SQL DDL (Data Definition Language) interview questions and explanations:-

---

1. **What is DDL? List some common DDL commands.**
    - DDL stands for **Data Definition Language**, used to define, modify, and manage the structure of database objects. Common DDL commands are:
        - **CREATE**: To create new tables, views, indexes, etc.
        - **ALTER**: To modify existing database structures, such as adding columns to a table.
        - **DROP**: To delete objects, like tables or indexes.
        - **TRUNCATE**: To quickly remove all records from a table without logging each row deletion.
        - **RENAME**: To rename a database object, such as a table or column.

2. **Explain the purpose and usage of the CREATE TABLE statement.**

    The CREATE TABLE statement is used to define a new table in a database.
    It specifies the table name, column names, data types, and constraints (e.g., primary key, foreign key, not null).

    Example: -

```
CREATE TABLE employees (
  id INT PRIMARY KEY,
  name VARCHAR(50),
  salary DECIMAL(10,2)
);

CREATE TABLE Employees (
    EmployeeID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    BirthDate DATE
);
```

3. **What are constraints? Name a few and explain their usage.**

- Constraints enforce rules on data in a table. Some common constraints are:
    - **PRIMARY KEY**: Uniquely identifies each record.
    - **FOREIGN KEY**: Enforces a relationship between two tables.
    - **UNIQUE**: Ensures all values in a column are unique.
    - **CHECK**: Restricts values based on a condition.
    - **NOT NULL**: Ensures that a column cannot have NULL values.

4. **What is the difference between ALTER TABLE and DROP TABLE statements?**
    - ALTER TABLE is used to modify the structure of an existing table, such as adding/removing columns, modifying data types, or adding/removing constraints.
    - DROP TABLE is used to completely remove a table and its data from the database.
    - Example ALTER TABLE: `ALTER TABLE employees ADD email VARCHAR(50);`
    - Example DROP TABLE: `DROP TABLE employees;`

5. **Explain the purpose and usage of the TRUNCATE TABLE statement.**
    - TRUNCATE TABLE is used to quickly remove all rows from a table, while preserving the table structure.
    - It is faster than using a DELETE statement without a WHERE clause, as TRUNCATE does not generate transaction logs.
    - Example: `TRUNCATE TABLE orders;`

6. **What is the difference between DROP TABLE and TRUNCATE TABLE?**
    - DROP TABLE removes the table definition and all its data from the database.
    - TRUNCATE TABLE removes all the data from the table, but the table structure (columns, constraints, etc.) remains.
    - DROP TABLE is a DDL operation, while TRUNCATE TABLE is a DML (Data Manipulation Language) operation.
    - DROP TABLE is irreversible, while TRUNCATE TABLE can be rolled back within a transaction.

7. **What is the difference between DELETE, TRUNCATE, and DROP?**
    - **DELETE**: Deletes specific rows from a table. It can be rolled back and generates a log.
    - **TRUNCATE**: Removes all rows from a table without logging each row, so it's faster but cannot be rolled back.
    - **DROP**: Deletes an entire database object (like a table), along with its structure. It is also non-reversible.

8. **Explain the purpose and usage of the RENAME TABLE statement.**
    - RENAME TABLE is used to change the name of an existing table in the database.
    - Example: `RENAME TABLE old_employees TO new_employees;`

9. **What is the difference between ADD COLUMN and ALTER COLUMN in the ALTER TABLE statement?**
    - ADD COLUMN is used to add a new column to an existing table.
    - ALTER COLUMN is used to modify the definition of an existing column, such as changing the data type or column constraints.
    - Example ADD COLUMN: `ALTER TABLE employees ADD email VARCHAR(50);`
    - Example ALTER COLUMN: `ALTER TABLE employees MODIFY salary DECIMAL(12,2);`

10. **Explain the CREATE INDEX command and its use.**

The **CREATE INDEX** command creates an index on a table to speed up data retrieval. For example:

```
CREATE INDEX idx_lastname ON Employees (LastName);
```

11. **Explain the purpose and usage of the CREATE VIEW statement.**

- CREATE VIEW is used to define a virtual table based on the result of a SQL query.
- Views provide a way to encapsulate complex queries and present them as a simple table.

- Example: `CREATE VIEW high_salaries AS SELECT * FROM employees WHERE salary > 80000;`

12. What is the purpose of the IF EXISTS or IF NOT EXISTS clause in DDL statements?

- These clauses are used to conditionally create, alter, or drop database objects (e.g., tables, views, indexes).
- They help prevent errors when the object already exists (or doesn't exist) and allow for more robust and flexible database management.
- Example: `DROP TABLE IF EXISTS old_table;`

13. How can you add a primary key to an existing table?

Eg.. `ALTER TABLE Employees`
`ADD CONSTRAINT PK_EmployeeID PRIMARY KEY (EmployeeID);`

14. How can you delete a column from an existing table?

Eg.. `ALTER TABLE Employees`
`DROP COLUMN Salary;`

15. What is the difference between the primary key and unique constraints?

- A **primary key** uniquely identifies each row and doesn't allow NULL values. A **unique constraint** also enforces uniqueness but allows one NULL value.

16. Can DDL commands be rolled back? Explain why or why not.

- **Answer**: Generally, DDL commands cannot be rolled back because they make immediate changes to the database structure, which are auto-committed in most systems.

Maintaining a User Tracking Table

```
CREATE TABLE user_activity (
 id SERIAL PRIMARY KEY,
 user_id INT NOT NULL,
 action VARCHAR(50) NOT NULL,
 timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Modifying a Existing Table Structure

```
ALTER TABLE user_activity
ADD COLUMN ip_address VARCHAR(15),
ADD COLUMN device VARCHAR(50);
```

Renaming a Table and Column

```
RENAME TABLE user_activity TO user_logs;
ALTER TABLE user_logs
RENAME COLUMN action TO activity;
```

Creating an Index for Performance Optimization

```
CREATE INDEX idx_user_logs_user_id
ON user_logs (user_id);
```