

9,10,11 & 12 (expS)

Exp 9. DDL

1. Create the table called product with attributes pno, pname, pprice, qty and total add the constraint not null for the required columns and add check constraint to price >0.

Query:

```
CREATE TABLE product (  
  Pno number(10),  
  Pname varchar(20),  
  Pprice number(10) CHECK (Pprice > 10),  
  Qty number(10),  
  Total number(10,2)  
);
```

2. Add the constraint primary key to pno by modifying the table.

Query:

```
ALTER TABLE product ADD CONSTRAINT pk PRIMARY KEY (Pno);
```

3. Insert five records to product table at run time.

Query:

```
INSERT INTO product(Pno, Pname, Pprice, Qty)  
VALUES(&Pno, '&Pname', &Pprice, &Qty, &P+&q);
```

4. Then add new column special_offer.

Query:

```
ALTER TABLE product ADD special_offer number(10);
```

5. Delete first three records.

Query:

```
DELETE FROM product WHERE rownum < 4;
```

6. Truncate the table product.

Query:

```
TRUNCATE TABLE product;
```

Exp 10. VIEWS

1. Create a view empvu20 which has all employees details work for the deptno 20.

Query:

```
CREATE OR REPLACE VIEW empvu20 AS SELECT * FROM emp WHERE deptno = 20;
```

2. Write a query to create view on emp and dept table which has details like ename, deptno, sal, dname and loc for deptno 10, and salary should be between 1000 to 2000 and who are not drawing commission.

Query:

```
CREATE VIEW emp_dept AS  
SELECT ename, e.deptno, sal, dname, loc  
FROM emp e, dept d  
WHERE e.deptno = d.deptno  
AND e.deptno = 10  
AND sal BETWEEN 1000 AND 2000  
AND comm IS NULL;
```

3. Write a query to create view as EMPDETAILS from the tables EMP, DEPT and SALGRADE which contains empno, ename, job, sal, deptno, dname, loc and grade, where employees working as ANALYST and are not belongs to deptno 10 and 20.

Query:

```
CREATE VIEW EMPDETAILS (id_number, name, job, sal, deptno, dname, loc,  
grade) AS  
SELECT empno, ename, job, sal, dept.deptno, dname, loc, grade  
FROM emp, dept, salgrade  
WHERE dept.deptno NOT IN (10, 20);
```

4. Modify the EMPDETAILS view by using required clause. Add an alias for each column name.

Query:

```
CREATE OR REPLACE VIEW empview AS  
SELECT empno, ename, job, sal, deptno  
FROM emp  
WHERE deptno = 10  
WITH CHECK OPTION CONSTRAINT chk;
```

5. Create a complex view dept_sum_view, store dname, minimum salary, maximum salary and average salary department wise.

Query:

```
CREATE VIEW dept_sum_vu(name, minsal, maxsal, avgsal) AS  
SELECT d.dname, MIN(e.sal), MAX(e.sal), AVG(e.sal)  
FROM emp e, dept d
```

WHERE e.deptno = d.deptno
GROUP BY d.dname;

Exp 11 SET OPERATORS

1. Create table job_history By Copying the structure data from emp table.

Query:

```
CREATE TABLE job_history AS SELECT * FROM emp;
```

2. Insert three records to emp table.

Query:

```
INSERT INTO emp VALUES (&empno, '&ename', '&job', &mgr, '&hiredate', &sal,  
&comm, &deptno);
```

3. WAQ to display unique records from both table emp and job_history.

Query:

```
SELECT FROM emp UNION SELECT FROM job_history;
```

4. WAQ to Display the employees work for the company from since from beginning to till date from both the table emp and job_history.

Query:

```
SELECT FROM emp INTERSECT SELECT FROM job_history;
```

5. WAQ to display Employees joined recently leaving who has joined since beginning.

Query:

```
SELECT empno, ename, job FROM emp MINUS SELECT empno, ename, job  
FROM job_history;
```

Exp 12: DCL

1. Create the new user called JAINBCAIISEM with the password jain.

Query:

```
CREATE USER JAINNBCAIISEM IDENTIFIED BY jain;
```

2. Change the password jain as BCA for the user JAINBCAIISEM.

Query:

```
ALTER USER JAINNBCAIISEM IDENTIFIED BY jain;
```

3. Provide the permission to the user JAINBCAIISEM for create session, creating table, view.

Query:

```
GRANT CREATE SESSION, CREATE TABLE, CREATE VIEW TO JAINBCAIISEM;
```

4. Take the permission create view from the user JAINBCAIIISEM.

Query:

```
REVOKE CREATE VIEW FROM JAINBCAIIISEM;
```

5. Take back all the DML permission from the user JAINBCAIIISEM.

Query:

```
REVOKE INSERT, UPDATE, DELETE ON emp FROM JAINBCAIIISEM;
```

6. Create the role called HR and provide the privileges create table, create view to HR.

Query:

```
CREATE ROLE HR;
```

```
GRANT CREATE TABLE, CREATE VIEW TO HR;
```

7. Provide the HR role to ram and shyam.

Query:

```
GRANT HR TO ram, shyam;
```

8. Remove the created role HR and remove ram and shyam.

Query:

```
DROP USER ram;
```

```
DROP ROLE HR;
```