

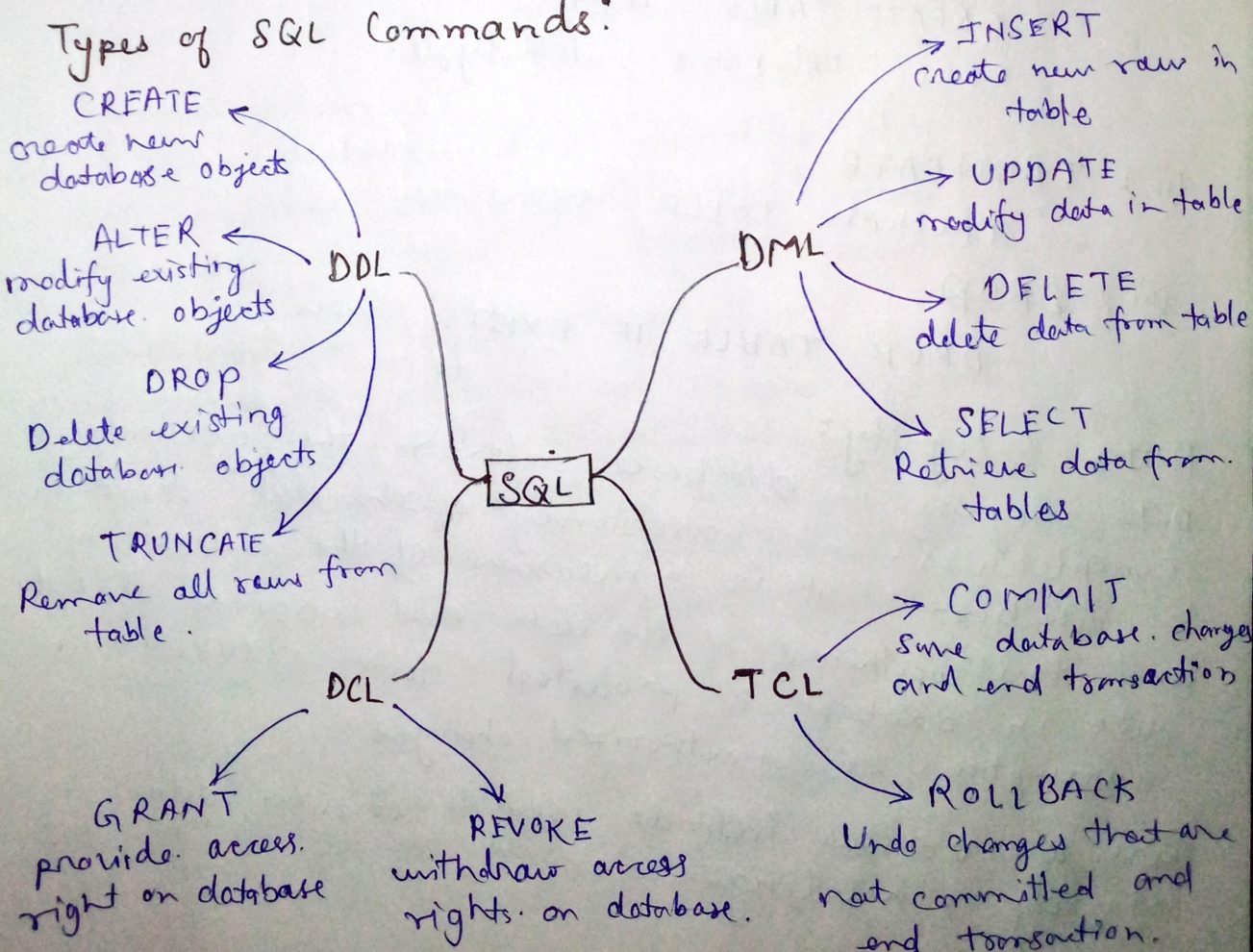
30 Oct 2024

SQL

SQL:

SQL (Structured Query language) is a programming language used for managing and manipulating data in relational databases. It allows you to insert, update, retrieve and delete data in a database. It is widely used for data management in many applications, websites, and businesses. In simple terms, SQL is used to communicate with and control databases.

Types of SQL Commands:



SQL DDL COMMANDS

1. DDL Commands for databases:

(i) CREATE

CREATE DATABASE student.

CREATE DATABASE IF NOT EXISTS student.

(ii) DROP

DROP DATABASE student.

DROP DATABASE IF EXISTS student.

2. DDL Commands for tables:

(i) CREATE

CREATE TABLE users(
col-name data-type.
)

(ii) TRUNCATE

TRUNCATE TABLE users

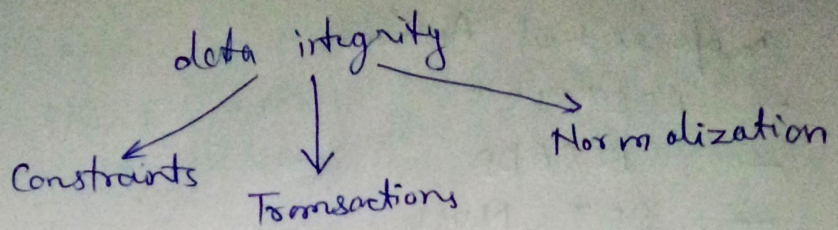
(iii) DROP

DROP TABLE IF EXISTS user.

Data Integrity:

Data integrity in databases, refers to the accuracy, completeness, and consistency of the data stored in a database. It is a measure of the reliability and trustworthiness of the data and ensures that the data in database is protected from errors, corruption, or unauthorized changes.

There are various methods used to ensure data integrity, including:



Constraints:

Constraints in databases are rules or conditions that must be met for data to be inserted, updated or deleted in a database table. They are used to enforce the integrity of the data stored in a database and to prevent data from becoming inconsistent or corrupted.

Transactions:

A sequence of database operations that are treated as a single unit of work.

Normalization:

A design technique that minimizes data redundancy and ensures data consistency by organizing data into separate tables.

Constraints in MySQL:

Constraints in databases are rules or conditions that must be met for data to be inserted, updated or deleted in a database table. They are used to enforce the integrity of the data stored in a database and to prevent data from becoming inconsistent or corrupted.

- NOT NULL
- UNIQUE (combo)
another way to create constraint
- PRIMARY KEY
- AUTO INCREMENT
- CHECK
- DEFAULT
- FOREIGN KEY

Referential Actions

- RESTRICT
- CASCADE
- SET NULL
- SET DEFAULT

(i) NOT NULL :

```
CREATE TABLE users(  
    user_id INTEGER NOT NULL)
```

(ii) UNIQUE :

```
— CREATE TABLE users(  
    user_id INTEGER NOT NULL,  
    name VARCHAR(255) NOT NULL UNIQUE)
```

```
— CREATE TABLE users (  
    user_id INTEGER NOT NULL,  
    name VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL,  
  
    CONSTRAINT users_email_unique UNIQUE (name, email))
```

the second method is useful when we make multiple column unique or we provide a specific name for this here is users_email_unique

(iii) PRIMARY KEY :

```
— CREATE TABLE users (  
    user_id INTEGER NOT NULL PRIMARY KEY)
```

```
— CREATE TABLE users(  
    user_id INTEGER NOT NULL,  
    CONSTRAINT users_primary_key PRIMARY KEY (user_id))
```


(iv) AUTO_INCREMENT:

```
CREATE TABLE users (  
  user_id INTEGER PRIMARY KEY AUTO INCREMENT)
```

(v) CHECK:

```
- CREATE TABLE students(  
  student_id INTEGER PRIMARY KEY AUTO_INCREMENT,  
  name VARCHAR(255) NOT NULL,  
  age INTEGER CHECK (age > 6 AND age < 25))
```

```
- CREATE TABLE students(  
  student_id INTEGER PRIMARY KEY AUTO_INCREMENT,  
  name VARCHAR(255) NOT NULL,  
  age INTEGER,  
  CONSTRAINT student_age_check CHECK (age > 6 AND age < 25))
```

(vi) DEFAULT:

```
CREATE TABLE ticket(  
  ticket_id INTEGER PRIMARY KEY,  
  name VARCHAR(255) NOT NULL,  
  travel_date DATETIME DEFAULT CURRENT_TIMESTAMP)
```

(vii) FOREIGN KEY:

```
CONSTRAINT orders_fk FOREIGN KEY (cid) REFERENCES customers(cid)
```

3. ALTER TABLE command:

The "ALTER TABLE" statement in SQL is used to modify the structure of an existing table. Some of the things that can be done using the ALTER TABLE statement include.

- (i) Add columns
- (ii) Delete columns
- (iii) Modify columns.

(i) Add columns:

- ALTER TABLE customers ADD COLUMN password VARCHAR(255) NOT NULL
- ALTER TABLE customers ADD COLUMN surname VARCHAR(255) NOT NULL AFTER name.
- ALTER TABLE customers.
ADD COLUMN pan-number VARCHAR(255)
AFTER surname.
ADD COLUMN joining-date DATETIME NOT NULL
DEFAULT CURRENT-TIMESTAMP

(ii) Delete Columns:

ALTER TABLE customers DROP COLUMN pan-number

(iii) Modify Columns:

ALTER TABLE customers MODIFY COLUMN
surname INTEGER AUTO-INCREMENT

* using ALTER we can add or deleting the constraints also