**NAME** :- Manish Shashikant Jadhav

**UID** :- 2023301005.

**BRANCH** :- Comps -B. **BATCH:** B.

**EXPERIMENT 8: To create a different view of database.**

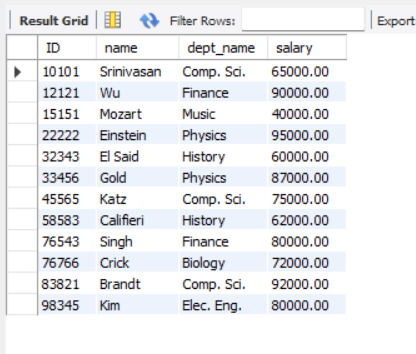
**SUBJECT** :- DBMS (DATABASE MANAGEMENT SYSTEM)

* **Create and update view for university database and execute any 5 queries on views.**

1. **View 1:**

create view vinstructor as (select \* from instructor where salary>30000);

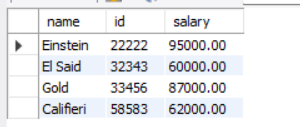
select \* from vinstructor;



* **QUERY: Replace the view vinstructor and display name id and salary where dept\_name is history or physics.**

create or replace view vinstructor as

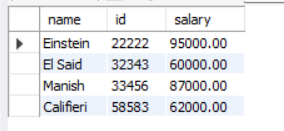
(select name,id,salary from instructor where dept\_name="History" or dept\_name="Physics");



* **QUERY : Update name in view instructor set it to manish where name is Gold.**

update vinstructor set name="Manish"

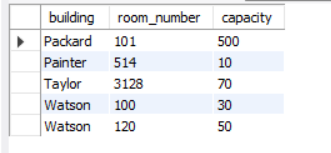
where name="Gold";



1. **View 2:**

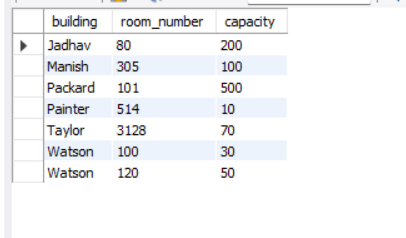
Create view vclassroom as(select \* from classroom);

select \* from vclassroom;



* **QUERY : Insert new values in vclassroom**

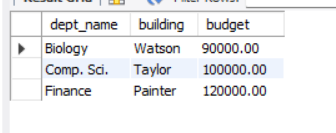
insert into vclassroom(building,capacity,room\_number) values ("Manish",100,305),("Jadhav",200,80);



1. **View 3:**

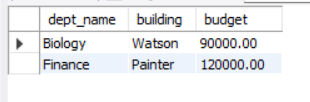
create view vdept as (select \* from department where budget>=90000);

select \* from vdept;

****

* **QUERY: Delete from vdept where building name is Painter**

Delete from vdept where building="Taylor";



**Advantages of SQL views:**

**1. Simplicity and Abstraction:** Views simplify complex queries by abstracting underlying table structures.

**2. Data Security:** Views provide a layer of security by allowing users to access specific columns, rows, or aggregated data without direct table access.

**3. Code Reusability:** Views promote code reusability by encapsulating commonly used queries, reducing redundancy in SQL code.

**4. Performance Optimization:** Views can enhance performance by precomputing results or aggregations, saving computation time during query execution.

**Disadvantages of SQL views:**

**1. Performance Overhead:** Views may introduce performance overhead as they require additional processing to generate virtual result sets.

**2. Complexity in Maintenance:** Views can make database maintenance more complex, especially when dealing with nested or dependent views.

**3. Limited Update Functionality**: Updating data through views can be restricted, leading to potential challenges in certain scenarios.

**4. Dependency Management:** Changes in underlying table structures may impact views, necessitating careful management of dependencies to avoid errors.

**Conclusion:** Hence by completing this experiment I came to know about how to create a different view of database.