**SQL Job Preparation Assignment 2**

**Note: -** For below all queries I used **Live SQL** by oracle as they provide inbuilt **HR Schema** for practice purpose and in my local system, I don’t have HR Schema.

1. In SQL, what are DRL, DDL, DML, TCL, and DCL? Explain each with a single example.

Sol: -

**DRL: - Data Retrieval Language**

It has only 1 command i.e., **SELECT** to retrieve the data from database.

Ex: - SELECT \* FROM EMPLOYEE WHERE LAST\_NAME=’King’;

**DDL: - Data Definition Language**

It has **CREATE, ALTER, RENAME, DROP.** Which are used to define the table, schema, views, stored procedure and so on.

Ex: 🡪 CREATE TABLE T1(ID INT, NAME VARCHAR(10));

🡪 ALTER TABLE T1 MODIFY NAME VARCHAR(20);

🡪 RENAME TABLE T1 TO TABLE1;

🡪 DROP TABLE TABLE1;

**DML: - Data Manipulation Language**

We have **INSERT, UPDATE, DELETE** commands for manipulating the data present in database in form of rows and columns.

🡪 INSERT INTO TABLE1 VALUES(1,`Kishan`);

🡪 UPDATE TABLE1 SET NAME=`KISHAN` where ID=1;

🡪 DELETE FROM TABLE1 WHERE ID=1;

**TCL: - Transaction Control Language**

Here commands will deal with transaction concepts. We have **COMMIT, ROLLBACK, SAVEPOINT.**

🡪 DELETE FROM TABLE1 WHERE ID=1;

COMMIT;

🡪 DELETE FROM TABLE1 WHERE ID=1;

ROLLBACK;

🡪 DELETE FROM TABLE1 WHERE ID=1;

SAVEPOINT SP1;

**DCL: - Data Control Language**

There we deal with permissions and rights. We have **GRANT, REVOKE.**

🡪 GRANT SELECT ON TABLE1 TO USER\_KISHAN;

🡪 REVOKE SELECT ON TABLE1 FROM USER\_KISHAN;

2. For example, explain the criteria for column Alias.

Sol: - We use **‘as’** keywordor some time NO keyword followed by new column name.

* If we join two columns with some operation like +, -, \* then new column will be formed with no proper column name. To overcome the complex column READABILITY we give alias for the new column.

Ex: -

🡪 SELECT CONCAT (FIRST\_NAME, LAST\_N AME) **AS** FULL\_NAME

FROM EMPLOYEES;

🡪 SELECT SALARY+NVL(COMMISSION\_PCT,0) NEW\_COMM

FROM EMPLOYEES;

3. Replace the hardcoded value in the ENAME column with a dynamic value: SELECT EMPNO, ENAME, SAL FROM EMP WHERE ENAME='scott';

Sol: -

DECLARE @SQL NVARCHAR(10000)

DECLARE @ENAME VARCHAR(10)

SET @ENAME=’scott’

SET @SQL=`SELECT EMPNO, ENAME, SAL

FROM EMP WHERE ENAME =`+@ENAME

EXEC(@SQL)

4. Create a query to DISPLAY THE EMPNO, ENAME, WITH A SALARY OVER 3000 AND ENAME IS THE KING.

Sol: - I had used Live SQL provided by oracle.

link:- <https://login.oracle.com/mysso/signon.jsp>.

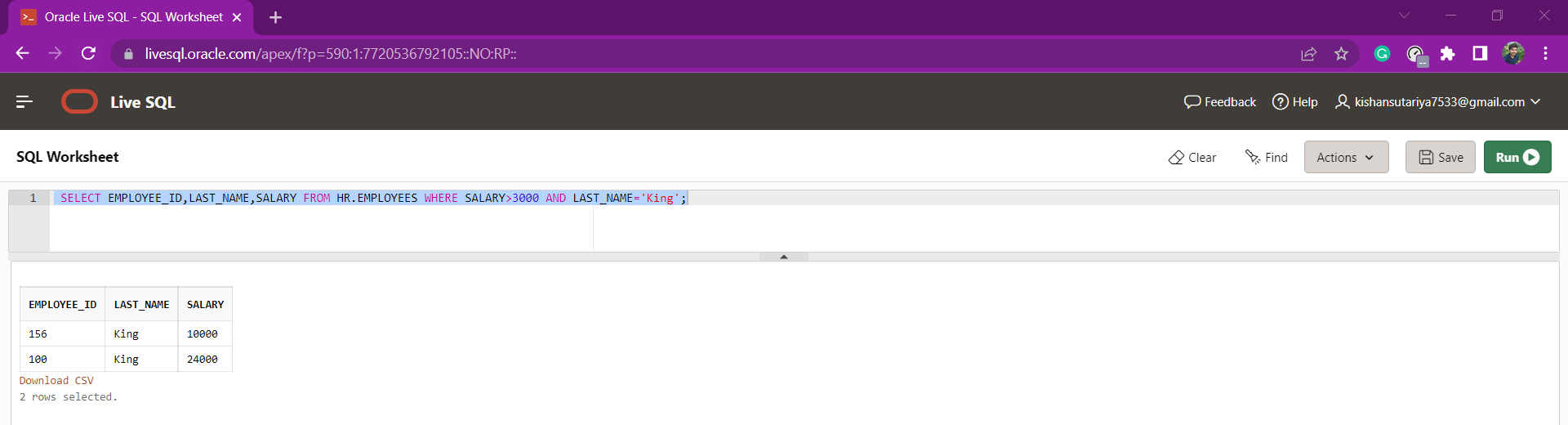
Here there is HR Schema in built so we can do the queries. Below is the query along with result.

QUERY: -

SELECT EMPLOYEE\_ID,LAST\_NAME, SALARY

FROM HR.EMPLOYEES

WHERE SALARY>3000 AND LAST\_NAME=`King`;



5. Create a query to DISPLAY THE EMPNO, ENAME, SAL WITHOUT A SALARY OF $3000

Sol: -

select employee\_id, last\_name, salary

from hr.employees

where salary !=3000;