

Student Name: Sahil Hans
Branch: M.C.A(A.I & M.L)
Semester: Second Sem
Subject Name: TECHINICAL SKILLS

UID: 25MCI10088
Section/Group: MAM-1 A
Date of Performance: 12/01/2026
Subject Code:

WORKSHEET 1.2

AIM: To design and implement a sample database system using DDL, DML, and DCL commands, including database creation, data manipulation, schema modification, and role-based access control to ensure data integrity and secure, read-only access for authorized users.

S/W Requirement: Oracle Database Express Edition and pgAdmin

OBJECTIVES:

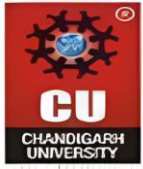
To gain practical experience in implementing Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL) operations in a real database environment. This will also include implementing role-based privileges to secure data.

Given:

You are given a database table named **Students** that stores academic and personal information of students in an institution. Each record in the table includes the Student ID, Student Name, City, and Percentage.

Table structure and data:

```
CREATE TABLE Students (  
    StudentID INT PRIMARY KEY,  
    StudentName VARCHAR(50),  
    City VARCHAR(50),  
    Percentage DECIMAL(5,2)  
);  
  
INSERT INTO Students (StudentID, StudentName, City, Percentage) VALUES  
(1, 'Aarav', 'Delhi', 96.5),  
(2, 'Neha', 'Mumbai', 89.2),  
(3, 'Rohit', 'Delhi', 91.0),
```



(4, 'Priya', 'Chandigarh', 97.3),
(5, 'Karan', 'Mumbai', 95.8),
(6, 'Simran', 'Chandigarh', 88.4),
(7, 'Aman', 'Delhi', 99.1),
(8, 'Riya', 'Mumbai', 92.6),
(9, 'Sahil', 'Chandigarh', 85.9),
(10, 'Ananya', 'Delhi', 94.7),
(11, 'Vikas', 'Mumbai', 97.9),
(12, 'Pooja', 'Chandigarh', 90.5);

	studentid [PK] integer	studentname character varying (50)	city character varying (50)	percentage numeric (5,2)
1	1	Aarav	Delhi	96.50
2	2	Neha	Mumbai	89.20
3	3	Rohit	Delhi	91.00
4	4	Priya	Chandigarh	97.30
5	5	Karan	Mumbai	95.80
6	6	Simran	Chandigarh	88.40
7	7	Aman	Delhi	99.10
8	8	Riya	Mumbai	92.60
9	9	Sahil	Chandigarh	85.90
10	10	Ananya	Delhi	94.70
11	11	Vikas	Mumbai	97.90
12	12	Pooja	Chandigarh	90.50

(i) Count of students with percentage **greater than 95**

- Using **CASE statement**

Query:

```
SELECT COUNT(CASE WHEN Percentage > 95 THEN 1 END) AS CountAbove95  
FROM Students;
```

	countabove95 bigint
1	5

- And **without CASE statement**

Query:

```
SELECT COUNT(*) AS CountAbove95 FROM Students WHERE Percentage > 95;
```

	countabove95 bigint
1	5

(ii) Find the average percentage of students in each city whose percentage is greater than 95

- Using **CASE statement**
- And **order by percentage in descending order**

Query:

```
SELECT City, AVG(CASE WHEN Percentage > 95 THEN Percentage END) AS AvgPercentage
FROM Students GROUP BY City ORDER BY AvgPercentage DESC;
```

	city character varying (50)	avgpercentage numeric
1	Delhi	97.8000000000000000
2	Chandigarh	97.3000000000000000
3	Mumbai	96.8500000000000000

Learning Outcomes:

1. Understand the structure of a relational database table and the use of appropriate data types for storing student information.
2. Learn how to create tables using the CREATE TABLE statement with primary key constraints.
3. Gain practical knowledge of retrieving data using conditional filtering with the WHERE clause.
4. Understand the use of the **CASE statement** for conditional logic in SQL queries.
5. Learn how to apply aggregate functions such as COUNT() and AVG() to perform statistical analysis on data.
6. Develop the ability to group data city-wise using GROUP BY and sort results using ORDER BY in descending order.