Name: Sarthak Jain

Programme: B.Tech CSE (Spec. AI&ML)

Course: Advanced Database Management Systems

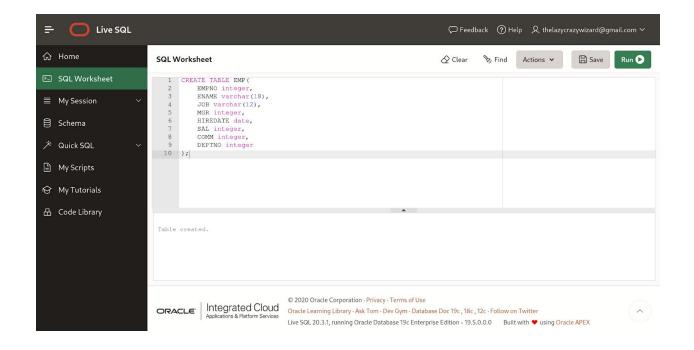
#### LAB 5

Title: Use of different SQL clauses and join

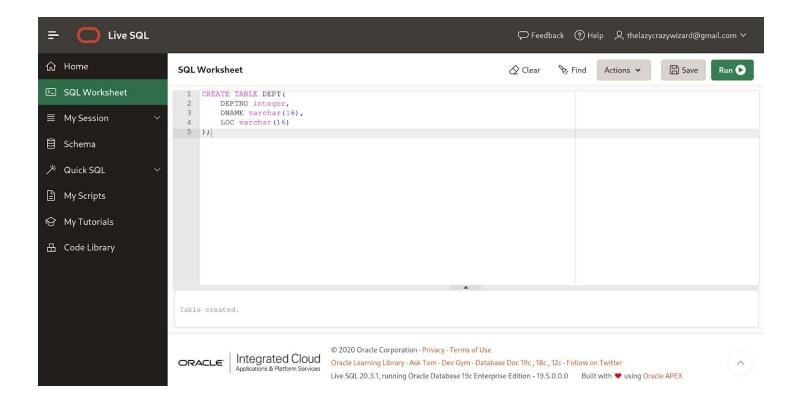
**Objective:** To understand the use of group by and having clause and execute the SQL commands using JOIN

1. Write the SQL Queries for the following queries (use emp\_table and dept\_table of Experiment 4).

```
CREATE TABLE EMP(
EMPNO integer,
ENAME varchar(18),
JOB varchar(12),
MGR integer,
HIREDATE date,
SAL integer,
COMM integer,
DEPTNO integer
```



```
CREATE TABLE DEPT(
DEPTNO integer,
DNAME varchar(16),
LOC varchar(16)
);
```



#### \* Inserting values \*

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7369, 'SMITH', 'CLERK', 7902, '17-DEC-80', 500, 800, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7499, 'ALLEN', 'SALESMAN', 7698, '20-FEB-81', 1600, 300, 30);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7521, 'WARD', 'SALESMAN', 7698, '22-FEB-81', 1250, 500, 30);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7566, 'JONES', 'MANAGER', 7839, '02-APR-81', 2975, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7654, 'MARTIN', 'SALESMAN', 7698, '28-SEP-81', 1250, 1400, 30);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7698, 'BLAKE', 'MANAGER', 7839, '01-MAY-81', 2850, 30);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7782, 'CLARK', 'MANAGER', 7839, '09-JUN-81', 2450, 10);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7788, 'SCOTT', 'ANALYST', 7566, '09-DEC-82', 3000, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, SAL, DEPTNO) VALUES (7839, 'KING', 'PRESIDENT', '17-NOV-81', 5000, 10);

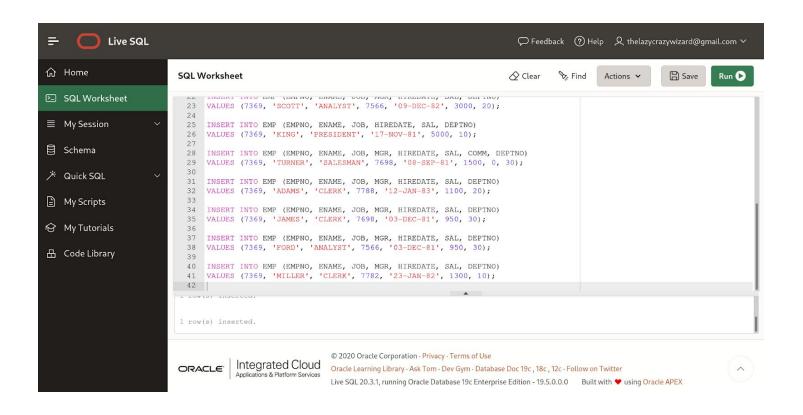
INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7844, 'TURNER', 'SALESMAN', 7698, '08-SEP-81', 1500, 0, 30);

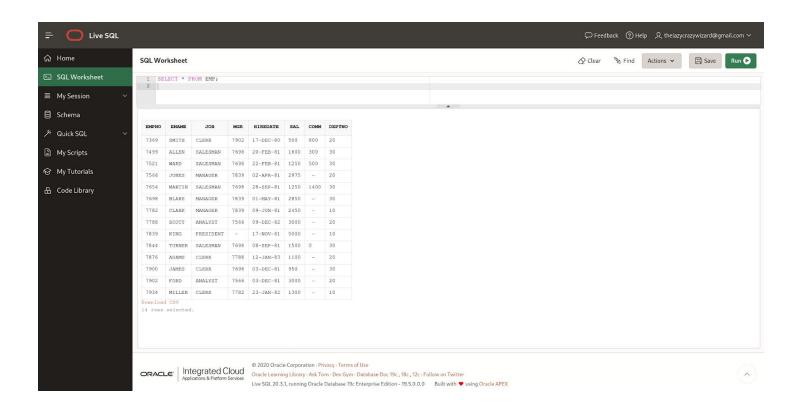
INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7876, 'ADAMS', 'CLERK', 7788, '12-JAN-83', 1100, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7900, 'JAMES', 'CLERK', 7698, '03-DEC-81', 950, 30);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7902, 'FORD', 'ANALYST', 7566, '03-DEC-81', 3000, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, DEPTNO) VALUES (7934, 'MILLER', 'CLERK', 7782, '23-JAN-82', 1300, 10);



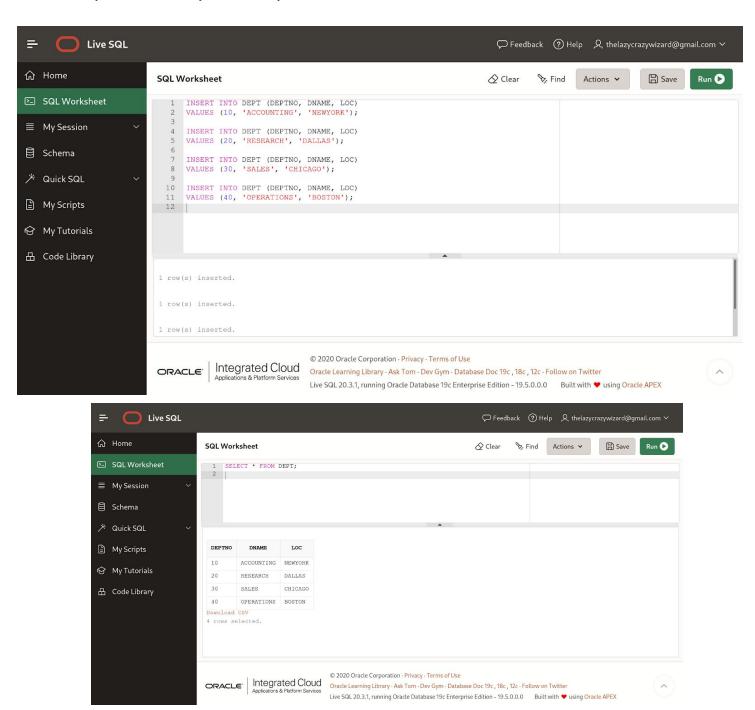


INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (10, 'ACCOUNTING', 'NEWYORK');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (20, 'RESEARCH', 'DALLAS');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (30, 'SALES', 'CHICAGO');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (40, 'OPERATIONS', 'BOSTON');



#### 1.1. List the Deptno where there are no emps.

#### Answer 1.1:

```
SELECT * FROM (

SELECT DEPT.DNAME, COUNT(EMP.DEPTNO) C

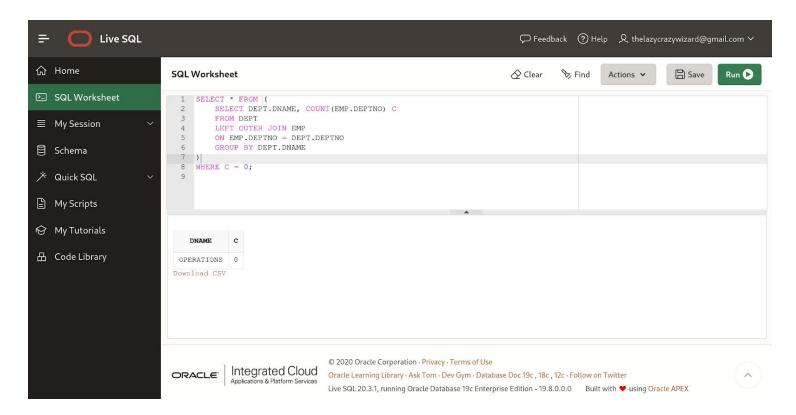
FROM DEPT

LEFT OUTER JOIN EMP

ON EMP.DEPTNO = DEPT.DEPTNO

GROUP BY DEPT.DNAME
)

WHERE C = 0;
```



### 1.2. List the No.of emp's and Avg salary within each department for each job. Answer 1.2:

```
SELECT D, J, COUNT(J), AVG(S) FROM (

SELECT DEPT.DNAME D, EMP.JOB J, EMP.SAL S

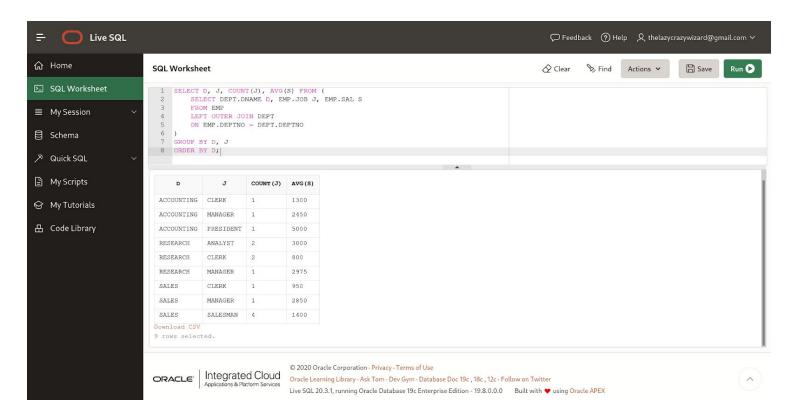
FROM EMP

LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO
)

GROUP BY D, J

ORDER BY D;
```



### 1.3. Find the maximum average salary drawn for each job except for 'President'. Answer 1.3:

```
SELECT D, MAX(S) FROM (

SELECT DEPT.DNAME D, EMP.JOB J, AVG(EMP.SAL) S

FROM EMP

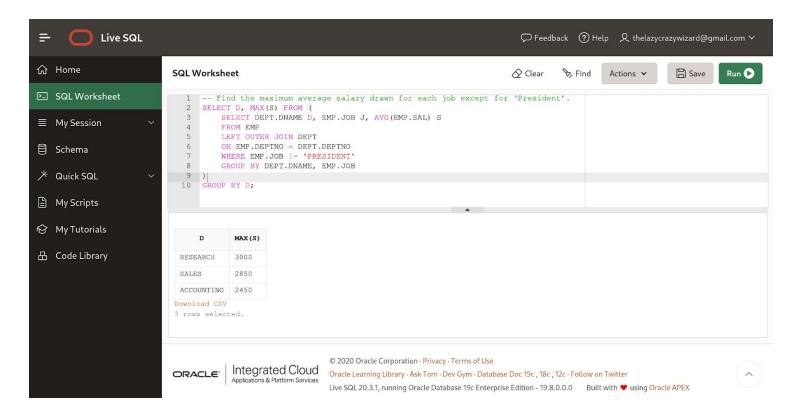
LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO

WHERE EMP.JOB != 'PRESIDENT'

GROUP BY DEPT.DNAME, EMP.JOB
)

GROUP BY D;
```



### 1.4. List the department details where at least two emps are working. Answer 1.4:

```
SELECT DEPT.*, S Employees FROM (

SELECT DEPT.DNAME D, COUNT(*) S

FROM EMP

LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO

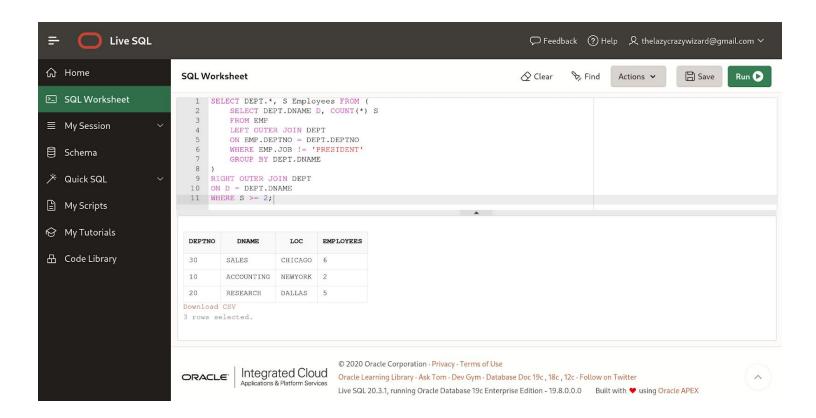
WHERE EMP.JOB != 'PRESIDENT'

GROUP BY DEPT.DNAME
)

RIGHT OUTER JOIN DEPT

ON D = DEPT.DNAME

WHERE S >= 2;
```



### 1.5. List the no. of emps in each department where the no. is more than 3. Answer 1.5:

```
SELECT D Department, S Employees FROM (

SELECT DEPT.DNAME D, COUNT(*) S

FROM EMP

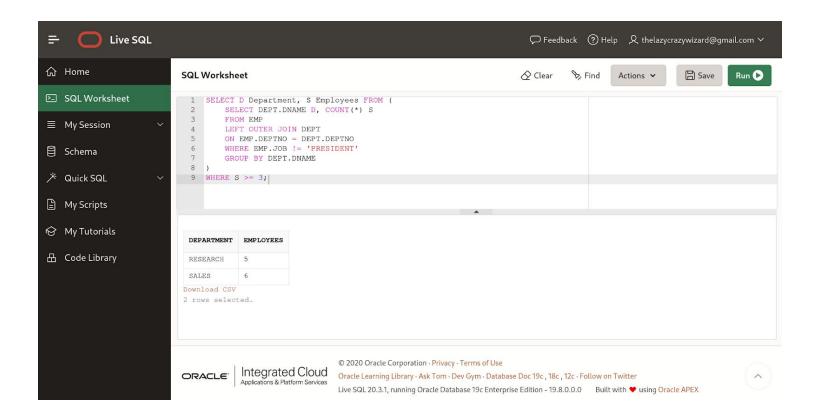
LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO

WHERE EMP.JOB != 'PRESIDENT'

GROUP BY DEPT.DNAME
)

WHERE S >= 3;
```



### 1.6. List the names of the emps who are getting the highest sal dept wise. Answer 1.6:

```
SELECT EMP.ENAME, D DEPARTMENT, SALARY FROM (

SELECT DEPT.DNAME D, MAX(EMP.SAL) SALARY

FROM EMP

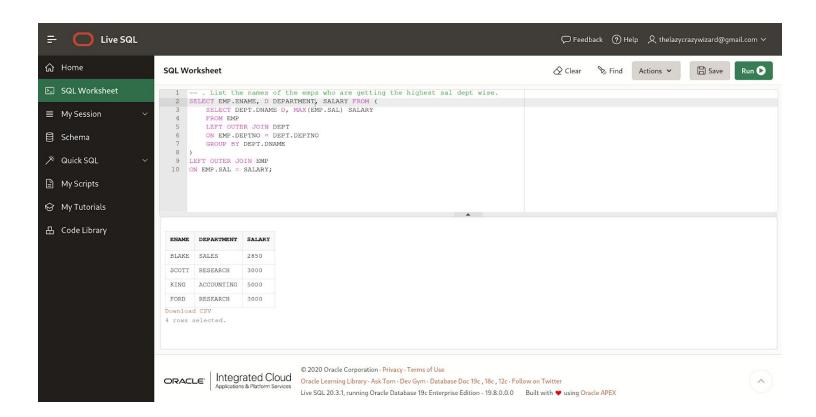
LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO

GROUP BY DEPT.DNAME
)

LEFT OUTER JOIN EMP

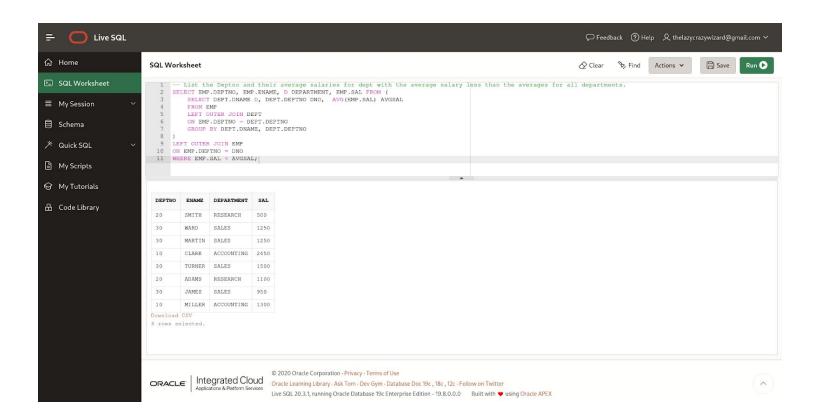
ON EMP.SAL = SALARY;
```



# 1.7. List the Deptno and their average salaries for dept with the average salary less than the averages for all departments.

Answer 1.7:

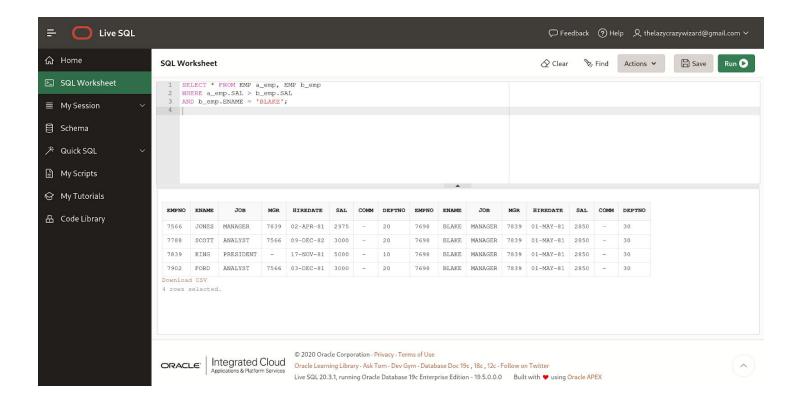
```
SELECT EMP.DEPTNO, EMP.ENAME, D DEPARTMENT, EMP.SAL FROM (
SELECT DEPT.DNAME D, DEPT.DEPTNO DNO, AVG(EMP.SAL) AVGSAL
FROM EMP
LEFT OUTER JOIN DEPT
ON EMP.DEPTNO = DEPT.DEPTNO
GROUP BY DEPT.DNAME, DEPT.DEPTNO
)
LEFT OUTER JOIN EMP
ON EMP.DEPTNO = DNO
WHERE EMP.SAL < AVGSAL;
```



#### Question 2. Execute the experiment 4 using sql join.

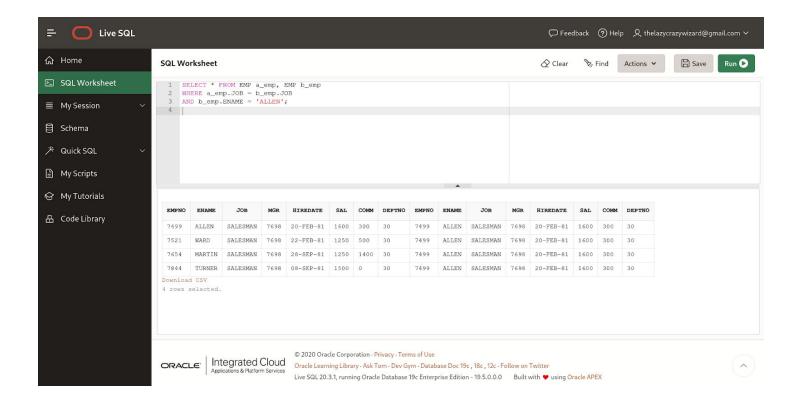
### 2.1. List the details of the emps whose Salaries more than the employee BLAKE. Answer 2.1:

SELECT \* FROM EMP a\_emp, EMP b\_emp WHERE a\_emp.SAL > b\_emp.SAL AND b\_emp.ENAME = 'BLAKE';



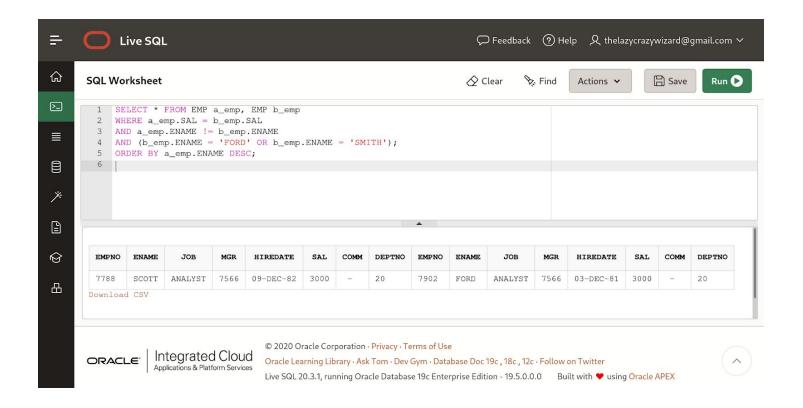
### 2.2. List the emps whose Jobs are the same as ALLEN. Answer 2.2:

SELECT \* FROM EMP a\_emp, EMP b\_emp WHERE a\_emp.JOB = b\_emp.JOB AND b\_emp.ENAME = 'ALLEN';



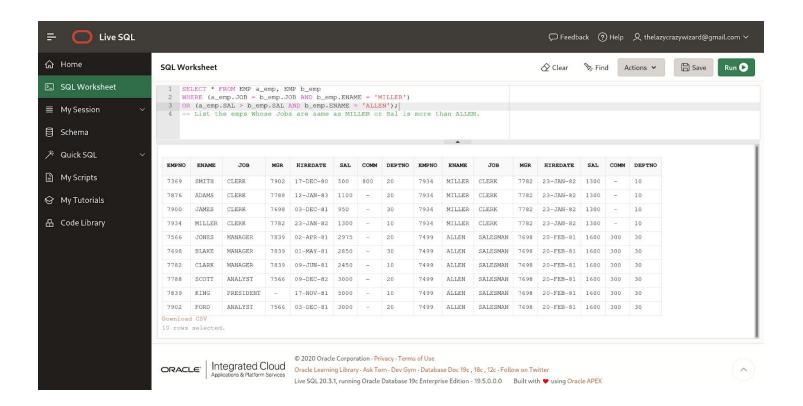
#### 2.3. List the Emps whose Sal is the same as FORD or SMITH in desc order of Names. Answer 2.3:

```
SELECT * FROM EMP a_emp, EMP b_emp
WHERE a_emp.SAL = b_emp.SAL
AND a_emp.ENAME != b_emp.ENAME
AND (b_emp.ENAME = 'FORD' OR b_emp.ENAME = 'SMITH');
ORDER BY a_emp.ENAME DESC;
```



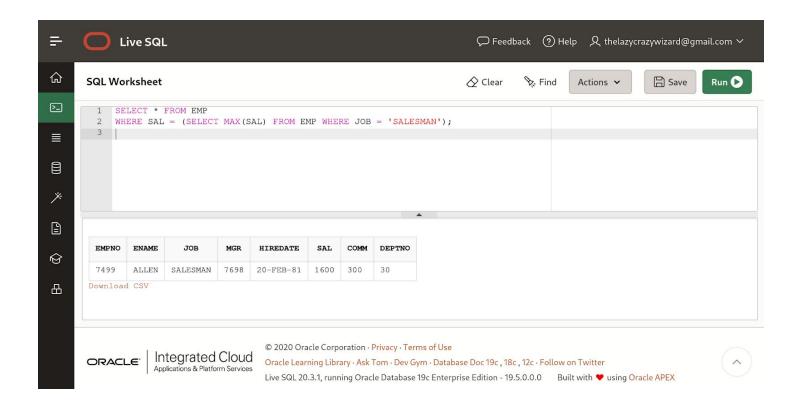
#### 2.4. List the emps Whose Jobs are the same as MILLER or Sal is more than ALLEN. Answer 2.4:

SELECT \* FROM EMP a\_emp, EMP b\_emp
WHERE (a\_emp.JOB = b\_emp.JOB AND b\_emp.ENAME = 'MILLER')
OR (a\_emp.SAL > b\_emp.SAL AND b\_emp.ENAME = 'ALLEN');



## 2.5. Find the highest paid employee of the sales department. Answer 2.5:

SELECT \* FROM EMP
WHERE SAL = (SELECT MAX(SAL) FROM EMP WHERE JOB = 'SALESMAN');



# 2.6. List the employees who are senior to the most recently hired employee working under King.

#### Answer 2.6:

- -- Everyone works under King since he is the president
- -- Seniors will have more salary then juniors\

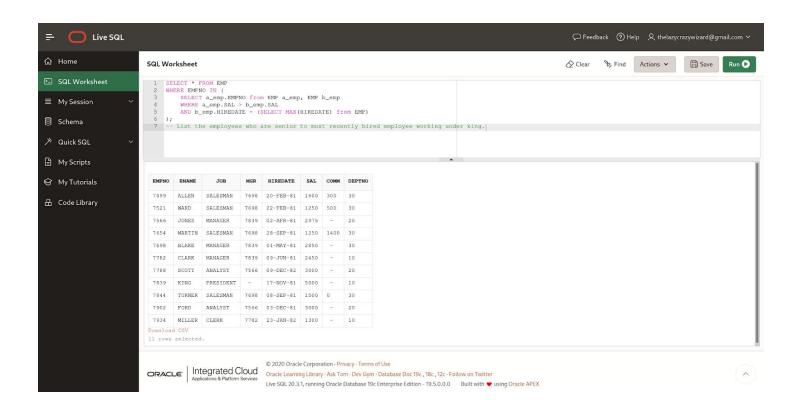
```
SELECT * FROM EMP

WHERE EMPNO IN (

SELECT a_emp.EMPNO from EMP a_emp, EMP b_emp

WHERE a_emp.SAL > b_emp.SAL

AND b_emp.HIREDATE = (SELECT MAX(HIREDATE) from EMP)
);
```



### 2.7. List the names of the emps who are getting the highest sal dept wise. Answer 2.7:

```
SELECT EMP.ENAME, D DEPARTMENT, SALARY FROM (

SELECT DEPT.DNAME D, MAX(EMP.SAL) SALARY

FROM EMP

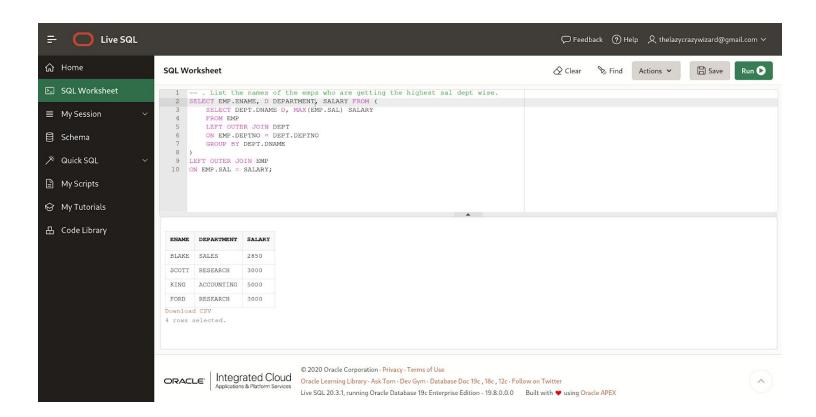
LEFT OUTER JOIN DEPT

ON EMP.DEPTNO = DEPT.DEPTNO

GROUP BY DEPT.DNAME
)

LEFT OUTER JOIN EMP

ON EMP.SAL = SALARY;
```



### 2.8. List the emps whose sal is equal to the average of max and minimum Answer 2.8:

```
SELECT * FROM EMP

WHERE SAL = (

SELECT AVG(SAL) FROM emp

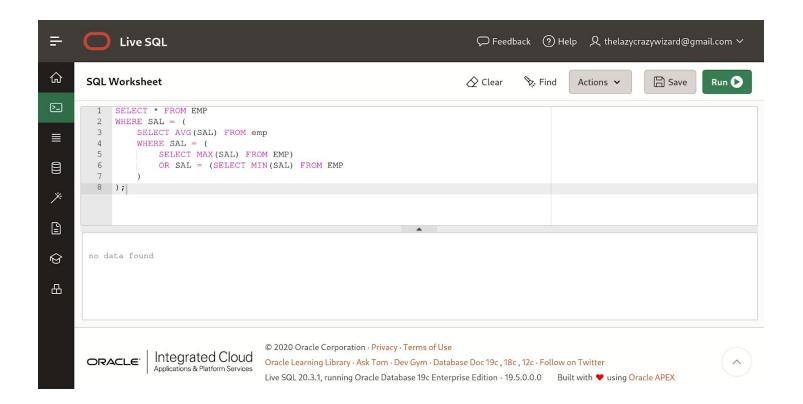
WHERE SAL = (

SELECT MAX(SAL) FROM EMP)

OR SAL = (SELECT MIN(SAL) FROM EMP

)

);
```



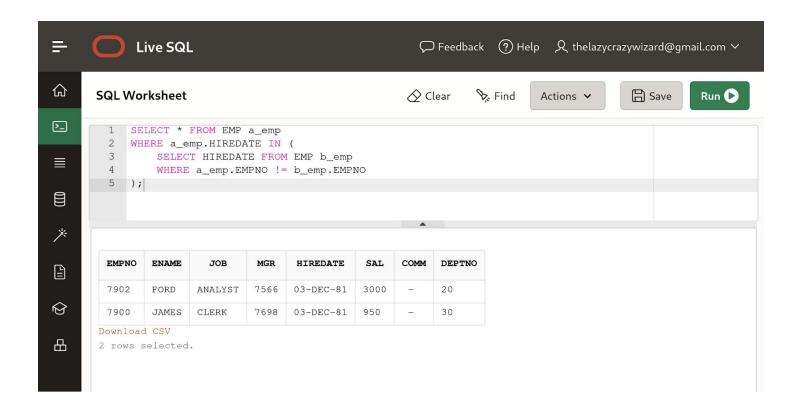
## 2.9. List the emps who joined the company on the same date. Answer 2.9:

```
SELECT * FROM EMP a_emp

WHERE a_emp.HIREDATE IN (

SELECT HIREDATE FROM EMP b_emp

WHERE a_emp.EMPNO != b_emp.EMPNO
);
```



### 2.10. Find out the emps who joined in the company before their Managers. Answer 2.10:

SELECT a\_emp.\* FROM EMP a\_emp, EMP b\_emp WHERE a\_emp.HIREDATE < b\_emp.HIREDATE AND a\_emp.MGR = b\_emp.EMPNO;

