Ex.No.: 4		
Date:	16/08/2024	

1) Add a table-level PRIMARY KEY constraint to the EMP table on the ID column. The constraint should be named at creation. Name the constraint my\_emp\_id\_pk.

```
alter table EMP1 add constraint my emp id pk PRIMARY KEY(ID);
```

2) Create a PRIMAY KEY constraint to the DEPT table using the ID colum. The constraint should be named at creation. Name the constraint my\_dept\_id\_pk.

```
alter table DEPT add constraint my_dept_id_pk PRIMARY KEY(ID);
```

 Add a column DEPT\_ID to the EMP table. Add a foreign key reference on the EMP table that ensures that the employee is not assigned to nonexistent department. Name the constraint my\_emp\_dept\_id\_fk.

```
alter table emp
add DEPT_ID Numbe(10);

alter table emp
add constraint my emp dept id fk FOREIGN KEY(DEPT ID) references dept(ID);
```

4) Modify the EMP table. Add a COMMISSION column of NUMBER data type, precision 2, scale 2. Add a constraint to the commission column that ensures that a commission value is greater than zero.

```
alter table emp
add COMMISSION Number(2,2);
alter table emp
add CONSTRAINT commission_gt_zero CHECK(COMMISSION > 0);
```

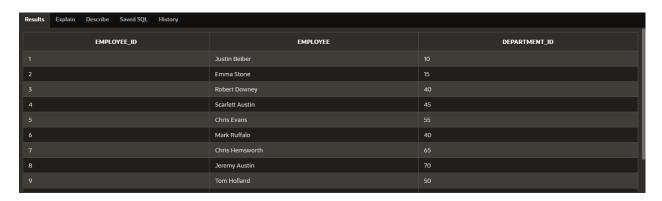
Ex.No.: 5		CDE ATING VIEWS	
<b>Date:</b> 23/08/20	24	CREATING VIEWS	

1) Create a view called EMPLOYEE\_VU based on the employee numbers, employee names and department numbers from the EMPLOYEES table. Change the heading for the employee name to EMPLOYEE.

create view EMPLOYEE\_VU as select employee\_id , first\_name || ' ' || last\_name as "EMPLOYEE", department\_id from employees;

2) Display the contents of the EMPLOYEES\_VU view.

select \* from EMPLOYEE VU;



Select the view name and text from the USER\_VIEWS data dictionary views.

select VIEW\_NAME, TEXT from USER\_VIEWS where VIEW NAME = 'EMPLOYEE VU';



4) Using your EMPLOYEES\_VU view, enter a query to display all employees names and Department.

## FROM EMPLOYEE\_VU;



5) Create a view named DEPT50 that contains the employee number, employee last names and department numbers for all employees in department 50.Label the view columns EMPNO, EMPLOYEE and DEPTNO. Do not allow an employee to be reassigned to another department through the view.

CREATE VIEW DEPT50 AS
SELECT employee\_id AS EMPNO,
employee AS EMPLOYEE,
department\_id AS DEPTNO
FROM EMPLOYEE\_VU
WHERE department\_id = 50
WITH READ ONLY;



6) Display the structure and contents of the DEPT50 view.

## Desc dept50;



7) Attempt to reassign Matos to department 80.

```
UPDATE EMPLOYEES
SET department_id = 80
WHERE first name = 'Matos';
```

8) Create a view called SALARY\_VU based on the employee last names, department names, salaries, and salary grades for all employees. Use the Employees, DEPARTMENTS and JOB\_GRADE tables. Label the column Employee, Department, salary, and Grade respectively.

