***TASK 4:-***

* **CREATE THE TABLE STUDENT:-**

CREATE TABLE student (

student\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL,

date\_of\_birth DATE,

email VARCHAR(100) UNIQUE

);

* **INSERT DATA INTO TABLE STUDENT:-**

INSERT INTO student (first\_name, last\_name, date\_of\_birth, email)

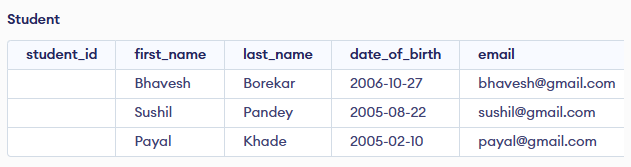
VALUES ('Bhavesh', 'Borekar', '2006-10-27', 'bhavesh@gmail.com'),

('Sushil', 'Pandey', '2005-08-22', 'sushil@gmail.com'),

('Payal', 'Khade', '2005-02-10', 'payal@gmail.com');

* **SHOW THE TABLE STUDENT**

SELECT \* FROM student;



* **SIMULATE RESTORE BY RE-INSERTING DATA**

INSERT INTO student (first\_name, last\_name, date\_of\_birth, email)

VALUES ('Bhavesh', 'Borekar', '2006-10-27', 'bhavesh@gmail.com'),

('Sushil', 'Pandey', '2005-08-22', 'sushil@gmail.com'),

('Payal', 'Khade', '2005-02-10', 'payal@gmail.com');

* **VERIFYING RESTORATION**

SELECT \* FROM student;

* **BACKUP AND RESTORE IN POSTRESQL**
* **CREATE THE TABLE STUDENT:-**

CREATE TABLE student (

student\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL,

date\_of\_birth DATE,

email VARCHAR(100) UNIQUE

);

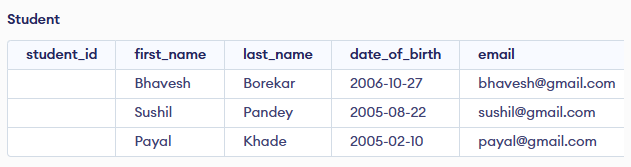
* **INSERT DATA INTO TABLE STUDENT:-**

INSERT INTO student (first\_name, last\_name, date\_of\_birth, email)

VALUES ('Bhavesh', 'Borekar', '2006-10-27', 'bhavesh@gmail.com'),

('Sushil', 'Pandey', '2005-08-22', 'sushil@gmail.com'),

('Payal', 'Khade', '2005-02-10', 'payal@gmail.com');



* **Backup and Recovery Script**

1. **Backup SQL Script (MySQL/PostgreSQL Simulation)**

SELECT \* FROM student;

1. **Restore SQL Script**

INSERT INTO student (first\_name, last\_name, date\_of\_birth, email)

VALUES ('Bhavesh', 'Borekar', '2006-10-27', 'bhavesh@gmail.com'),

('Sushil', 'Pandey', '2005-08-22', 'sushil@gmail.com'),

('Payal', 'Khade', '2005-02-10', 'payal@gmail.com');

#### ****Backup Process****

**Step 1**: Create the student table and insert sample data.

**Step 2**: Use a SELECT \* FROM student query to simulate a backup by copying the results.

**Step 3**: Save the query result manually as a backup for restoration.

**Backup Command:**

mysql -u root -p student > task4.sql

* -u username: Specifies the MySQL username.
* -p: Prompts you for the password of the MySQL user.
* database\_name: The name of the database you want to back up.
* backup\_file.sql: The name of the SQL file where the database will be backed up.
* **Restore Process**

**Step 1**: Use the INSERT INTO student command to restore the data from the backup.

**Step 2**: Verify that the restored data matches the original data by running the SELECT \* FROM student query.

**Restore Command:**

mysql -u root -p student< task4.sql

* -u username: Specifies the MySQL username (e.g., root).
* -p: Prompts you for the password.
* database\_name: The name of the database you want to restore.
* backup\_file.sql: The .sql backup file you want to restore.

### 2. ****Backup and Restore in PostgreSQL****

#### ****Backing Up a PostgreSQL Database****

In PostgreSQL, the pg\_dump command is used to back up a database.

**Backup Command:**

pg\_dump -U username -W -F c student > task4.dump

* -U username: Specifies the PostgreSQL username (e.g., postgres).
* -W: Prompts for the PostgreSQL password.
* -F c: Specifies the format of the backup (custom format .dump).
* database\_name: The name of the database you want to back up.
* backup\_file.dump: The backup file where the database will be saved.

***Command launch***

pg\_dump -U postgres -W -F c student> task.dump

#### ****Restoring a PostgreSQL Database****

To restore a PostgreSQL database from a backup, you can use the pg\_restore command.

**Restore Command:**

* ***Command launch***

pg\_restore -U postgres -d student -W task.dump

* -U username: Specifies the PostgreSQL username.
* -d database\_name: The name of the database to restore into.
* -W: Prompts for the PostgreSQL password.
* backup\_file.dump: The .dump backup file you want to restore.

#### ****Backing Up a SQL Server Database****

In SQL Server, you can use the BACKUP command to create a backup.

**Backup Command (T-SQL):**

BACKUP DATABASE database2

TO DISK = 'C:\user\task4.bak';

* database2: The name of the database you want to back up.
* 'C:\user\task4.bak':The location and name of the backup file.

**Restoring a SQL Server Database**

To restore a database, use the RESTORE command.

**Restore Command (T-SQL):**

RESTORE DATABASE database2

FROM DISK = 'C:\user\task4.bak';

* database2: The name of the database you want to restore into.
* 'C:\user\task4.bak': The backup file from which you want to restore.