

Intro to Database and SQL

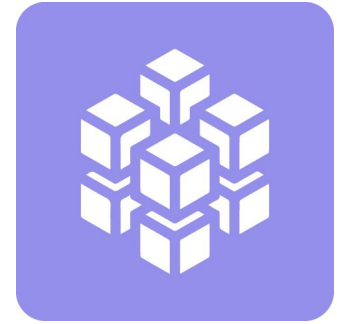
Relevel
by Unacademy



What is Data Management

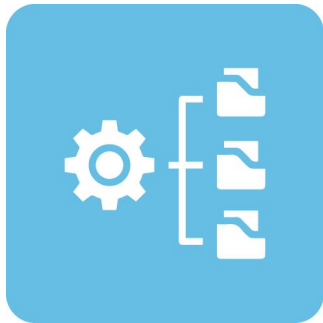
Data management is the process of collecting, stockpiling, organising, and safeguarding data so that it may be conveniently retrieved and evaluated for improved business results. Data management solutions are crucial for preserving information while assuring accessibility as firms develop and gather data at ever-increasing rates.

Data management is critical for businesses of all sizes. Businesses can successfully manage and analyse one of their most important resources by having trusted access to data safely and securely, allowing them to make educated decisions, optimise operations, and cut expenditures while growing revenue and profits.



Structured Vs Unstructured Data

Data that conforms to a predetermined data model is referred to as **Structured Data**. It's simple to map into designated fields. A US ZIP code (for example, 90210) can be saved as a five-digit string, a State as a two-character abbreviation (for example, CA), and so on. In a typical relational database, structured data is easily stored and retrieved.











There is no specified data model for unstructured data. As a result, it's more difficult to categorise into the preset tables and rows of a relational database. Standard features may exist in satellite photos, audio files, video files, or communications.

Structured Vs Unstructured Data

	Structured Data	Unstructured Data
Characteristics	<ul style="list-style-type: none">• Pre-defined data models• Usually text only• Easy to search	<ul style="list-style-type: none">• No pre-defined data model• May be text, images, sound, video or other formats• Difficult to search
Resides in	<ul style="list-style-type: none">• Relational databases• Data warehouses	<ul style="list-style-type: none">• Applications• NoSQL databases• Data warehouses• Data lakes
Generated by	Humans or machines	Humans or machines
Typical applications	<ul style="list-style-type: none">• Airline reservation systems• Inventory control• CRM systems• ERP systems	<ul style="list-style-type: none">• Word processing• Presentation software• Email clients• Tools for viewing or editing media
Examples	<ul style="list-style-type: none">• Dates• Phone numbers• Social security numbers• Credit card numbers• Customer names• Addresses• Product names and numbers• Transaction information	<ul style="list-style-type: none">• Text files• Reports• Email messages• Audio files• Video files• Images• Surveillance imagery

Examples of Unstructured Data

Unstructured data types

 Text files and documents	 Server, website and application logs	 Sensor data	 Images
 Video files	 Audio files	 Emails	 Social media data

Examples of Structured Data

CUSTOMER_ID	LAST_NAME	FIRST_NAME	STREET	CITY	ZIP_CODE	COUNTRY
10302	Boucher	Leo	54, rue Royale	Nantes	44000	France
11244	Smith	Laurent	8489 Strong St	Las Vegas	83030	USA
11405	Han	James	636 St Kilda Road	Sydney	3004	Australia
11993	Mueller	Tomas	Berliner Weg 15	Tamm	71732	Germany
12111	Carter	Nataly	5 Tomahawk	Los Angeles	90006	USA
14121	Cortez	Nola	Av. Grande, 86	Madrid	28034	Spain
14400	Brown	Frank	165 S 7th St	Chester	33134	USA
14578	Wilson	Sarah	Seestreet #6101	Emory	1734	USA
14600	Tanen	John	71 San Diego	Los Angeles	90004	USA

Various modes of data storage

Data is primarily introduced in three sources :

- Database
- Data Warehouse
- Data Lake



Intro to Database

A database is a storage location that stores, search, and report on structured data from a single source.

For organizations, the use cases for databases include:

- Creating reports for various business functions such as Business review reports, sales reports
- Automating business processes
- Auditing data entry

Popular databases are:

- Oracle
- PostgreSQL
- MongoDB
- Redis
- Elasticsearch
- Apache Cassandra



Intro to Data Warehouse

A data warehouse is used to store large amounts of structured data from multiple databases in a centralized place.

Organizations invest in building data warehouses because of their ability to deliver business insights from across the company, and quickly. Data Warehouse is a combination of multiple databases.

Popular companies that offer data warehouses include:

- Snowflake
- Yellowbrick
- Teradata



Intro to Data Lake

A data lake stores structured, semi-structured and unstructured data, supporting the ability to store raw data from all sources without the need to process or transform it at that time.

Only when the data needs to be retrieved will some structure be applied. Storing data in data lakes is much cheaper than in a data warehouse. Data lakes are very popular in the modern stack because of their flexibility and costs, but they do not replace data warehouses or relational databases.

Popular data lake companies are:

- Hadoop
- Azure
- Amazon S3



Difference between Database, Data Warehouse & Data Lake?

Data Storage Comparison

Key benefits & drawbacks of data storage types

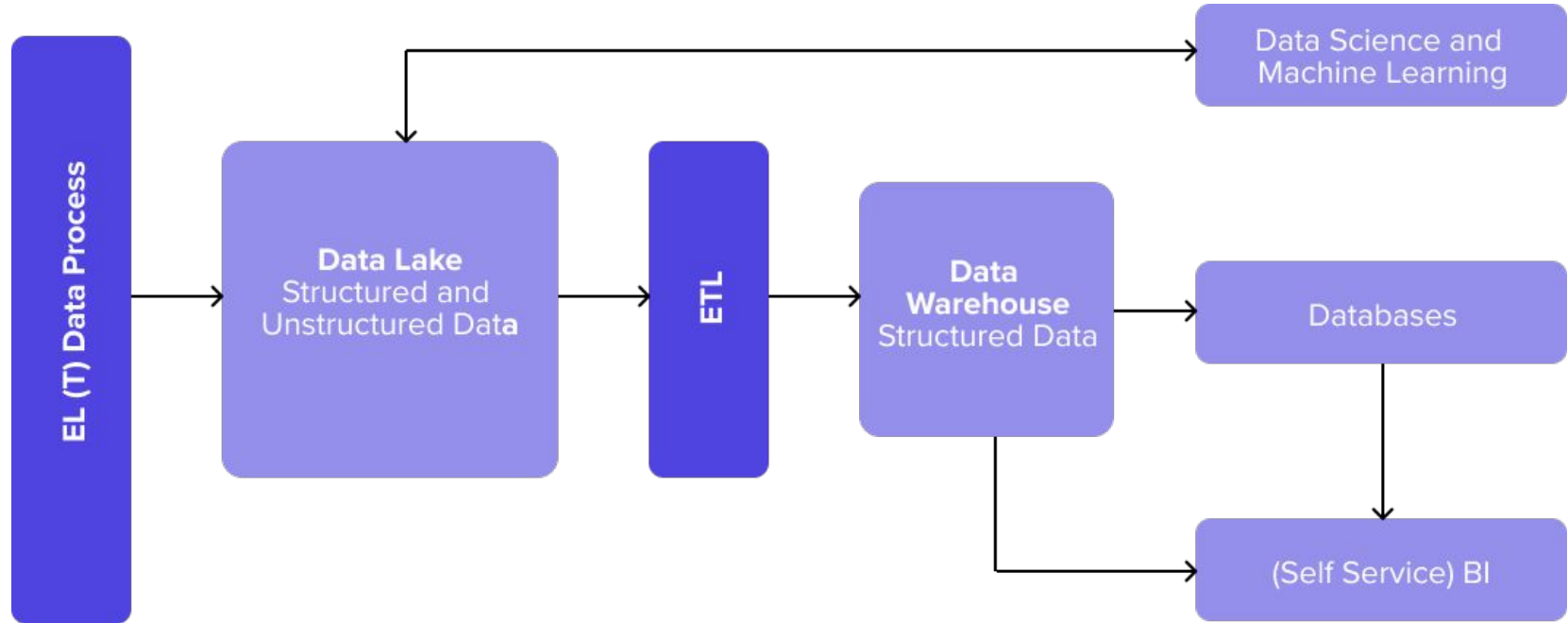
	Database	Data Warehouse	Data Lake
Data	Structured	Structured	Raw & unstructured
Processing	Schema-on-write	Schema-on-write	Schema-on-read
Cost	Free to \$	\$\$\$	\$
Agility	Varies	Minimal	Maximum
Security	Immature	Mature	Immature
Users	Anyone	IT/business users	Data scientists
Use cases	Reporting, analysis & automation	Machine learning	Data science & research

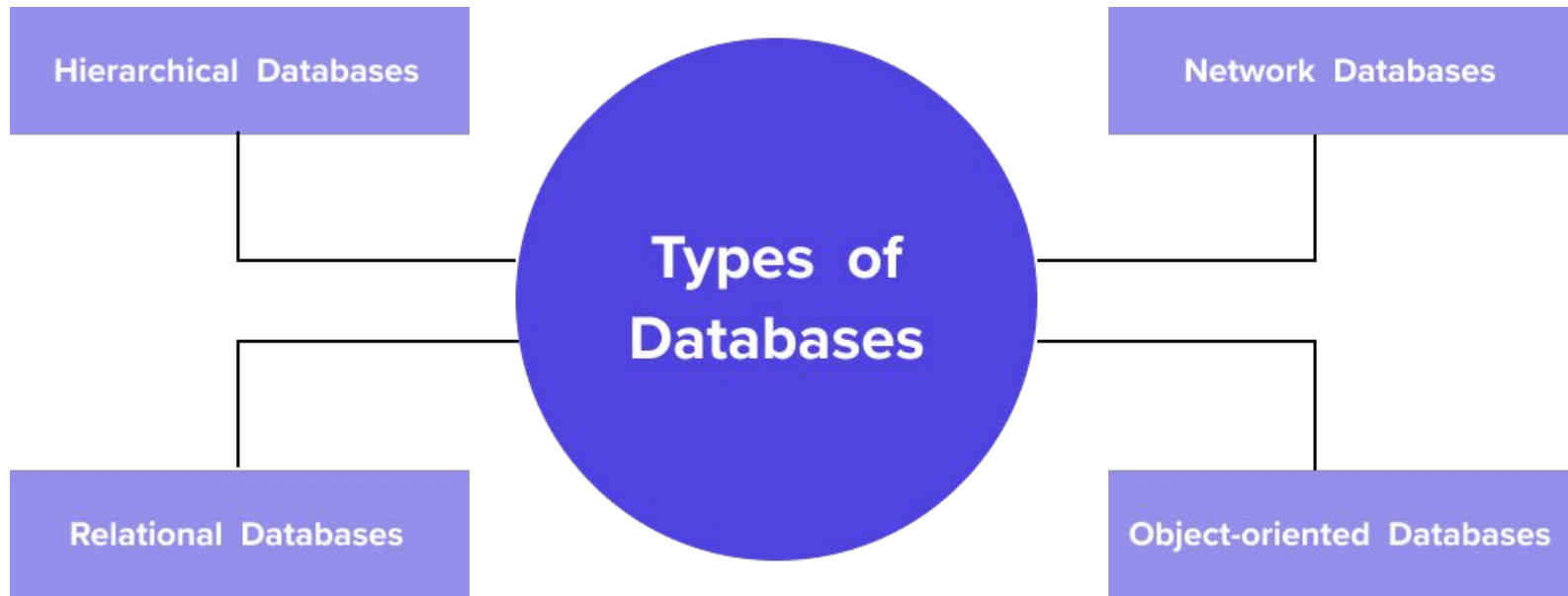
How does data flow happen for a product?

For a product, data flows in three steps:

- **Data Logging:** This step is taken care of by the engineering team. At the time of product development, the Engineering team consults the data engineer, Product Manager, and Data Science/Data analyst to decide what features to log. They provide a data dump(unstructured data) through PHP/javascript. The information is generally stored in Data Lake.
- **Data transformation:** In this step, Data Engineers transform unstructured data to structured data through the ETL(Extract, Transform, Load) process and store it into a databases/Data Warehouse. The development of databases is done in accordance with the business needs.
- **Data to Meaningful Insights:** Data analysts/scientists will use databases for metrics development/analysis and provide actionable insights to improve product

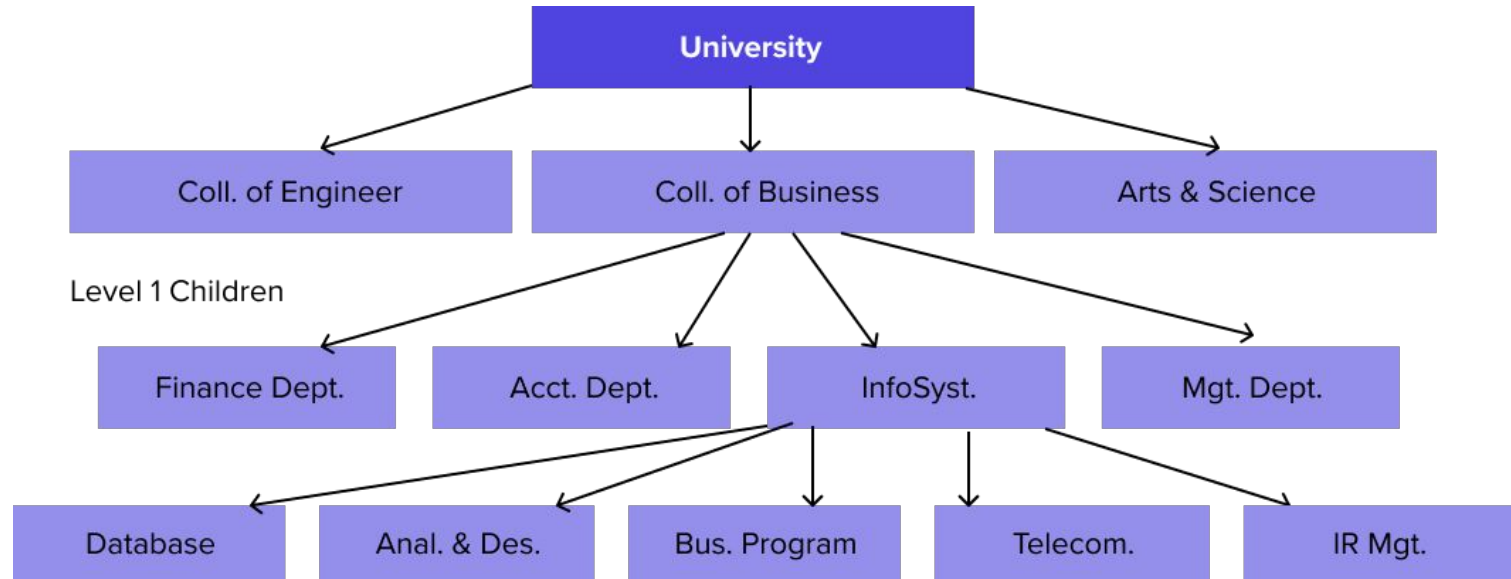
A typical data logging flow





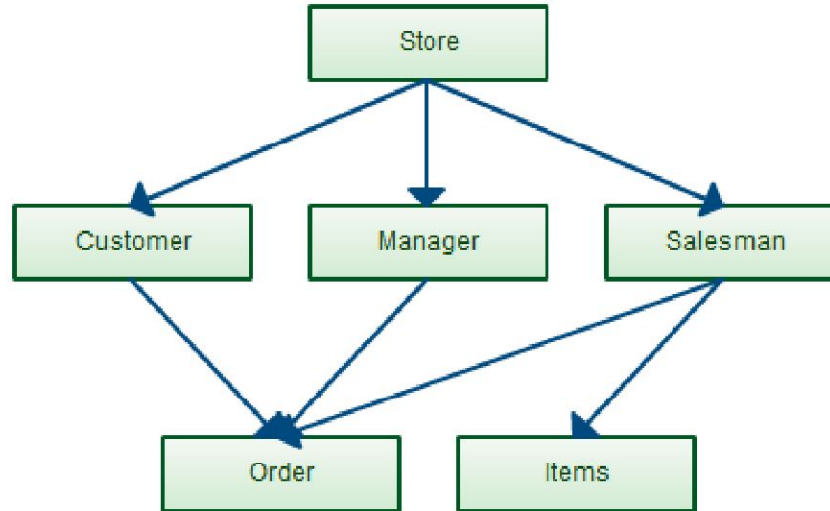
Hierarchical Database

A hierarchical database is a data architecture in which data is organized into a tree-like or parent-child structure. One parent node can have numerous child nodes connected through links.



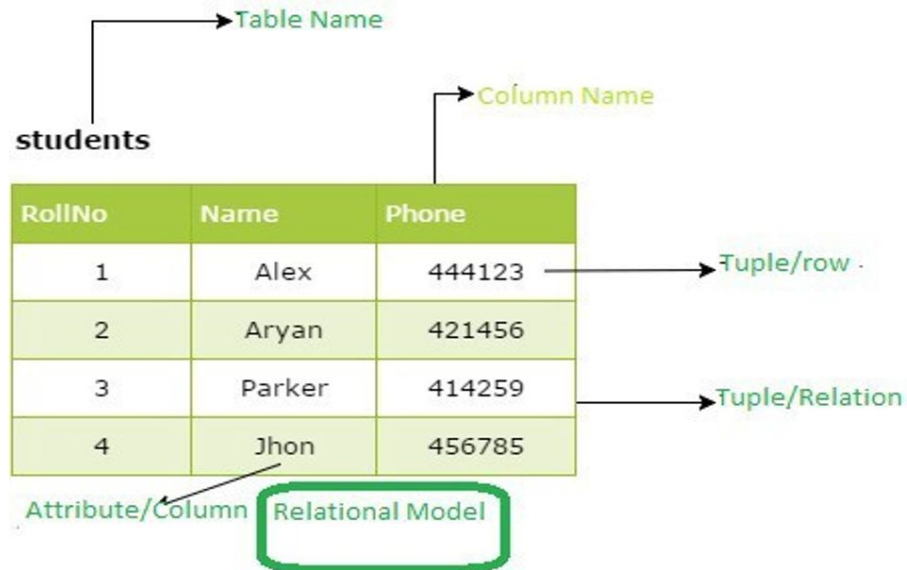
Network Database

A network database is a database model in which several member records or files can be linked to different owner files and vice versa.



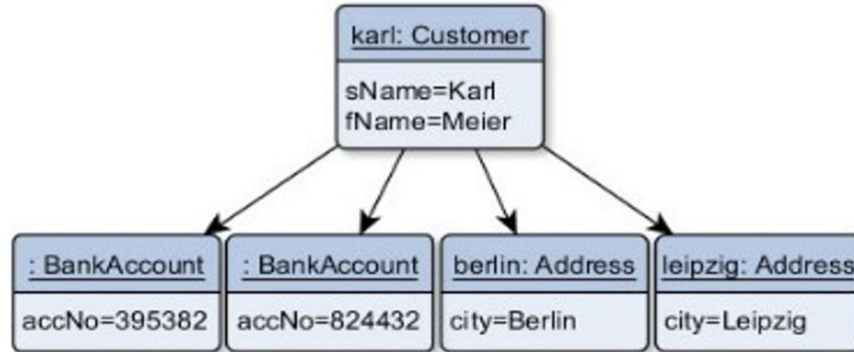
Relational Database

A relational database is a sort of database that employs a structure that enables us to locate and access information about other data in the database. The rows and columns in this database are organized into tables.



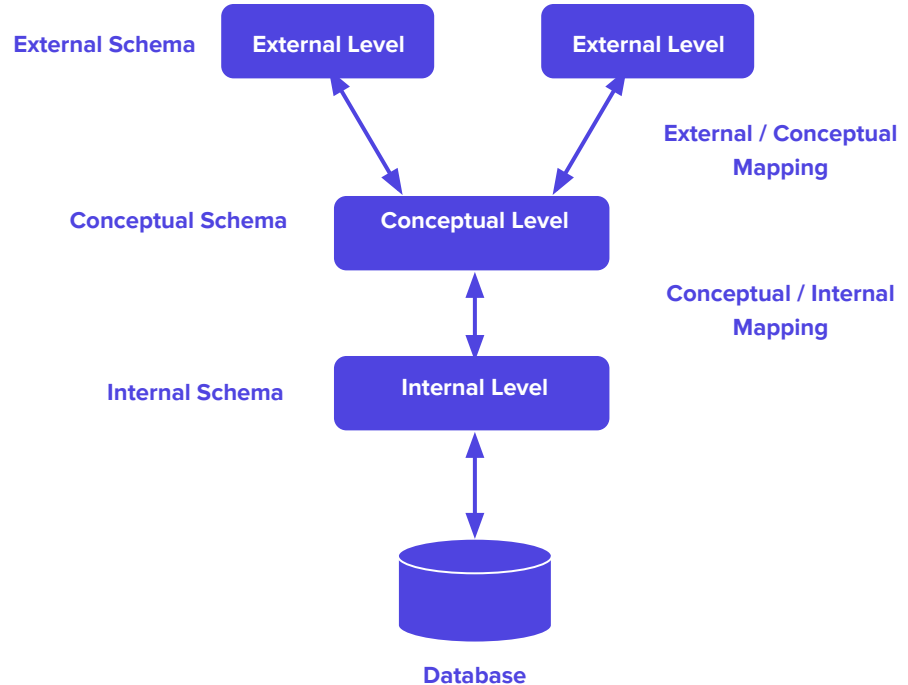
Object-Oriented Database

Object-oriented databases (OODB) is a database that represents information in the form of objects as used in object-oriented programming. OODBMS allows object-oriented programmers to develop products, store them as objects and replicate or modify existing objects to produce new ones within OODBMS.



Schema

- Schema is basically a blueprint which shows how our database is going to be. The schema here is database schema which describes the structure of our database in a formal language.



KEY

A key refers to a database component that helps us to identify an entire row uniquely in a table.

Types of Keys-

1. Primary key- Key which uniquely identifies an entity of a table.
2. Candidate Key- Except for primary key attributes rest of the attributes are considered as Candidate key.
3. Foreign Key- These are the attributes of the tables used to point primary key.
4. Alternate Key - All the attributes left after subtracting primary key from candidate key.

Apart from this 2 more keys are there i.e Composite Key and Artificial key.



Data Warehouse Schema

The Data Warehouse Schema is a framework that rationally describes the contents of the Data Warehouse by facilitating operations on the Data Warehouse and the Data Warehouse system's maintenance activities, which often includes thorough descriptions of databases, tables, views, and indexes.

A few Popular Data Warehouse Schemas are:

- Star Schema
- Snowflake Schema



Types Of Keys in Data warehouse Schema

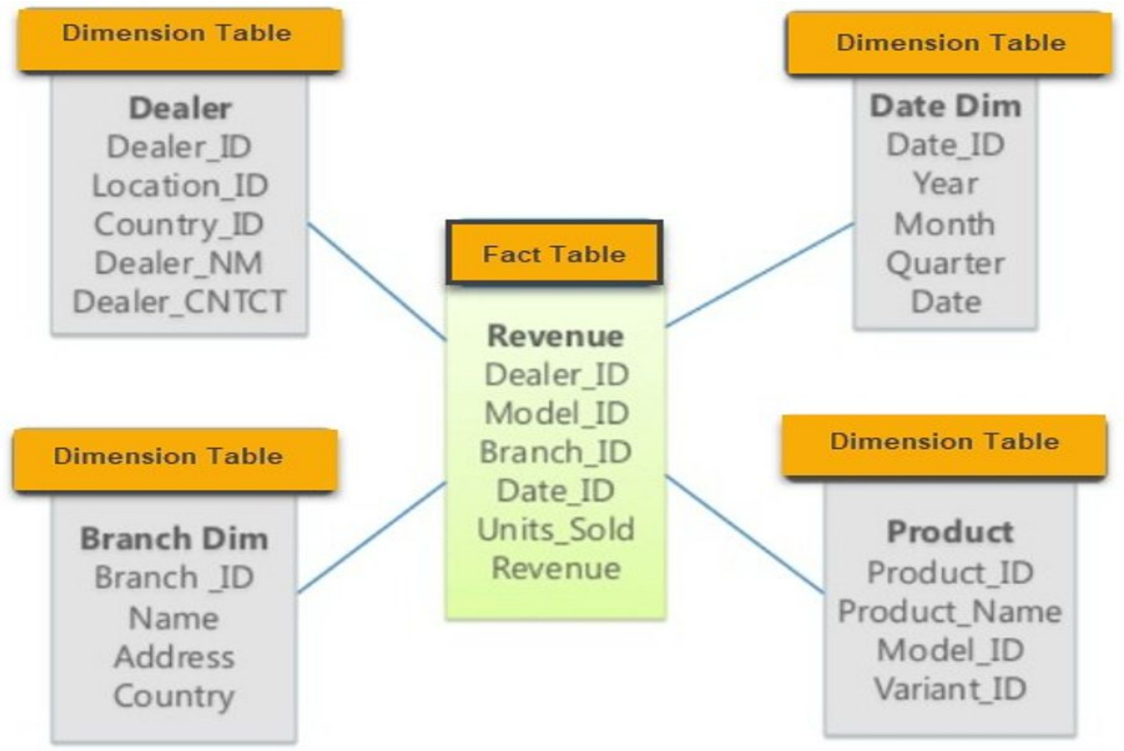
Primary key: A primary key acts as a unique identifier for every row in the table. It doesn't contain null values and no value can be removed in a primary key.

Foreign Key: It establishes a relationship between two different tables to uniquely identify a row of the same table or another table.

Star Schema

- A star schema has one fact table in the middle and multiple associated dimension tables, similar to the structure of a star. Because this form resembles a star, it is referred to as a star schema.
- The primary data in the data warehouse is represented in this fact table. It contains several fact tables and surrounds the smaller dimension lookup tables. In the fact table, each dimension's **primary key** is linked to a significant foreign gift.
- This implies that the fact table has two columns: dimension table foreign keys and numeric fact measures. The fact table is at the core of the star, and the dimension tables are at the points of the star.
- A single one-dimensional table should represent every dimension in the star schema. A fact table should be linked to the dimension table. A key and a measure should be included in the fact table.

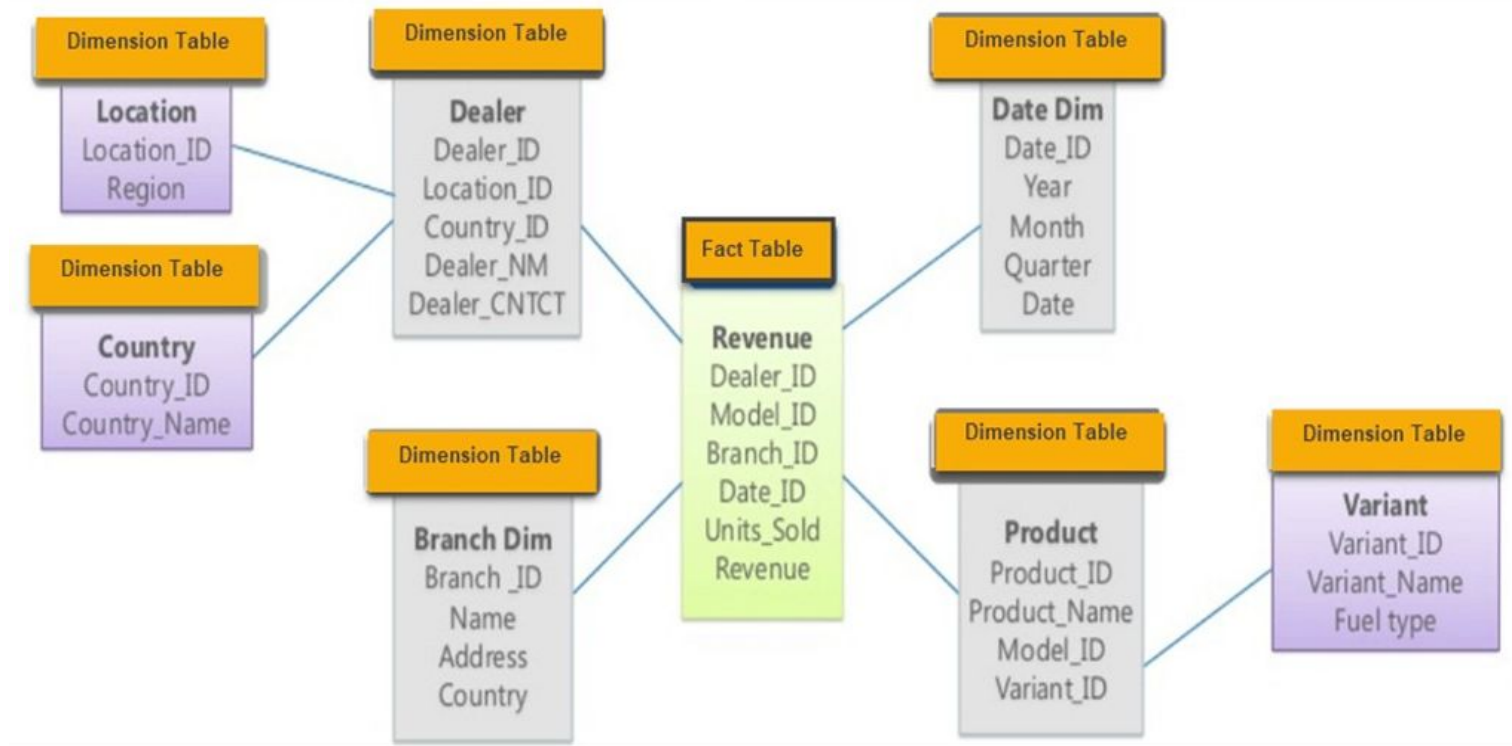
Star Schema



Snow Flake Schema

- Snowflake schema acts as an extended version of a star schema. There are additional dimensions added to the Star schema. Because of its structure, this schema is known as a snowflake schema.
- The centralised fact table will be linked to numerous dimensions in this structure. The dimensions present are in **normalized** form from the numerous related tables current. The snowflake structure is detailed and structured when compared to the star schema.
- There are multiple levels of relationships and child tables with numerous parent tables. In the snowflake schema, the affected tables are only dimension tables, not fact tables.
- The difference between a star and a snowflake schema is that a snowflake schema keeps its dimensions to reduce data redundancy. The tables are simple to organise and keep clean. They also help you conserve space in your storage.

Snow Flake Schema



What is SQL?

SQL (Structured Query Language) is a programming language used to manage relational databases and execute various operations on the data contained inside them.

Database administrators and developers who are building data integration scripts and data analysts who want to set up and perform analytical queries frequently employ SQL.

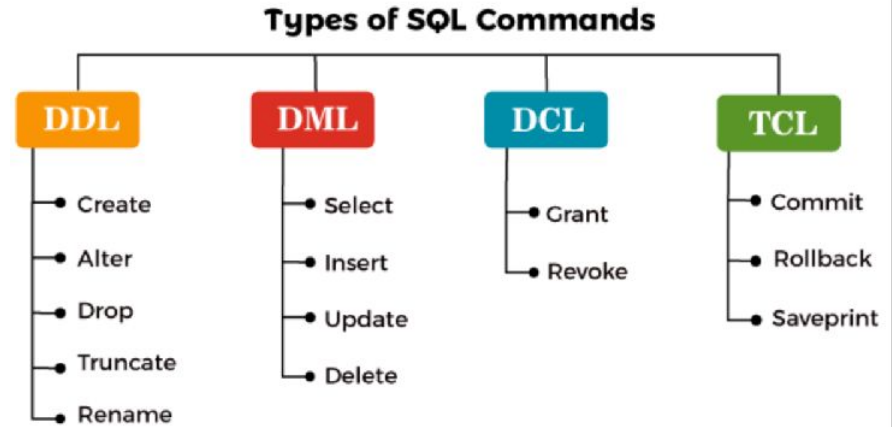
SQL is used for transaction processing and analytics applications, such as editing database tables and index structures, adding, updating, and removing rows of data, and accessing subsets of information from inside a database.



Types of SQL Commands

There are primarily four kinds of SQL commands:

- DDL(Data Definition Language)
- DML(Data Manipulation Language)
- DCL(Data Control Language)
- TCL(Transaction Control Language)



Data Definition Language(DDL)

DDL stands for **data definition language**. DDL Commands deal with the schema, i.e., the table in which our data is stored.

All the structural changes such as creation, deletion, and alteration on the table can be carried with the DDL commands in SQL.

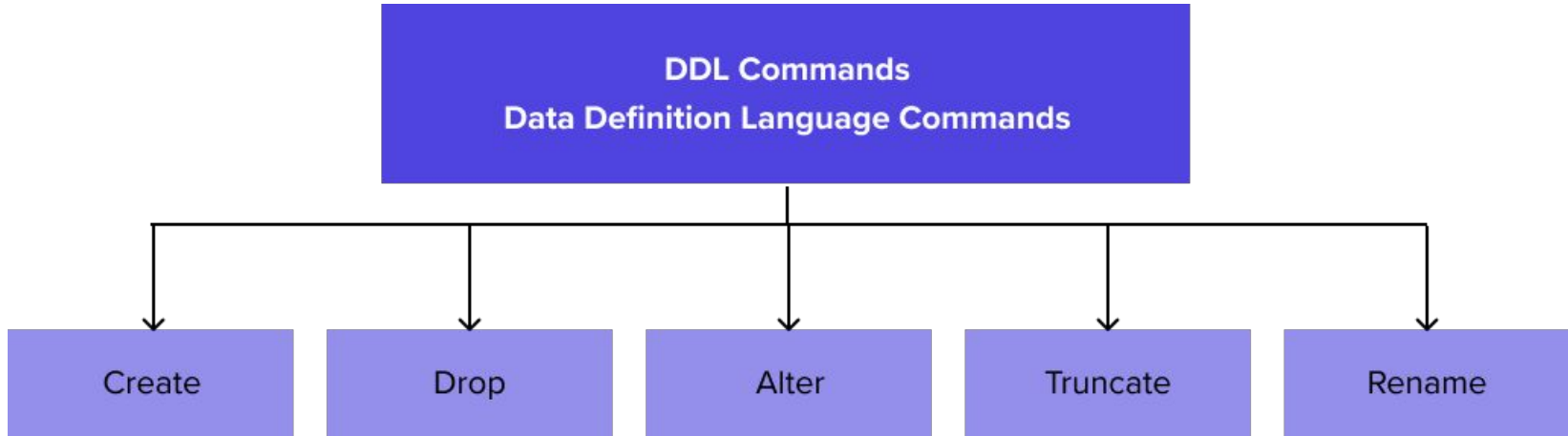
Usually DBA -Data base Administrator only has access to these commands

Commands covered under DDL are:

- CREATE
- ALTER
- DROP
- TRUNCATE
- RENAME



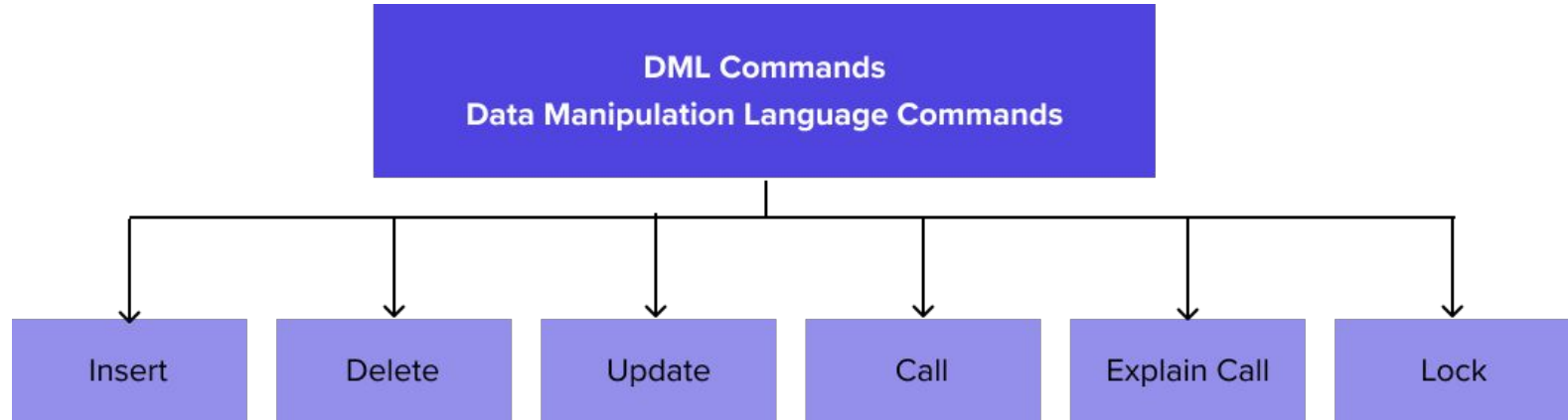
Data Definition Language(DDL)



Data Manipulation Language(DML)

- Data Manipulation Language (DML) is a programming language for manipulating data. We can alter the data in tables using DML commands in SQL.
- DML commands in SQL can be used to change data or get data from SQL tables at any time.
- DML instructions in SQL update data by **inserting new records, deleting existing records, or updating existing records** in SQL tables. We can also extract all of the data from SQL tables to meet our needs.
- Commands covered under DML are:
 - INSERT
 - SELECT
 - UPDATE
 - DELETE
- Except select, the other commands are usually controlled by DBA -Data base Administrator only has access to these commands

Data Manipulation Language(DML)

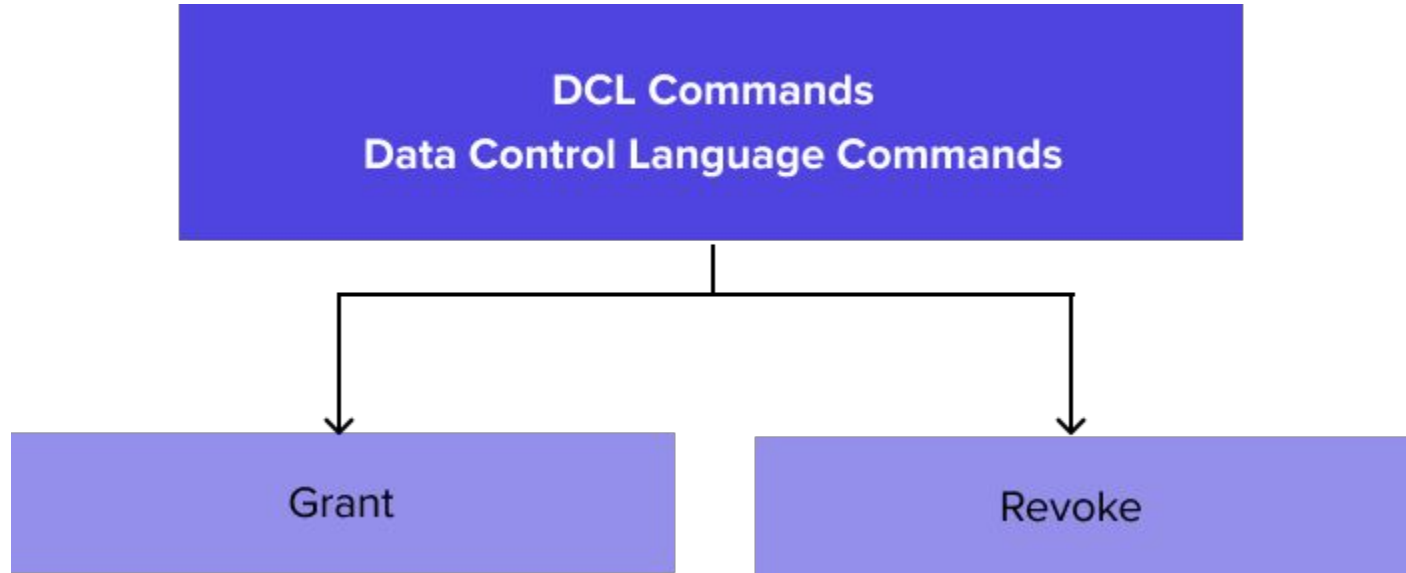


Data Control Language(DCL)

DCL stands for **Data Control Language**.

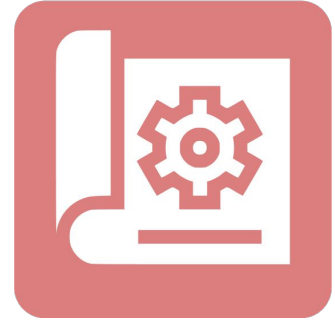
- We will utilize SQL DCL commands to control data stored in SQL tables. Authorized users can only access the data in the tables.
- Every user will have some pre-defined privileges, and only that user can view the data. We can benefit the user on the SQL database and tables by using DCL instructions or canceling the user's privileges.
- Commands covered under DCL are:
 - Grant
 - Revoke

Data Control Language(DCL)



Transaction Control Language(TCL)

- TCL stands for **Transaction Control Language**. TCL commands are generally used in transactions.
- Using TCL commands in SQL, we can save our transactions to the database and roll them back to a precise point in the transaction. We can also use the **SAVEPOINT** command to save a specific part of our transaction.
- Commands covered under TCL are:
 - Commit
 - Rollback



Transaction Control Language(TCL)



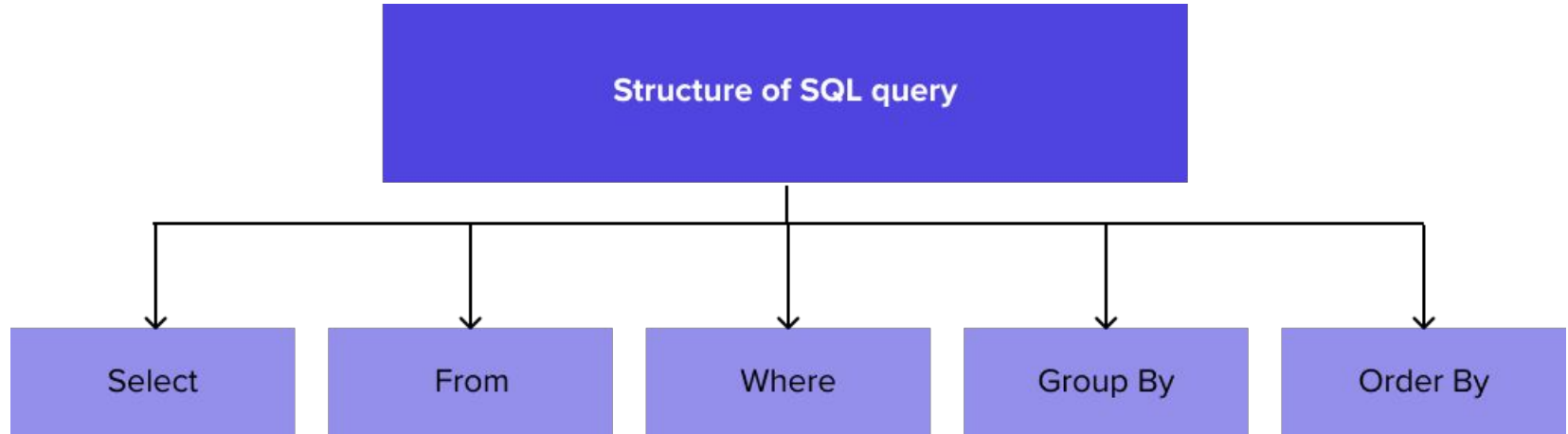
Scope of the course

SQL is a vast language and is widely used in different field of engineering for various purposes like data engineering, database administration, business analytics, data science, etc.

This course is specially designed keeping in mind the use of SQL by Business Analysts.

The scope of this course mainly focuses on Data Manipulation Language(DML) and some concepts of Data Definition Language(DDL) which are important from the point of view of Business Analysts.

Structure of SQL Query



Selecting a Single Column

Table *Store_Information*

Store_Name	Sales	Txn_Date
Los Angeles	1500	Jan-05-1999
San Diego	250	Jan-07-1999
Los Angeles	300	Jan-08-1999
Boston	700	Jan-08-1999

To select a single column, we specify the column name between **SELECT** and **FROM** as follows:

```
SELECT Store_Name FROM Store_Information;
```

Result:

```
Store_Name  
Los Angeles  
San Diego  
Los Angeles  
Boston
```

Selecting Multiple Column

We can use the **SELECT** statement to retrieve more than one column. To select Store_Name and Sales columns from **Store_Information**, we use the following SQL:

```
SELECT Store_Name, Sales FROM Store_Information;
```

Result:

<u>Store_Name</u>	<u>Sales</u>
Los Angeles	1500
San Diego	250
Los Angeles	300
Boston	700

Selecting all Column

There are two ways to select all columns from a table. The first is to list the column name of each column. The second, and the easier way is to use the symbol *. For example, to select all columns from Store_Information, we issue the following SQL:

```
SELECT * FROM Store_Information;
```

Result:

<u>Store_Name</u>	<u>Sales</u>	<u>Txn_Date</u>
Los Angeles	1500	Jan-05-1999
San Diego	250	Jan-07-1999
Los Angeles	300	Jan-08-1999
Boston	700	Jan-08-1999

Steps to log in to mode.com

1. Sign up in mode.com
1. Click on the "+" sign
1. You would get the SQL editor

Practice Question

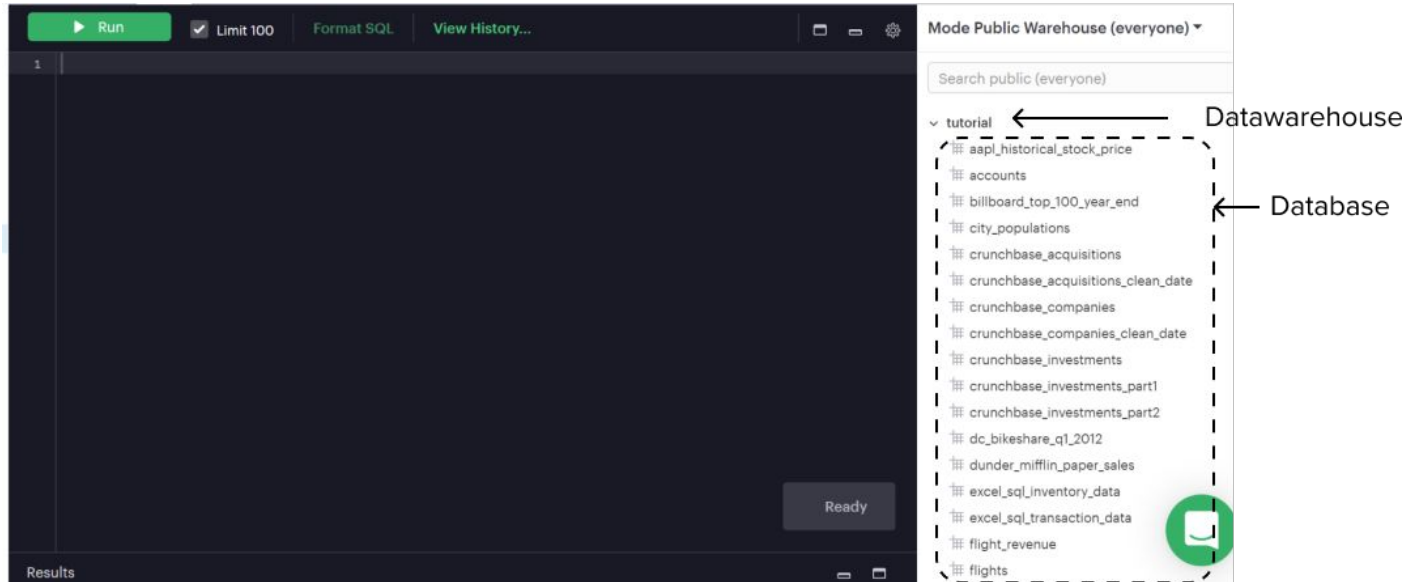
1. Understanding data Warehouse and database.

Open the mode.com and try to identify tutorial data Warehouse

Practice Question

Solution:

Understanding data Warehouse and database



Practice Question

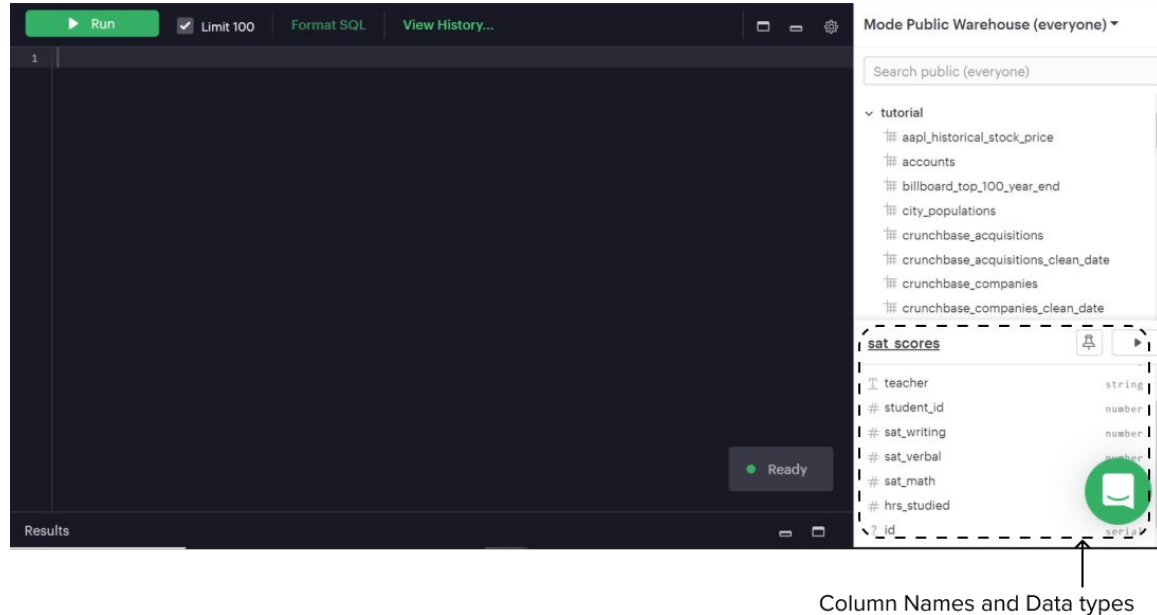
2. Understanding the columns and data types in database.

Instructions: find the sat_scores table and click to see all the columns

Practice Question

Solution:

Understanding the columns and data types in database.



Practice Question

3. Extracting the data from sat_scores table

Instructions: Go to the coding console and write a code for extracting all the columns.

Practice Question

Solution:

Extracting the data from sat_scores table

```
1 SELECT * from tutorial.sat_scores
2
```

✓ Succeeded in 462ms

✓ 100 rows | 6KB returned in 462ms

	school	teacher	student_id	sat_writing	sat_verbal	sat_math	hrs_stu
1	Washington ...	Frederickson	1	583	307	528	
2	Washington ...	Frederickson	2	401	791	248	
3	Washington ...	Frederickson	3	523	445	756	
4	Washington ...	Frederickson	4	306	269	327	
5	Washington ...	Frederickson	5	300	539	743	
6	Washington ...	Frederickson	6	213	500	771	

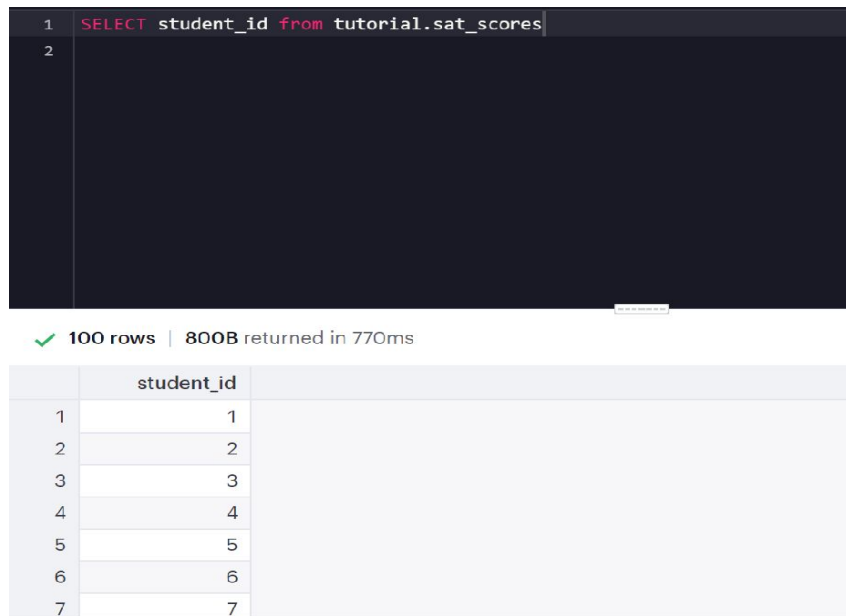
Practice Question

4. Extract the `student_id` column from the `sat_scores` table

Practice Question

Solution:

Selecting student_id column from the sat_scores table



The screenshot shows a SQL query execution interface. The query entered is `SELECT student_id from tutorial.sat_scores`. Below the query editor, a status bar indicates that 100 rows were returned in 770ms. The results are displayed in a table with two columns: an index and the `student_id` values.

	student_id
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Practice Question

5. Selecting sat_writing, sat_verbal, sat_math column from the sat_scores table

Practice Question

Solution:

Selecting sat_writing, sat_verbal, sat_math column from the sat_scores table

```
1 SELECT
2   sat_writing,
3   sat_verbal,
4   sat_math
5 FROM
6 tutorial.sat_scores
7
```

✓ 100 rows | 2KB returned in 706ms

	sat_writing	sat_verbal	sat_math
1	583	307	528
2	401	791	248
3	523	445	756
4	306	269	327
5	300	539	743
6	213	500	771
7	548	683	740
8	314	503	341

Conclusion

In the next class we will study:



Creating databases using DDL and DML commands