* USING(*primary_key*);

SELECT *column_table1, column_table2* FROM *table1* RIGHT JOIN *table2* USING(*primary_key*);

Exercise-2

1. Which album by Metallica is divided in two discs ?
2. How many songs are in each of the top 3 playlist (provide names of playlists)?
3. How much did Martha Silk spend on songs in total?
4. How many playlists include songs by Iron Maiden?

Advanced SQL

Views: WITH (...) AS *xyz*

Window Functions:

Aggregate to Arrays:

Geospatial querying with PostGIS & PostgreSQL:

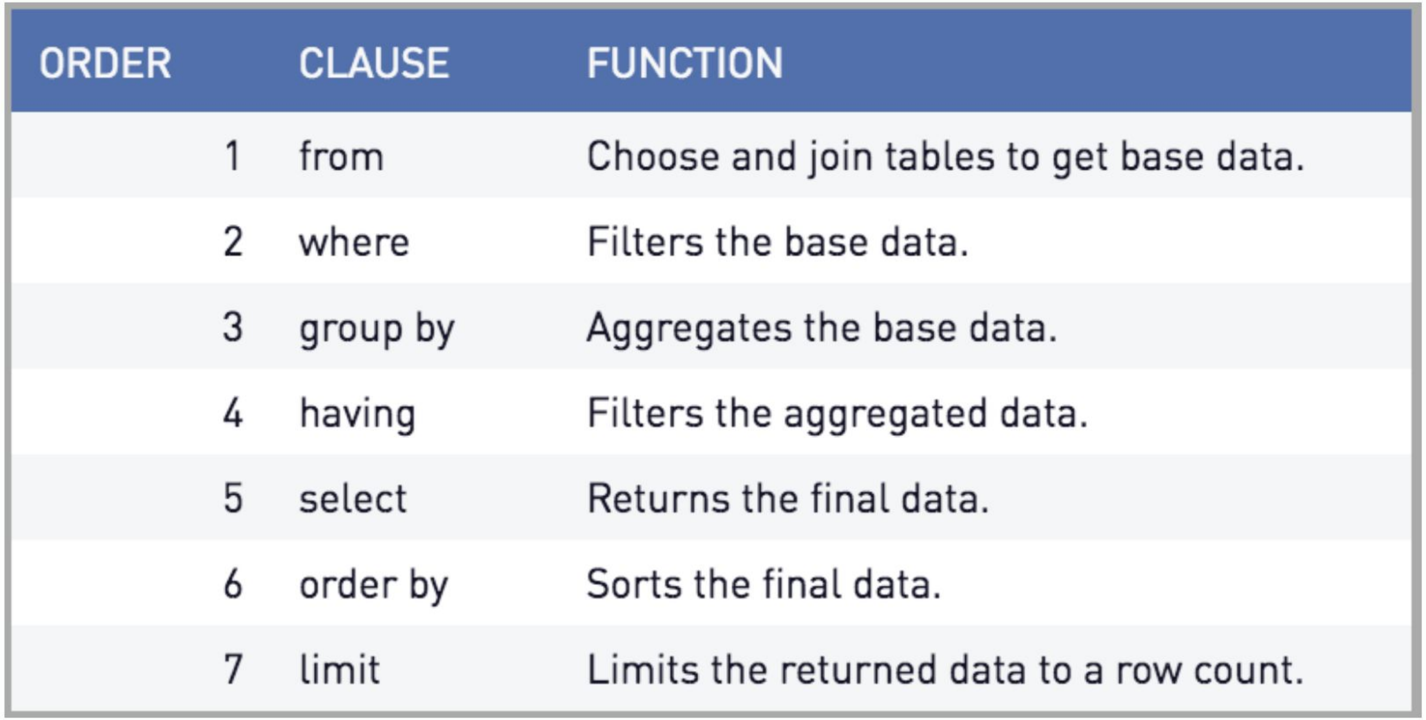
Best Practices

Indexing: Speeds up querying immensely on the columns that you expect to filter and group by often

Order of Joins: Smallest to largest. Use views to filter before joining.

Test before you run: Use limit to restrict size to test your query on before running it on the full table

Order of Execution (behind the scenes)



ORDER	CLAUSE	FUNCTION
1	from	Choose and join tables to get base data.
2	where	Filters the base data.
3	group by	Aggregates the base data.
4	having	Filters the aggregated data.
5	select	Returns the final data.
6	order by	Sorts the final data.
7	limit	Limits the returned data to a row count.

HAPPY BDAY GEORGE