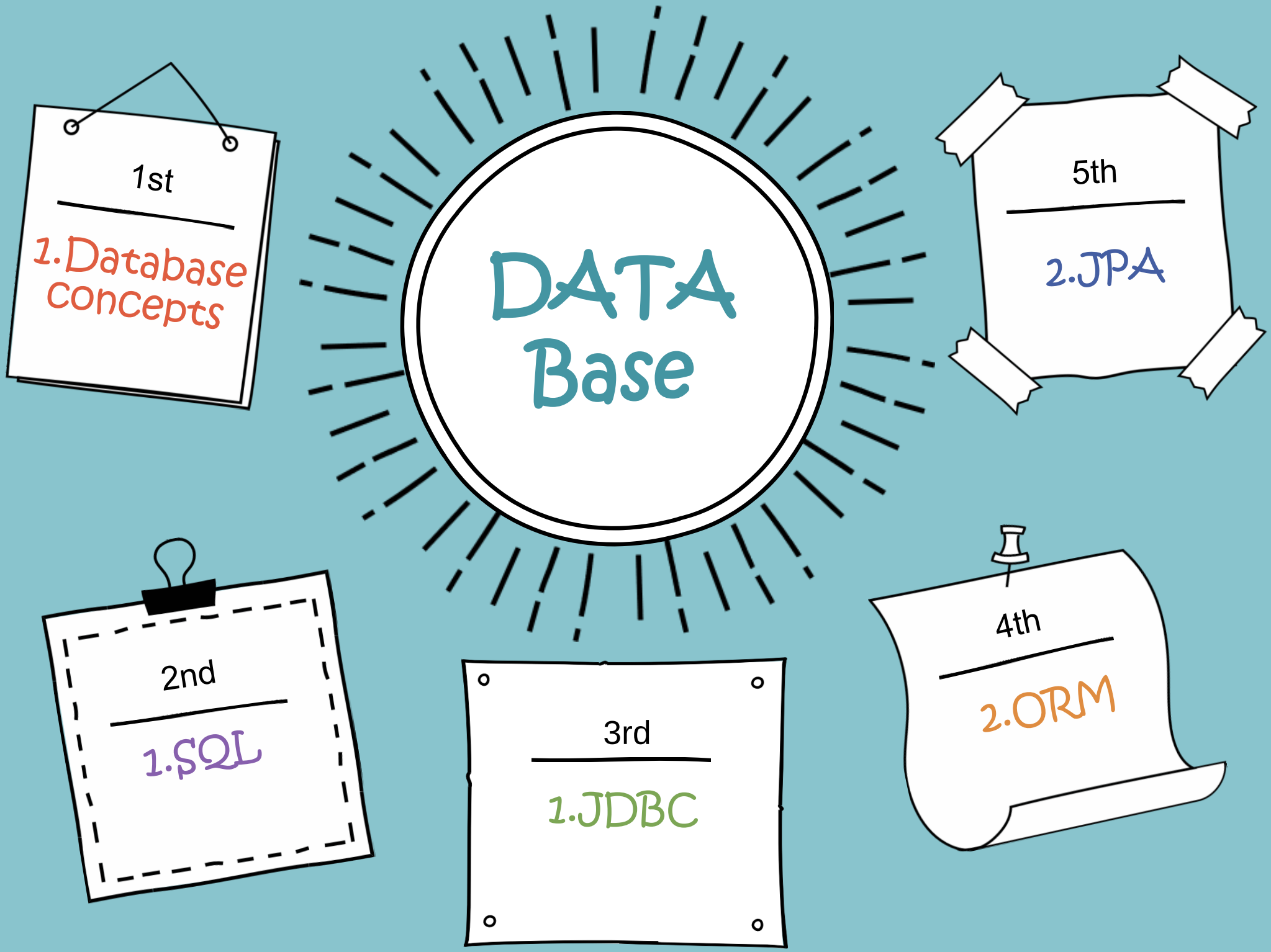
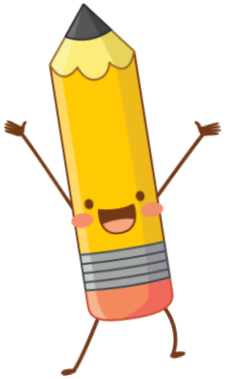

DATA
Base
1st
1.Database
concepts
2nd
1.SQL
3rd
1.JDBC
4th
2.ORM
5th
2.JPA



Concepts...

What is the database ?

Collection of related data saved together

Data(inputs) vs information(conclusions)

Has some aspect from the real world(miniworld) – universe of discourse(UOD)

Logically coherent - not a random assortment

Concepts ……..

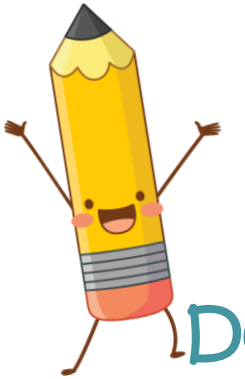
DDMS … ?

Collection of programs (software) that enable users to create DB and Maintain.

DBMS is general purpose software sys that facilitate process of defining, constructing, manipulating and sharing among various apps and users.

Concepts ……..

Types…



Depend on your purpose ? Types….?

Relational Databases (RDBMS):
mysql, oracle & sql server ….

NoSQL Databases: MongoDB,
Redis, …..

Geospatial Data Warehouses: Esri
Arc GIS, …..

Concepts ……..

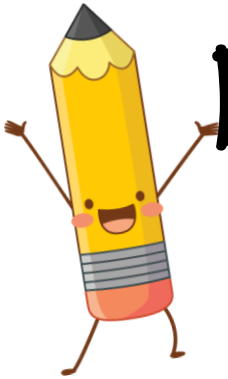
DDMS … ?

1.3 Advantages of DBMS.

Due to its centralized nature, the database system can overcome the disadvantages of the file system-based system

1. **Data independency:** Application program should not be exposed to details of data representation and storage. DBMS provides the abstract view that hides these details.
2. **Efficient data access:** DBMS utilizes a variety of sophisticated techniques to store and retrieve data efficiently.
3. **Data integrity and security:** Data is accessed through DBMS, it can enforce integrity constraints. E.g.: Inserting salary information for an employee.
4. **Data Administration:** When users share data, centralizing the data is an important task, Experience professionals can minimize data redundancy and perform fine tuning which reduces retrieval time.
5. **Concurrent access and Crash recovery:** DBMS schedules concurrent access to the data. DBMS protects user from the effects of system failure.
6. **Reduced application development time:** DBMS supports important functions that are common to many applications.

Concepts ……..



Database Charact. …?

It is a set of properties that ensure reliability and consistency in database transactions.

A Atomicity

C Consistency

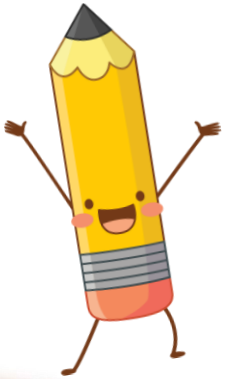
I Isolation

D Durability

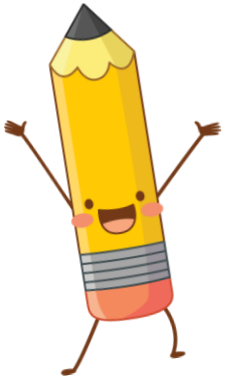
Commit vs Rollback

Concepts ……..

DDMS … ?



Concepts ……..

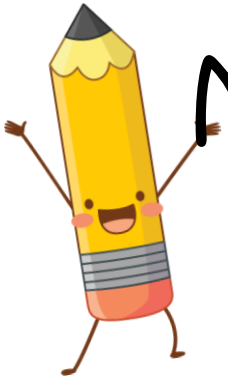


DDMS … ?

???

Who check is column exist to select

Concepts…



META-DATA(Catalog)

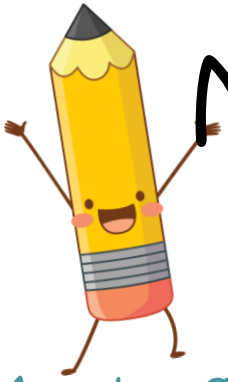
A data dictionary, also known as a data catalog,

is a centralized repository that contains metadata about the data stored in a database.

Metadata is data about data, providing information about the structure, constraints, relationships, and usage of the actual data stored in the database.

The data dictionary is an essential component of a (DBMS) as it helps in the management, organization, and understanding of data.

Concepts…



META-DATA(Catalog)

Key Functions of a Data Dictionary:

1. Metadata Storage: It stores detailed information about database objects such as tables, columns, indexes, views, triggers, and procedures.

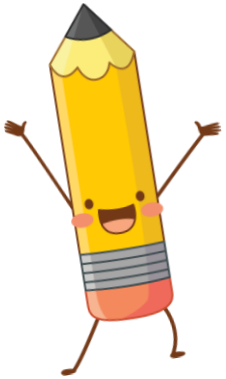
2. Data Definitions: Provides definitions and descriptions of data elements, including data types, lengths, default values, and constraints.

3. Data Relationships: Documents relationships between different data elements, such as primary and foreign key relationships.

4. Data Usage: Records information on how data is used, including user access permissions, data ownership, and usage statistics.

5. Data Standards: Ensures consistency in data naming conventions, formats, and usage across the database.

Concepts ……..



Example…

Actions will need …?

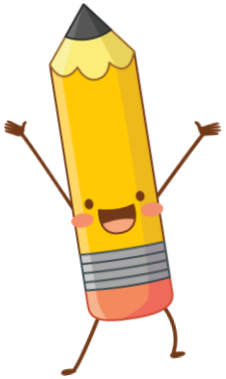
INSERT

SELECT

DELETE

UPDATE

Concepts ……..



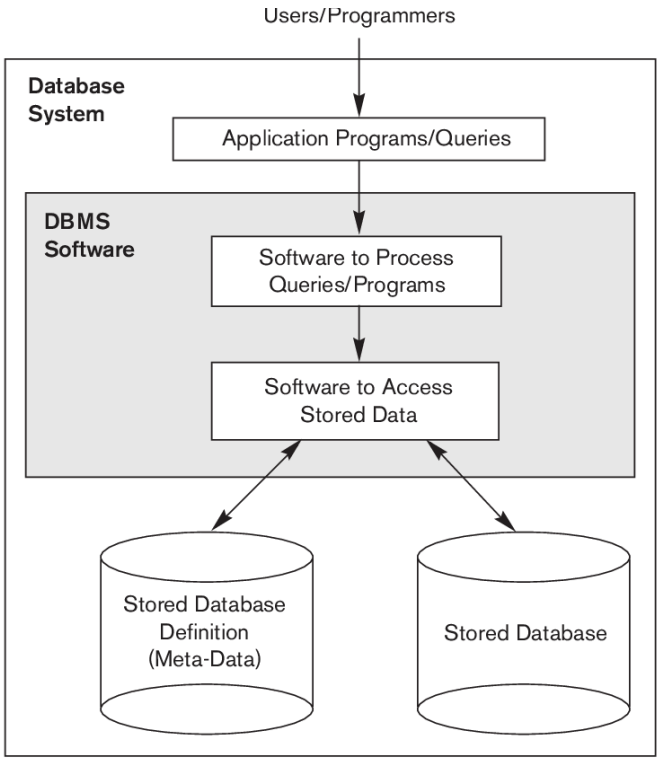
Processing Steps …?

1. Query Parsing and Analysis:

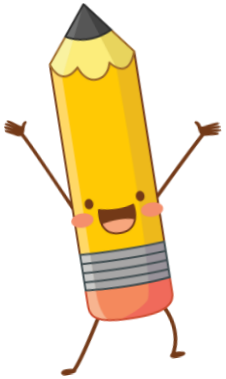
The query is parsed to ensure it follows the syntax rules of the query language and analyzed to ensure it conforms to the database schema

2. Query Optimization:

The DBMS optimizer analyzes the query to determine the most efficient way to execute it, considering factors like indexes



Concepts ……..



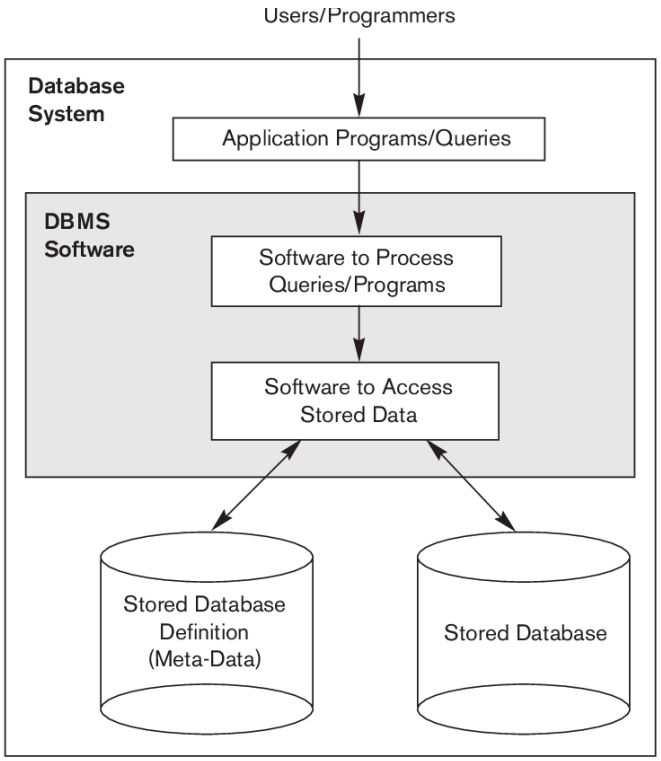
Processing Steps …?

3. Query Execution:

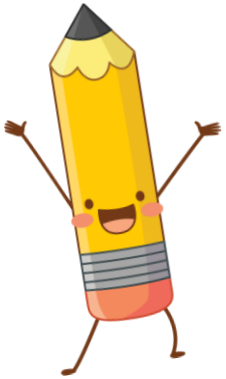
The optimized query is executed against the database

4. Result Presentation:

Finally, the query results are formatted and presented to the user or application that initiated the query



Concepts ……..



Actors on Scene

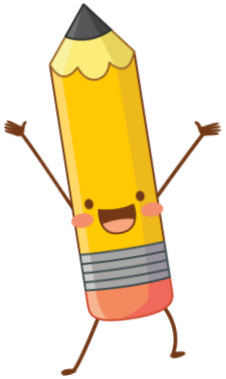
Database Administrator: install, conf, sec, perf, back, recovery

Database Designers: Designs the logical and physical structure of the database, defines data models, schemas, and relationships based on user requirements.

System Analysts And Software Engineers

End Users

Concepts ……..



DBMS Languages

DDL, DML

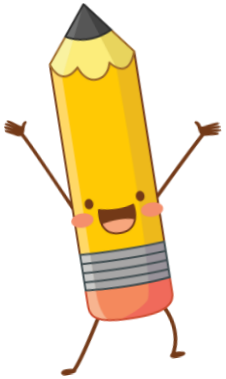
DDL:

Defines schemas

DML:

allow retrieve, insertion, deletion and updating

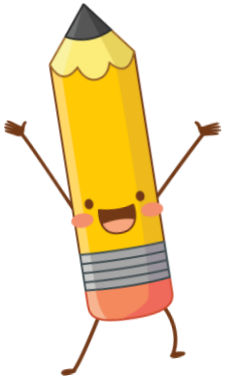
Concepts ……..



Database Arch.

Tiers

Concepts ……..



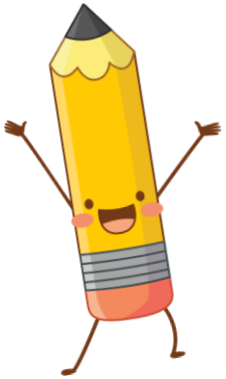
Database Arch.

1-Tier Architecture

Single Tier Architecture

in DBMS is the simplest architecture of Database in which the client, server, and Database all reside on the same machine. A simple one tier architecture example would be anytime you install a Database in your system and access it to practice SQL queries. But such architecture is rarely used in production.

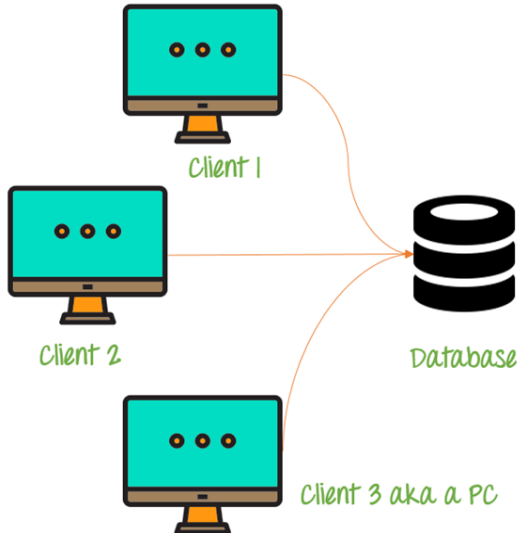
Concepts ……..



Database Arch.

2-Tier Architecture

The application is split into two parts: the client, which handles the user interface and application logic, and the server, which handles database management.

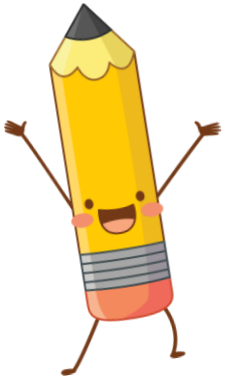


Client side - database

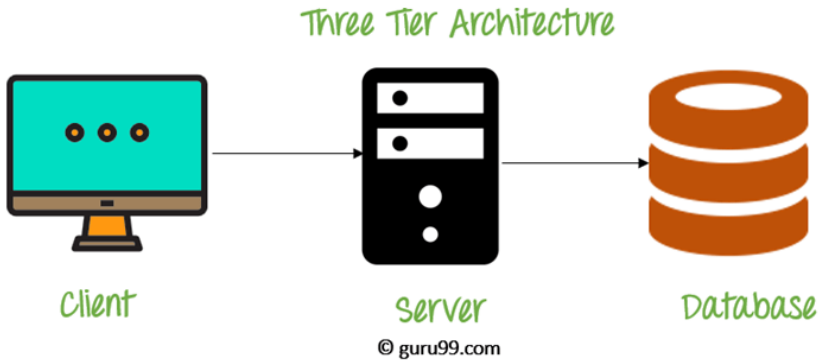
JDBC, ODBC sw

Concepts ……..

Database Arch.

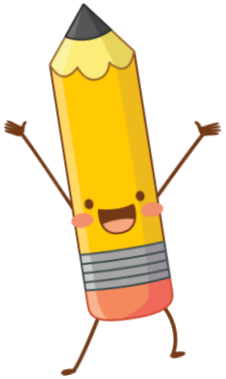


3-Tier Architecture



Application server / web server

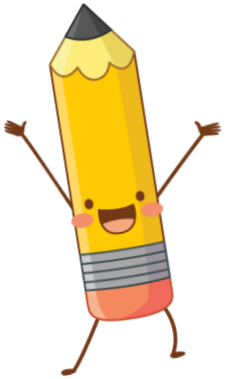
Concepts ……..



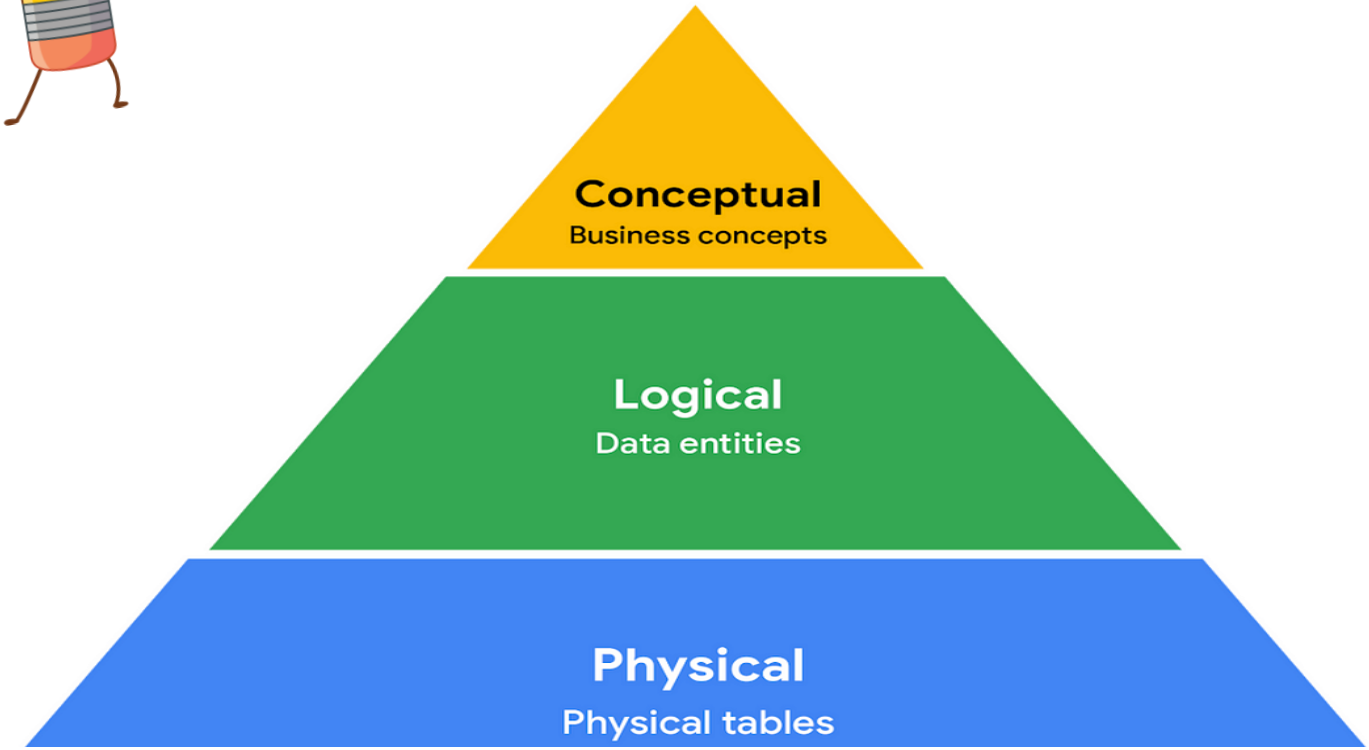
Database Arch.

N-Tier

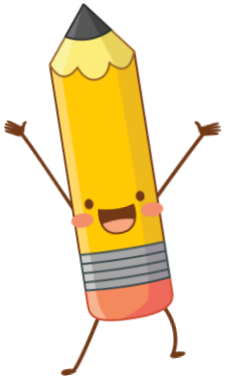
Concepts ……..



Data Model Categ.



Concepts ……..

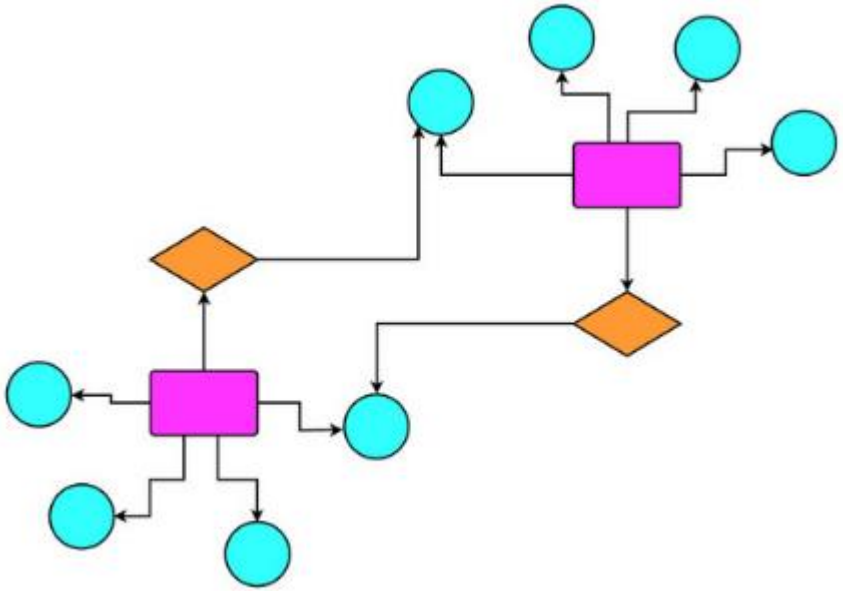


Conceptual Data M.

Entity

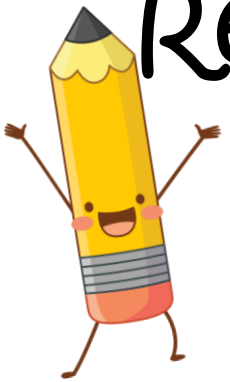
Attributes

Relationship

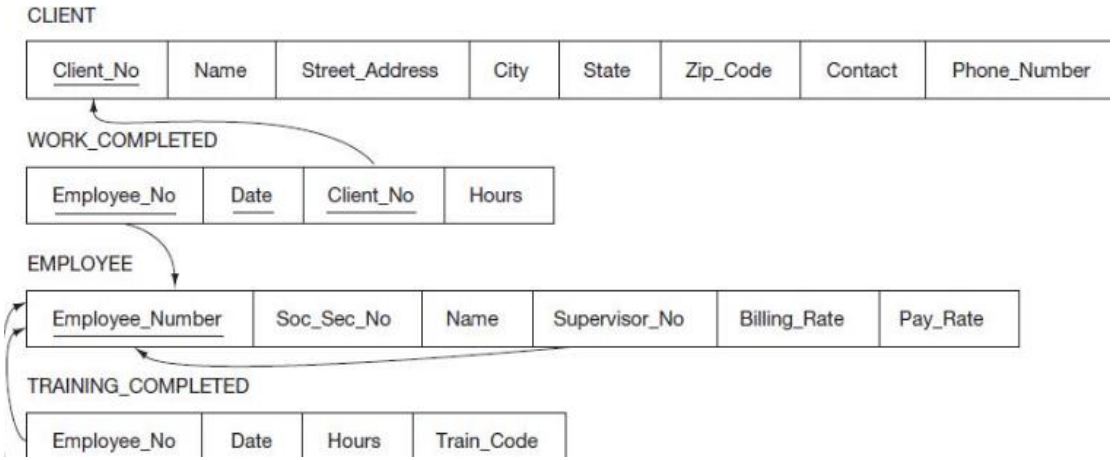


Concepts ……..

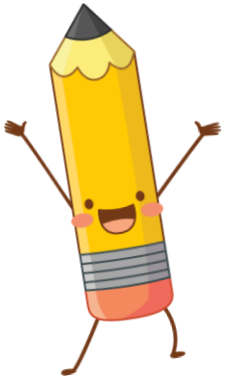
Representation Data M. (Relational)



Relation = table



Concepts ……..



Note …

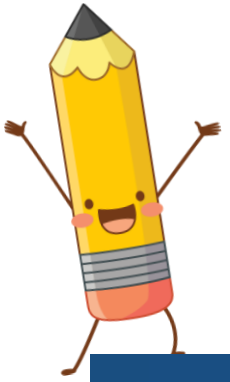
Database schema: description of the database(in database design step, columns, constraints, …..)

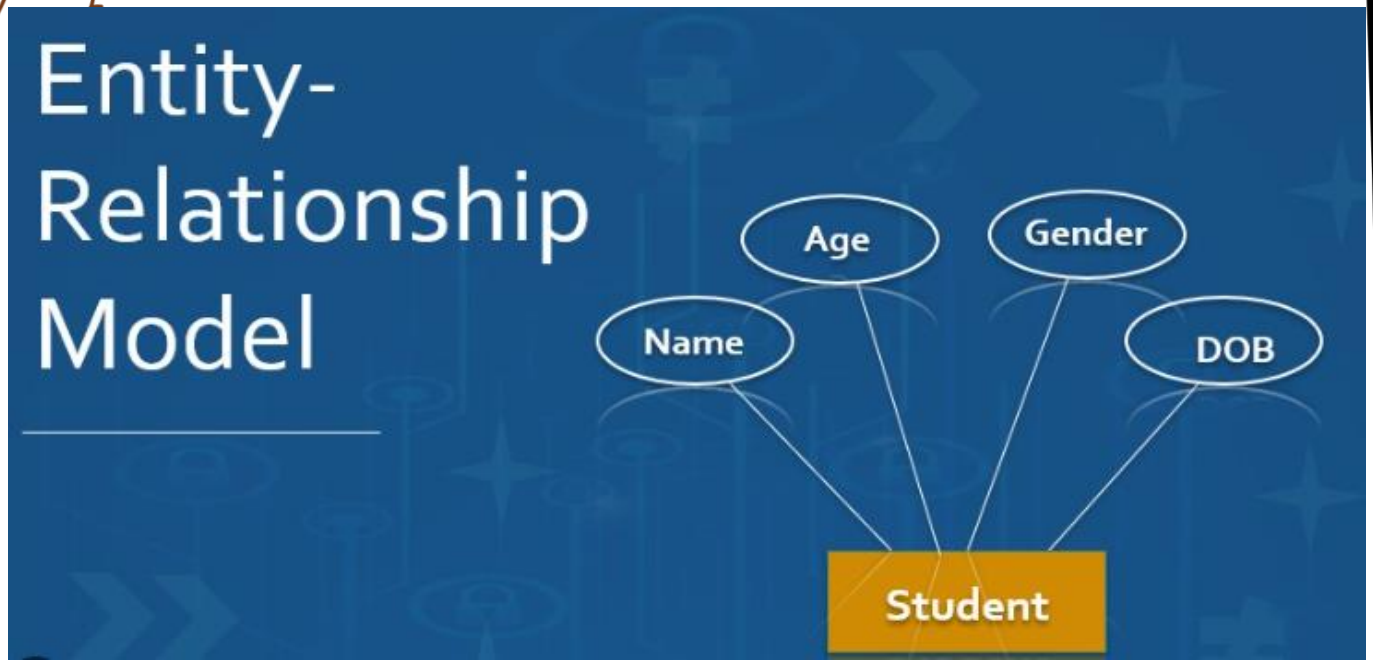
Data in database changed every time, so data in the particular moment is called database state or snapshot – current set of occurrences or instances

Instances === record

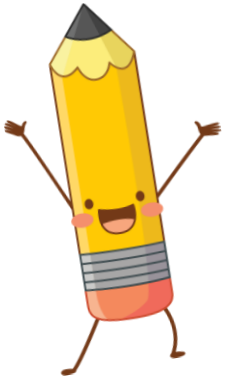
Concepts ……..

ER Model

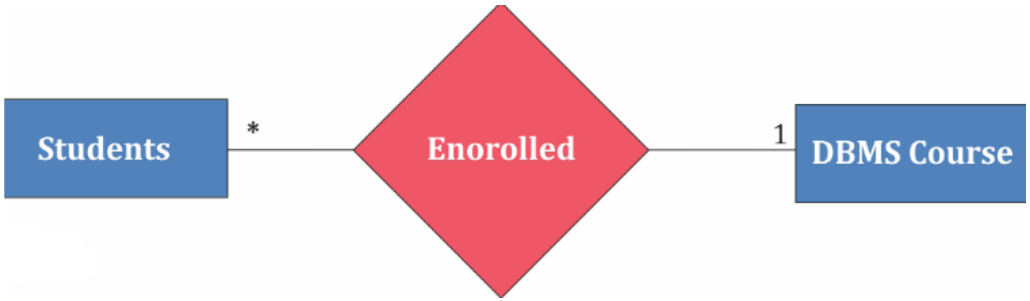




Concepts ……..



ER Model



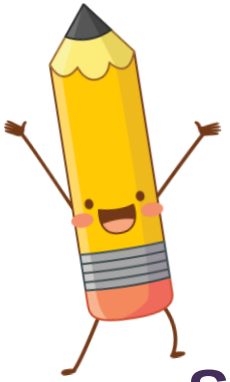
Entity

Attributes

Relationships

Concepts ……..

ER Model

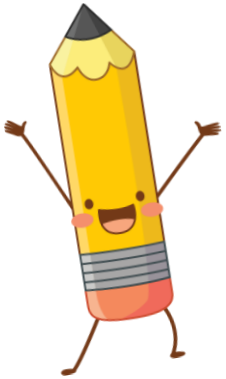


Attributes Types:

- **Simple attribute**
- **Composite attribute**
- **Single-valued attribute**
- **Multi-valued attribute**
- **Stored = normal**
- **Derived attribute => from saved**
- **Complex = multi-value + composite**

Concepts ……..

ER Model



Relationship degrees & Cardinality

degrees

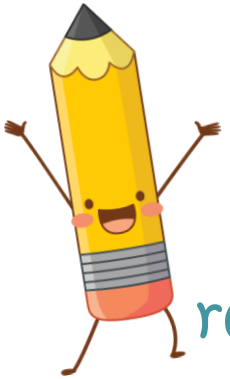
- Binary ()
- Ternary
- Recursive

Cardinality | constraints

- 1:1
- 1:M M:1
- M:N

Concepts ……..

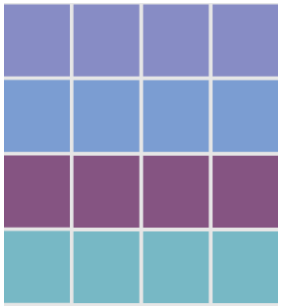
Normalization



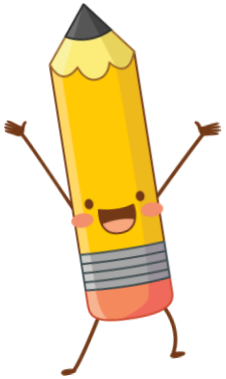
reduce data redundancy

And solve update anomaly

1. First Normal Form (1NF)
2. Second Normal Form (2NF)
3. Third Normal Form (3NF)



Concepts ……..



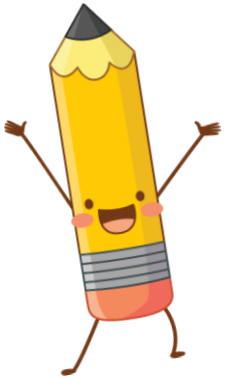
Normalization

1. First Normal Form (1NF)

Remove Multi Values

Remove Redundant

Concepts ……..

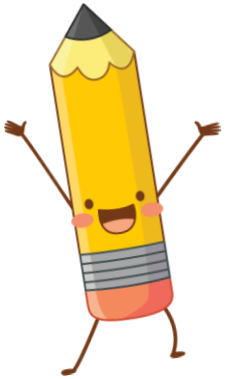


Normalization

2. Second Normal Form (2NF)

Remove Partial Dependency

Concepts ……..

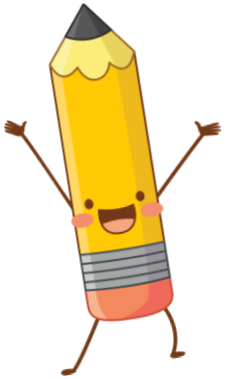


Normalization

3. Third Normal Form (3NF)

Remove Transitive Dependency

SQL Part



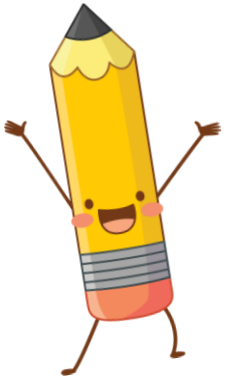
SQL

Structure Query Language

SQL is a standard language for accessing and manipulating databases.

International Organization for Standardization (ISO) in 1987

SQL Part



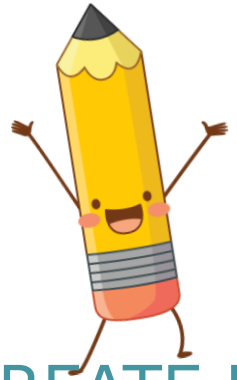
SQL

Structure Query Language

TABLE – COLUMN – ROW - FIELD

PK – FK – INDEX

SQL Part

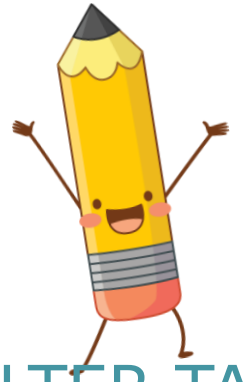


SQL - DDL

```
CREATE DATABASE databasename;

DROP DATABASE databasename;

CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
   ....
);

DROP TABLE table_name;
```

SQL Part

SQL - DDL

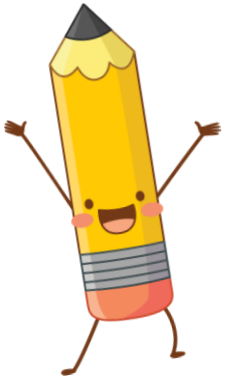


ALTER TABLE table_name
ADD column_name datatype;

ALTER TABLE table_name
DROP COLUMN column_name;

ALTER TABLE table_name
Alter/MODIFY COLUMN column_name
datatype;

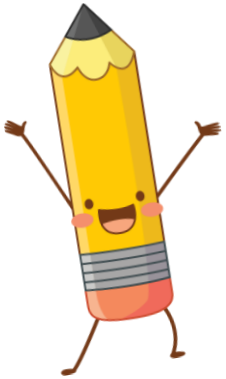
SQL Part



SQL - Constraints

- NOT NULL - Ensures that a column cannot have a NULL value
- UNIQUE - Ensures that all values in a column are different
- PRIMARY KEY - A combination of a NOT NULL and UNIQUE in table.
- FOREIGN KEY - Prevents actions that would destroy links between tables
- DEFAULT - Sets a default value for a column if no value is specified
- CREATE INDEX - To create and retrieve data from the database very quickly

SQL Part

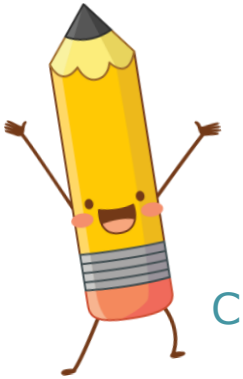


SQL - Constraints

```
CREATE TABLE Persons (
    ID int NOT NULL,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255) NOT NULL,
    Age int
);
ALTER TABLE Persons
MODIFY COLUMN Age int NOT NULL;
```

oracle 10G and later remove COLUMN

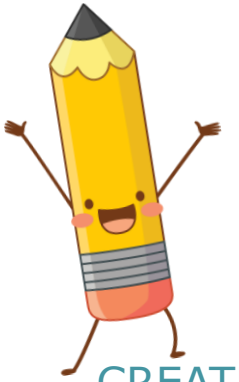
SQL Part



SQL - Constraints

```
CREATE TABLE Persons (
    ID int NOT NULL,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int,
    PRIMARY KEY (ID)
);

ALTER TABLE Persons

ADD PRIMARY KEY (ID);
```

SQL Part



SQL - Constraints

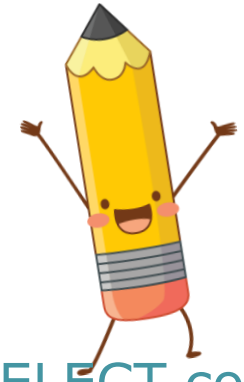
```
CREATE TABLE Orders (
    OrderID int NOT NULL,
    OrderNumber int NOT NULL,
    PersonID int,
    PRIMARY KEY (OrderID),
    FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)
);
```

```
ALTER TABLE Orders
ADD CONSTRAINT FK_PersonOrder
FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);
```

```
ALTER TABLE Orders
DROP CONSTRAINT FK_PersonOrder;
```

Mysql FOREIGN kEY

SQL Part



SQL-DML

```
SELECT column1, column2, ...
FROM table_name;
```

```
SELECT DISTINCT Country FROM Customers;
```

```
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

SQL Part

SQL-DML

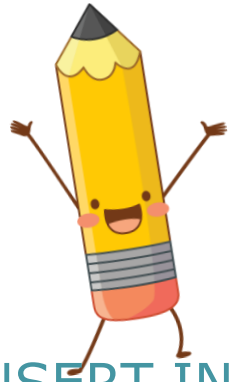
```
SELECT DISTINSELECT column1, column2, ...
FROM table_name
ORDER BY column1, column2, ... ASC|DESC;
```

```
SELECT column1, column2, ...
FROM table_name
WHERE condition1 AND condition2 AND condition3 ...;
```

```
SELECT column1, column2, ...
FROM table_name
WHERE NOT condition;
```

SQL Part

SQL-DML

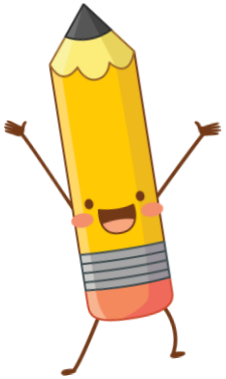


```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

```
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;
```

```
DELETE FROM table_name where … ;
```

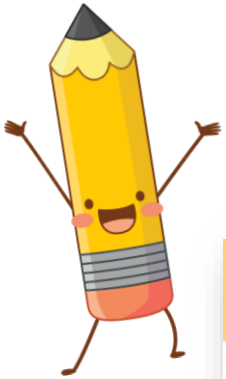
SQL Part



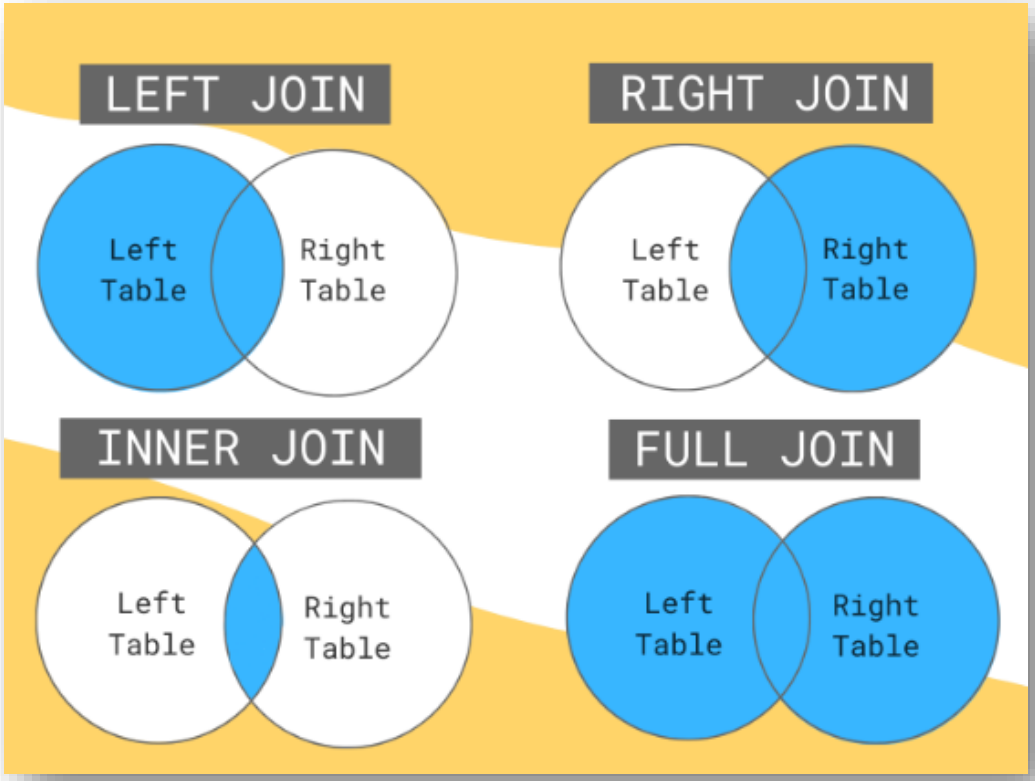
SQL-DML

JOIN clause is used to combine rows from two or more tables, based on a related column between them

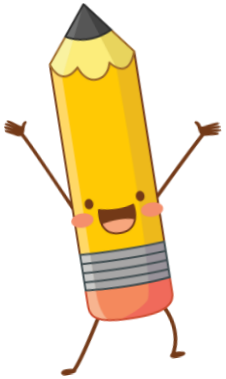
SQL Part



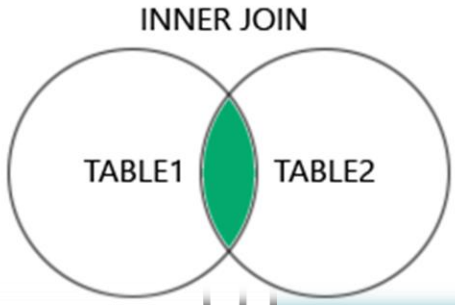
JOINS



SQL Part

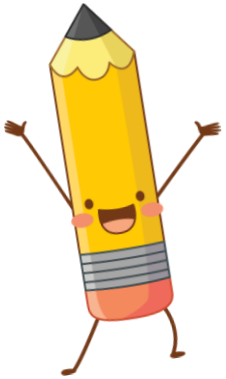


SQL-DML

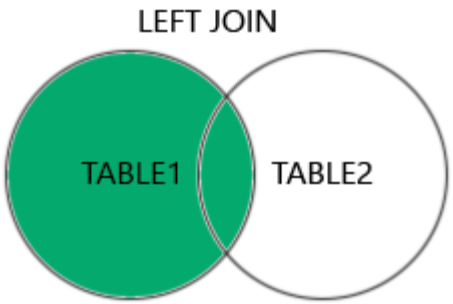



```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
```

SQL Part

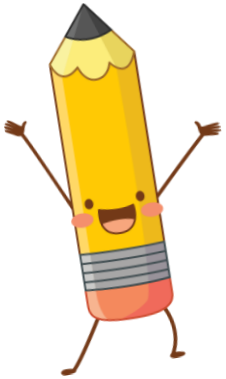


SQL-DML

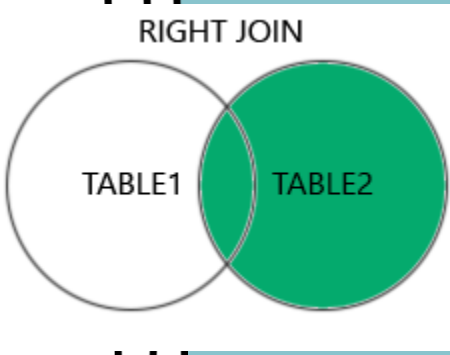



```
SELECT column_name(s)
FROM table1
LEFT JOIN table2
ON table1.column_name = table2.column_name;
```

SQL Part

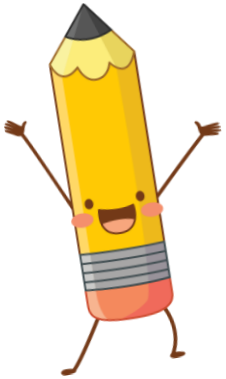


SQL-DML



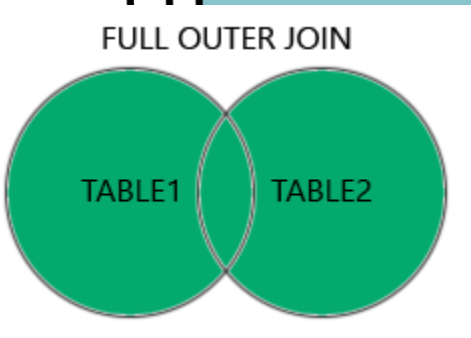

```
SELECT column_name(s)
FROM table1
RIGHT JOIN table2
ON table1.column_name = table2.column_name;
```

SQL Part

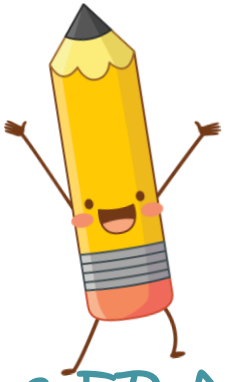


SQL-DML

```
SELECT column_name(s)
FROM table1
FULL OUTER JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;
```



SQL Part



Assignment

- 2 ER Model for different small businesses.
- Create full database schema for 1 of them.
- Create 8 DML queries on them, use at least 3 join queries.
- Mention main 2 vendors for RDBMS