Welcome to the Databases - SQL module!

Trainer: Diana Cavalcanti

Scope

- Relations
- Databases, Tables: Creating and Designing
- Data types, indexes, limitations
- SQL
- CRUD
- Complex queries with JOIN (INNER, OUTER, LEFT, RIGHT)
- having, group by, order by, limit
- (Optional)
- triggers, procedures
- Transactions
- ACID

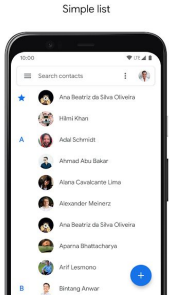
Software:
- MySQL 5.7.x+/8.x.y+
- MySQL Workbench 5.x.y+/8.x.y+

Important

Attendance list
Break time

Fundamentals

- Do you know what a database is?
  - A database is an organized collection of data
  - Would you know how to measure how much this area is present in your life?



Database system

A Database system is basically a computerized information storage system, that is, a computerized system whose main purpose is to maintain, store and make information available. ” (C.J. Date)

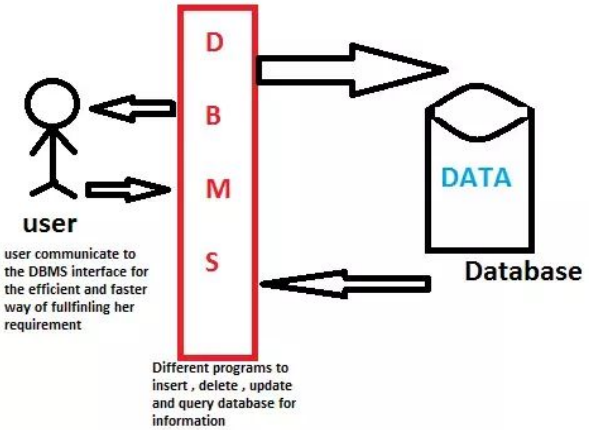
Main purpose:

- Organized storage aimed at:
  - System optimization
  - Facilitate insert, update, processing and consultation

https://en.wikibooks.org/wiki/Introduction_to_Database_Systems

A Database Management System (DBMS)

- DBMS is a system (software) that provides an interface to database for information storage and retrieval
  - capacity for large amount of data
  - an easy to use interface language (SQL-structured query language)
  - efficient retrieval mechanisms
  - multi-user support
  - security management
  - concurrency and transaction control
  - persistent storage with backup and recovery for reliability



https://en.wikibooks.org/wiki/Introduction_to_Database_Systems

A Database Management System (DBMS)

Examples of popular DBMS used these days:

- MySql
- Oracle
- SQL Server
- IBM DB2
- PostgreSQL

Relational databases

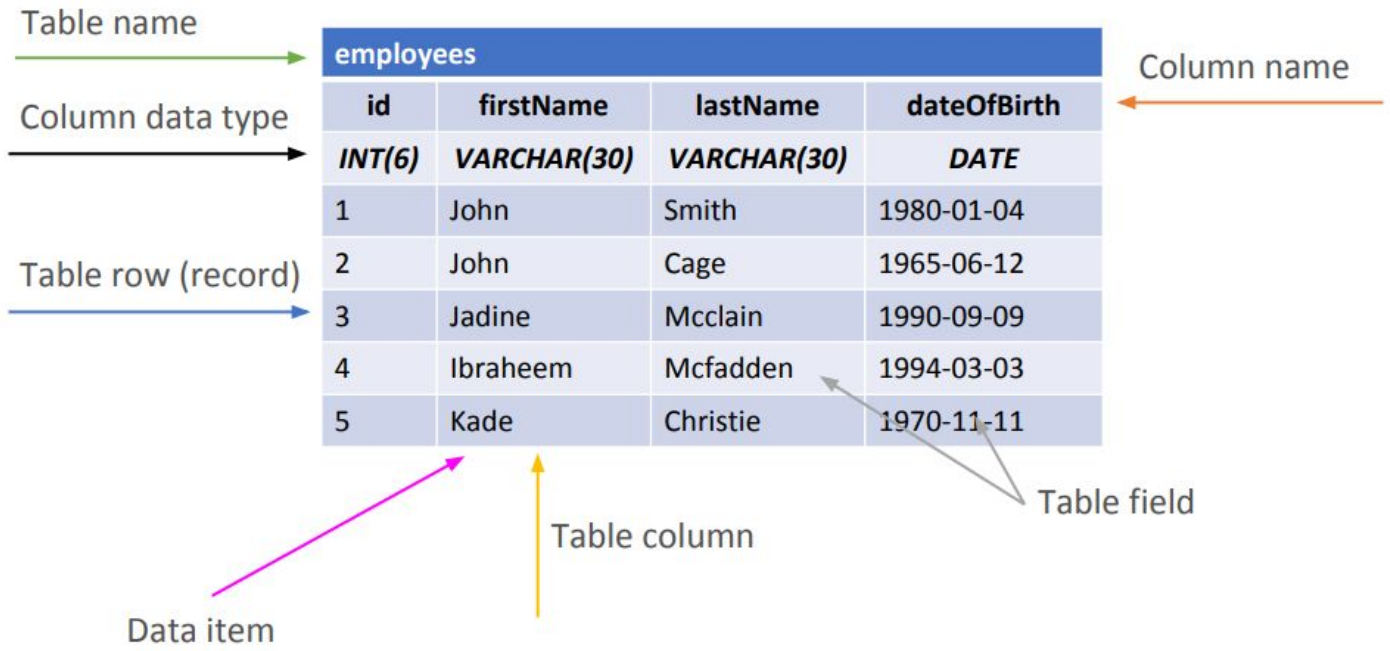
- This model organizes data into one or more tables (or "relations") of columns and rows, with a unique key identifying each row.

- A table is a collection of data held in a two dimensional structure.
- The two dimensions are rows and columns.
- A table is identified by a name.

https://www.oracle.com/database/what-is-database.html

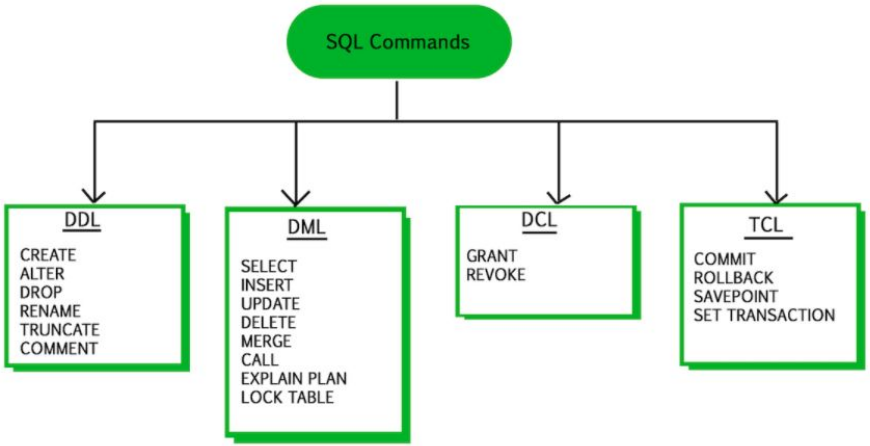
Relational databases

- Table



SQL - Structured Query Language

- **DDL** - **data definition language**. Helps users define what kind of data they are going to store and how they are going to model this data.
- **DML** - **data manipulation language**. Allows users to insert, update and delete data from the database.
- **DQL** - **data query language**. Helps users retrieve information from the database.
- **DCL** - **data control language**. Allows users to restrict and control access to the database.



SQL - Structured Query Language

- **DDL - Data Definition Language**
-
  - Create a database
    - **CREATE DATABASE** sda_course;
  - Select the database
    - use sda_course;
  - Delete a database
    - **DROP DATABASE** sda_course;

SQL - **DDL - Data Definition Language**

- Create a table

```
CREATE TABLE employees (
    id_employees INT,
    first_name VARCHAR(30),
    last_name VARCHAR(30),
    salary INT
);
```

- Column data types: The column data types define the type of information you can store in that particular column:
- **numeric:** int, tinyint, bigint, float, real, etc.,
- **date and time:** Date, Time, Datetime, etc.,
- **character and string:** char, varchar, text, etc.,
- **logical values:** TINYINT type value (0 or 1).

SQL - Structured Query Language

- **DDL - Data Definition Language**

  - describe employees;
  - Delete a table
    - **DROP TABLE** employees;

SQL - Structured Query Language

- **DDL - Data Definition Language**
-
    - Add a column

      ALTER TABLE employees
      ADD dateOfBirth **VARCHAR(10);**

    - Update a column

      ALTER TABLE employees
        MODIFY dateOfBirth **VARCHAR(50);**

SQL - Structured Query Language

- **DDL - Data Definition Language**
  - RENAME a column

    ALTER TABLE employees
    CHANGE COLUMN dateOfBirth date_of_birth **DATE**

  - DELETE a column

    ALTER TABLE employees
    DROP COLUMN date_of_birth **;**

SQL - Structured Query Language

- **DDL - Data Definition Language**

When defining a table the user can set certain properties on the columns:

- data type controls the type of values stored in the column,
- **NOT NULL** defines whether a column must be filled or not,
- **AUTOINCREMENT** states that the column value will be generated automatically (incrementation of the last inserted value) - this only works for numeric columns,
- **UNIQUE** states that there cannot be more than one row with the same value for that particular column.

SQL - Structured Query Language

-
- **NOT NULL**
  - **ALTER TABLE** employees **MODIFY** first_name **VARCHAR(30) NOT NULL;**
-
- **AUTOINCREMENT**
  - **ALTER TABLE** employees **CHANGE** id_employees id_employees **INT NOT NULL AUTO_INCREMENT PRIMARY KEY;**
  -
- **UNIQUE**
  - **ALTER TABLE employees ADD UNIQUE (last_name);**

Exercises

1. Create a new database: humanResources
2. Create a new table employees, with the following columns:
   a. employeeId - INTEGER ,
   b. firstName - VARCHAR,
   c. lastName - VARCHAR,
   d. dateOfBirth - DATE,
   e. postalAddress - VARCHAR.
3. Alter table employees and add the following columns:
   a. phoneNumber - VARCHAR,
   b. email - VARCHAR,
   c. salary - INTEGER.
4. Alter table employees and remove the postalAddress column.
5. Create a new table employeeAddresses,
   a. country_id - INTEGER
   b. country_name - VARCHAR.
6. Remove table employeeAddresses.

DML - Data Manipulation Language

- **Adding data**

INSERT INTO employees (id_employees, first_name, last_name, salary, date_of_birth) **VALUES**
(1, 'Michael', 'Harding', 20, '1937-07-25'),
(2, 'Ariana', 'Fox', 30, '1992-09-30'),
(3, 'Madelyn', 'Flynn', 35, '1953-03-05'),
(4, 'Fynley', 'Dodd', 40, '1973-03-27'),
(5, 'Aliza', 'Wyatt', 55, '1969-02-14'),
(6, 'Michael', 'Doss', 67, '1964-12-11')
(7, 'Michael', 'Watshon', 37, '1983-12-11');

*ALTER TABLE employees add date_of_birth DATE;

DML - Data Manipulation Language

- **Updating data**

```
UPDATE employees SET date_of_birth = '1988-12-11'
WHERE id_employees = 1 ;
```

```
SET SQL_SAFE_UPDATES=0;
```

```
SELECT * FROM employees
```

DML - Data Manipulation Language

- **Deleting data**

```
DELETE FROM employees WHERE id_employees = 7 ;
```

Exercises

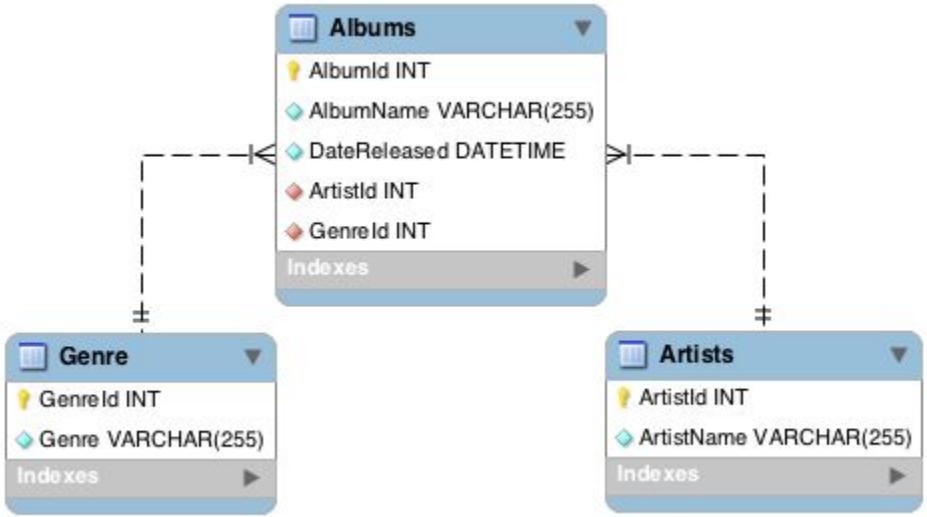
Use the database: humanResources

1. Insert a new entry into employees table:
   a. employeeId - 1,
   b. firstName - John,
   c. lastName - Johnson,
   d. dateOfBirth - 1975-01-01,
   e. phoneNumber - 0-800-800-314,
   f. email - john@johnson.com,
   g. salary - 1000.
2. Update dateOfBirth of John Johnson to 1980-01-01.
3. Delete everything from employees table.
4. Add two more entries in employees:
   a. 1, 'John' , 'Johnson', '1975-01-01', '0-800-800-888' , 'john@johnson.com', 1000
   b. 2,'James' , 'Jameson', '1985-02-02', '0-800-800-999' , 'james@jameson.com', 2000

Exercises

Using DDL

Create a new schema “music” and add the tables following the diagram below



Exercises

- Using DDL create a new schema “db_poems” and add the tables following the diagram below
- Use DML to insert data
- Ids are auto_increment
- Read and search about functions for Date
  - https://www.geeksforgeeks.org/sql-date-functions/
  - https://dataschool.com/learn-sql/dates/
  - https://www.tutorialspoint.com/sql/sql-date-functions.htm
- Insert data using a date function for the attribute ‘date_registered’