



<b>Experiment No.3</b>
------------------------

Create a database using Data Definition Language(DDL) and apply integrity constraints for the specified system
--

**Aim:-** Write a query to create tables for each relation in the relational schema of experiment no.2. Apply drop and alter commands on those tables.

**Objective:-** To learn commands of Data Definition Language(DDL) to create and define databases, and also learn to apply integrity constraints for the specified system.

**Theory:**

DDL Commands & Syntax:-

Data Definition Language (DDL) is a subset of SQL and a part of DBMS(Database Management System). DDL consist of Commands to commands like CREATE, ALTER, TRUNCATE and DROP. These commands are used to create or modify the tables in SQL.

DDL Commands:

1. Create
2. Alter
3. truncate
4. drop 5. Rename

**CREATE:**

This command is used to create a new table in SQL. The user must give information like table name, column names, and their data types.

Syntax –CREATE TABLE table\_name

(



## Vidyavardhini's College of Engineering and Technology

### Department of Artificial Intelligence & Data Science

---

column\_1 datatype,

column\_2 datatype,

column\_3 datatype,

.....

);

**ALTER :**

This command is used to add, delete or change columns in the existing table. The user needs to know the existing table name and can add, delete, or modify tasks easily.

Syntax –

ALTER TABLE table\_name

ADD column\_name datatype;

**TRUNCATE :**

This command is used to remove all rows from the table, but the structure of the table still exists.

Syntax –

TRUNCATE TABLE table\_name;

**DROP :**

This command is used to remove an existing table along with its structure from the Database.

Syntax –

DROP TABLE table\_name;

**RENAME :**

It is possible to change name of table with or without data in it using simple RENAME command. We can rename any table object at any point of time.

Syntax – RENAME TABLE <Table Name> To <New\_Table\_Name>;



# Vidyavardhini's College of Engineering and Technology

## Department of Artificial Intelligence & Data Science

---

### Implementation:

#### SQL Queries:

##### 1.CREATE

##### -- Create a table to store document metadata

```
CREATE TABLE Documents (  
    DocumentID INT PRIMARY KEY,  
    Title VARCHAR(255) NOT NULL,  
    Author VARCHAR(100),  
    UploadDate DATE,  
    FileSizeInBytes BIGINT,  
    ContentType VARCHAR(50)  
);
```

##### -- Create a table to store document versions

```
CREATE TABLE DocumentVersions (  
    VersionID INT PRIMARY KEY,  
    DocumentID INT NOT NULL,  
    VersionNumber INT NOT NULL,  
    Content BLOB,  
    UploadDate DATE,  
    CONSTRAINT fk_DocumentVersions_Documents  
        FOREIGN KEY (DocumentID) REFERENCES Documents(DocumentID)  
);
```

##### -- Create a table to store user information

```
CREATE TABLE Users (  
    UserID INT PRIMARY KEY,  
    Username VARCHAR(50) UNIQUE NOT NULL,  
    Email VARCHAR(100) UNIQUE NOT NULL,  
    PasswordHash VARCHAR(64) NOT NULL
```



# Vidyavardhini's College of Engineering and Technology

## Department of Artificial Intelligence & Data Science

---

);

**-- Create a table to track document access permissions**

```
CREATE TABLE DocumentPermissions (  
    PermissionID INT PRIMARY KEY,  
    DocumentID INT NOT NULL,  
    UserID INT NOT NULL,  
    PermissionType ENUM('Read', 'Write', 'Admin') NOT NULL,  
    CONSTRAINT fk_DocumentPermissions_Documents  
        FOREIGN KEY (DocumentID) REFERENCES Documents(DocumentID),  
    CONSTRAINT fk_DocumentPermissions_Users  
        FOREIGN KEY (UserID) REFERENCES Users(UserID)  
);
```

### 2.ALTER

**-- Add a new column to DocumentVersions**

```
ALTER TABLE DocumentVersions  
ADD COLUMN VersionNotes VARCHAR(255);
```

### 3.TRUNCATE

**-- Truncate the DocumentVersions table**

```
TRUNCATE TABLE DocumentVersions;
```

### 4.DROP

**-- Drop the Users table**

```
DROP TABLE Users;
```

### 5.USER

**-- Rename the DocumentPermissions table**

```
RENAME TABLE DocumentPermissions TO AccessPermissions;
```



# Vidyavardhini's College of Engineering and Technology

## Department of Artificial Intelligence & Data Science

---

### Conclusion:

1. Explain the concept of constraints in DDL. How are constraints used to enforce data integrity?

#### ➔ Constraints in DDL:

- **Definition:** Constraints are rules or conditions applied to database tables to ensure data integrity and consistency.
- **Purpose:**
  - **Data Integrity:** Constraints prevent invalid or inconsistent data from being inserted, updated, or deleted.
  - **Business Rules:** Constraints enforce business rules and maintain the accuracy of data.
- **Common Constraints:**
  - **Primary Key (PK):** Ensures uniqueness and identifies a unique record in a table.
  - **Foreign Key (FK):** Establishes relationships between tables by referencing the primary key of another table.
  - **Unique Constraint:** Ensures that values in a column are unique (but allows NULLs).
  - **Check Constraint:** Validates data against a specific condition (e.g., age > 18).
  - **Not Null Constraint:** Ensures that a column cannot have NULL values.
  - **Default Constraint:** Sets a default value for a column if no value is specified during insertion.

2. What is the significance of data types in DDL? Provide examples of commonly used data types in DDL.

#### ➔ Significance of Data Types in DDL:

- **Definition:** Data types define the kind of data that can be stored in a column.
- **Importance:**
  - **Storage Efficiency:** Data types determine how much space a value occupies in storage.
  - **Data Validation:** Data types enforce rules (e.g., numeric, string, date) and prevent invalid data.
  - **Query Optimization:** Proper data types improve query performance.
- **Examples of Common Data Types:**
  - **INTEGER (INT):** Whole numbers (e.g., employee ID).
  - **VARCHAR(n):** Variable-length character strings (e.g., names, addresses).
  - **DATE:** Represents dates (e.g., birthdate).



# Vidyavardhini's College of Engineering and Technology

## Department of Artificial Intelligence & Data Science

---

- **DECIMAL(p, s):** Fixed-point decimal numbers (e.g., salary).
- **BOOLEAN:** Represents true/false values (e.g., active status).
- **BLOB (Binary Large Object):** Stores binary data (e.g., images, files).