SGD LAB EXP – 3A

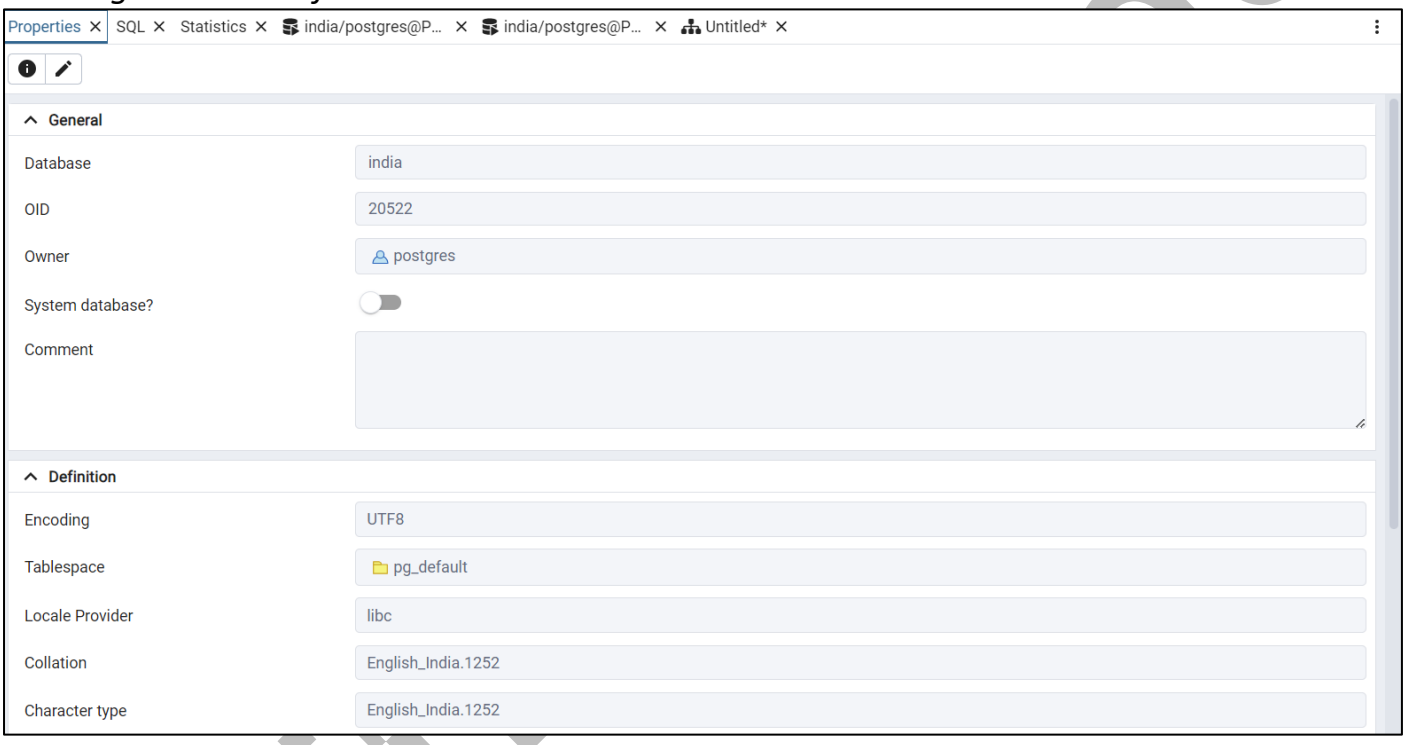
Name : Aditi Chhajed; **Reg. No.** : 221081009
Branch : IT ; **Course Instructor** : Prof. Vedashree Awati

Aim:

To execute DML,DCL and DDL queries.

Implementation :

Creating a database first.



a. Creating tables

```sql
CREATE TABLE Districts (
    id INT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    state_id INT,
    division VARCHAR(50),
    area DECIMAL(10,2),
    population BIGINT,
    FOREIGN KEY (state_id) REFERENCES States(id)
);
CREATE TABLE Rivers (
    id INT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    origin_state_id INT,
    total_length DECIMAL(10,2),
    dams_built TEXT,
    FOREIGN KEY (origin_state_id) REFERENCES States(id)
);
CREATE TABLE NationalParks (
    id INT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    location VARCHAR(100),
    area DECIMAL(10,2),
    establishment_year INT,
    visitors INT,
    river_ids TEXT
);
CREATE TABLE Mountains (
    id INT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    location VARCHAR(100),
    elevation DECIMAL(7,2),
    length DECIMAL(10,2),
    rock_type VARCHAR(50),
    highest_peak_name VARCHAR(50),
    highest_peak_elevation DECIMAL(7,2)
);
```

Data Output | Messages | Notifications

```
CREATE TABLE

Query returned successfully in 115 msec.
```

b. *To create Users and assign privileges.*

```
CREATE USER geo_user WITH PASSWORD 'root';

GRANT ALL PRIVILEGES ON DATABASE india TO geo_user;

REVOKE DELETE ON TABLE States FROM geo_user;

```

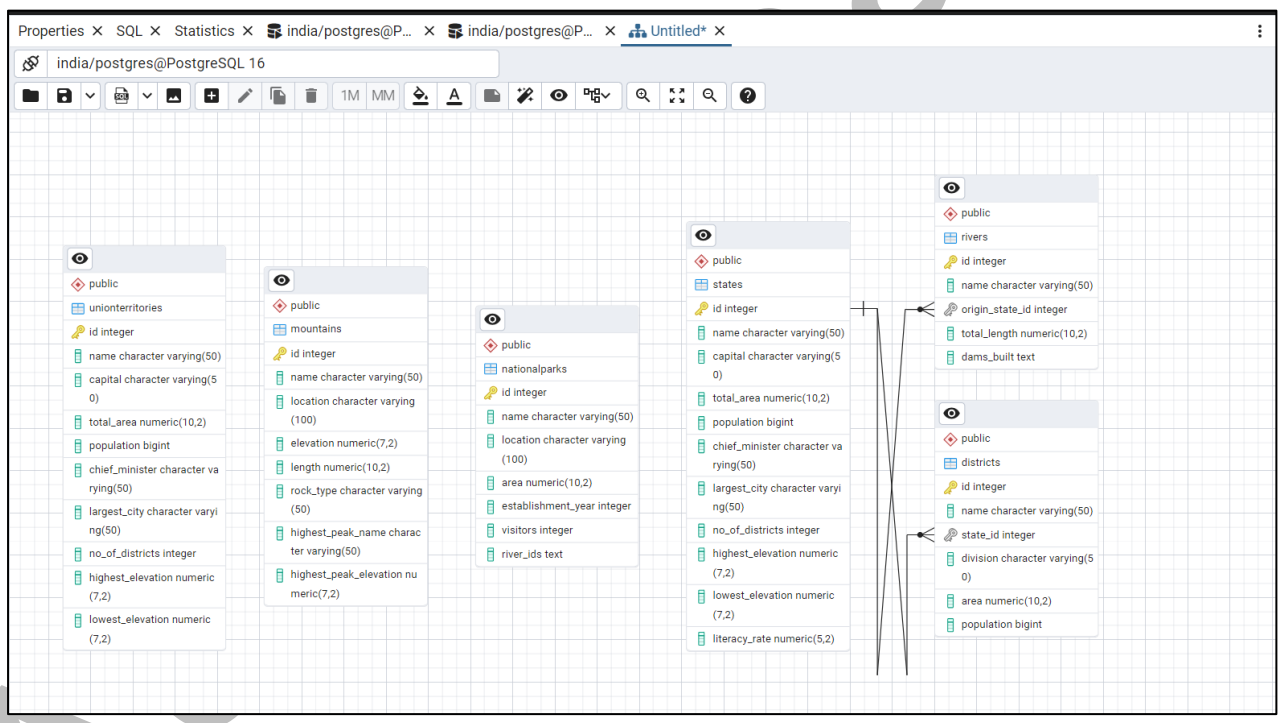
Data Output | Messages | Notifications

```
REVOKE

Query returned successfully in 123 msec.
```

c. *To Generate an ER diagram in postGIS.*

Expand your database in the Object Browser. Right-click on the ***database*** *(india) and select* ***ERD Tool*** *from the context menu. This is what we get.*



d. *Insert data into the created tables.*

```sql
INSERT INTO States (id, name, capital, total_area, population, chief_minister, largest_city, no_of_districts, highest_elevation, lowest_ele
VALUES
(1, 'Maharashtra', 'Mumbai', 307713, 112374333, 'Eknath Shinde', 'Mumbai', 36, 1646.0, 0.0),
(2, 'Karnataka', 'Bengaluru', 191791, 61095297, 'Siddaramaiah', 'Bengaluru', 31, 1925.0, 0.0),
(3, 'Kerala', 'Thiruvananthapuram', 38863, 33406061, 'Pinarayi Vijayan', 'Kochi', 14, 2695.0, 0.0);

INSERT INTO UnionTerritories (id, name, capital, total_area, population, chief_minister, largest_city, no_of_districts, highest_elevation,
VALUES
(1, 'Delhi', 'New Delhi', 1483, 16787941, 'Arvind Kejriwal', 'Delhi', 11, 239.0, 0.0),
(2, 'Chandigarh', 'Chandigarh', 114, 1055450, 'Administrator', 'Chandigarh', 1, 350.0, 304.0),
(3, 'Lakshadweep', 'Kavaratti', 32, 64473, 'Administrator', 'Kavaratti', 1, 2.0, 0.0);

INSERT INTO Districts (id, name, state_id, division, area, population)
VALUES
(1, 'Thane', 1, 'Konkan', 4214.0, 11054131),
(2, 'Ernakulam', 3, 'Central', 3068.0, 3279860),
(3, 'Bangalore Urban', 2, 'Bangalore Division', 2196.0, 9651494);

INSERT INTO Rivers (id, name, origin_state_id, total_length, dams_built)
VALUES
(1, 'Ganga', 2, 2525.0, 'Tehri Dam, Haridwar Barrage'),
(2, 'Godavari', 1, 1465.0, 'Jayakwadi, Sriram Sagar'),
(3, 'Periyar', 3, 244.0, 'Idukki, Mullaperiyar');
INSERT INTO NationalParks (id, name, location, area, establishment_year, visitors, river_ids)
VALUES
(1, 'Kaziranga National Park', 'Assam', 858.98, 1905, 100000, 'Brahmaputra'),
(2, 'Periyar National Park', 'Kerala', 925.0, 1982, 200000, 'Periyar'),
(3, 'Jim Corbett National Park', 'Uttarakhand', 1318.0, 1936, 700000, 'Ramganga');

INSERT INTO Mountains (id, name, location, elevation, length, rock_type, highest_peak_name, highest_peak_elevation)
VALUES
(1, 'Himalayas', 'India-Nepal Border', 8848.86, 2400.0, 'Metamorphic', 'Mount Everest', 8848.86),
(2, 'Western Ghats', 'India', 2695.0, 1600.0, 'Igneous', 'Anamudi', 2695.0),
(3, 'Vindhya Range', 'India', 881.0, 1050.0, 'Sedimentary', 'Sad-bhawna Hill', 881.0);
```

e. *Query the tables*

```sql
SELECT name, capital, population FROM States WHERE population > 50000000;
```

Data Output Messages Notifications

	name character varying (50)	capital character varying (50)	population bigint
1	Maharashtra	Mumbai	112374333
2	Karnataka	Bengaluru	61095297

```sql
SELECT name, dams_built FROM Rivers WHERE LENGTH(dams_built) > 0;
```

Data Output Messages Notifications

	name character varying (50)	dams_built text
1	Ganga	Tehri Dam, Haridwar Barrage
2	Godavari	Jayakwadi, Sriram Sagar
3	Periyar	Idukki, Mullaperiyar

```sql
SELECT name FROM Districts WHERE state_id = (SELECT id FROM States WHERE name = 'Maharashtra');
```

Data Output Messages Notifications

	name character varying (50)
1	Thane

f. Use other DML, DCL and DDL commands you know to query the tables.

1. ***Update.***

```
UPDATE Districts SET population = 12000000 WHERE name = 'Thane';
select * from Districts;
```

Updated table

	id [PK] integer	name character varying (50)	state_id integer	division character varying (50)	area numeric (10,2)	population bigint
1	2	Ernakulam	3	Central	3068.00	3279860
2	3	Bangalore Urban	2	Bangalore Division	2196.00	9651494
3	1	Thane	1	Konkan	4214.00	12000000

2. ***Delete.***

```
DELETE FROM Rivers WHERE name = 'Ganga';
select * from Rivers;
```

Updated table :

	id [PK] integer	name character varying (50)	origin_state_id integer	total_length numeric (10,2)	dams_built text
1	2	Godavari	1	1465.00	Jayakwadi, Sriram Sagar
2	3	Periyar	3	244.00	Idukki, Mullaperiyar

3. ***Alter.***

```
ALTER TABLE States ADD COLUMN literacy_rate DECIMAL(5,2);
select * from States;
```

Updated table :

Data Output Messages Notifications

	varying (50)	capital character varying (50)	total_area numeric (10,2)	population bigint	chief_minister character varying (50)	largest_city character varying (50)	no_of_districts integer	highest_elevation numeric (7,2)	lowest_elevation numeric (7,2)	literacy_rate numeric (5,2)
1	ntra	Mumbai	307713.00	112374333	Eknath Shinde	Mumbai	36	1646.00	0.00	[null]
2	a	Bengaluru	191791.00	61095297	Siddaramaiah	Bengaluru	31	1925.00	0.00	[null]
3		Thiruvananthapuram	38863.00	33406061	Pinarayi Vijayan	Kochi	14	2695.00	0.00	[null]

4. ***Drop :***

```
DROP TABLE Rivers;
select * from Rivers;
```

Data Output Messages Notifications

```
ERROR:  relation "rivers" does not exist
LINE 1: select * from Rivers;
                      ^

SQL state: 42P01
Character: 15
```

5. *DCL Queries:*

```
GRANT SELECT, INSERT ON NationalParks TO geo_user;
REVOKE DELETE ON Rivers FROM geo_user;
CREATE ROLE readonly_user;
GRANT SELECT ON ALL TABLES IN SCHEMA public TO readonly_user;
```

Conclusion:

In this project, we successfully created a relational database named "india" to store geographical and administrative information about states, union territories, districts, rivers, national parks, and mountain ranges. By executing DDL, DML, and DCL queries, we designed and populated the database with structured data.

- DDL operations were used to define tables with relevant attributes, primary keys, and foreign key relationships, ensuring proper data integrity.

- DCL operations were applied to create users and assign appropriate privileges, enforcing controlled access to the database.

- DML queries allowed us to insert, update, and retrieve data from the tables, demonstrating practical applications of the database schema.

- We also generated an ER diagram in PostGIS, visualizing the relationships between various tables.

Overall, this exercise showed us how relational databases can be used to organize complex, interconnected datasets and how various SQL operations help maintain data consistency, enforce relationships, and manage access control. The result is a robust and scalable model for managing geographical and administrative data for the country.