### **Lab 7- Check Constraint in PostgreSQL**

Below is a lab exercise to practice integrity constraints in PostgreSQL:

**Connect to PostgreSQL:**

* Open the PostgreSQL command-line interface or use a graphical client like pgAdmin.

**Create a Table with Constraints:**

* Create a sample table named students with the following columns and constraints:

CREATE TABLE students (

id INT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

age INT CHECK (age >= 18),

email VARCHAR(100) UNIQUE

);

**Insert Data Violating Constraints:**

* Try inserting some data that violates the constraints to see how PostgreSQL handles it:

INSERT INTO students (id, name, age, email) VALUES (101, 'John Doe', 16, 'john@example.com');

* You should receive an error message indicating that the CHECK constraint has been violated.

**Insert Data Meeting Constraints:**

* Insert data that adheres to the constraints:

INSERT INTO students (id, name, age, email) VALUES (101, 'Jane Smith', 20, 'jane@example.com');

**Update Data Violating Constraints:**

* Try updating data that violates the constraints:

UPDATE students SET age = 15 WHERE name = 'Jane Smith';

* You should receive an error message indicating that the CHECK constraint has been violated.

**Update Data Meeting Constraints:**

* Update data that adheres to the constraints:

UPDATE students SET age = 21 WHERE name = 'John Doe';

**View Data:**

* Retrieve and view the data from the students table:

SELECT \* FROM students;

By performing these operations, you can familiarize yourself with the implementation and handling of integrity constraints in PostgreSQL. You can execute these commands in the PostgreSQL command-line interface or any PostgreSQL client tool.