# Lab Assignment​

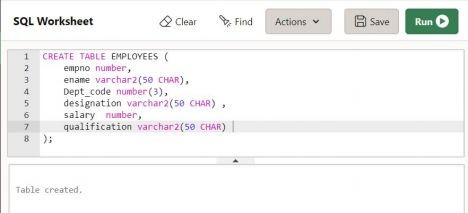
**AIM**:​ Write the queries for DataManipulation and Data Definition Language.

**Theory:**

**DML:** A data manipulation language (DML) is a family of syntax elements​ similar to a computer programming language used for selecting, inserting, deleting and updating data in a database. Performing read-only queries of data is sometimes also considered a component of DML. Commands in DML are:

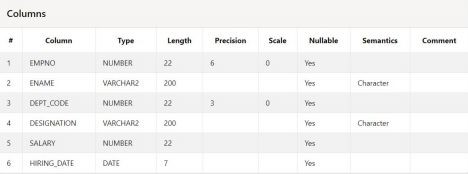
1. INSERT
2. UPDATE
3. DELETE

**Q 1:** Write a query to create a table employee with empno, ename,​ Dept code (fixed size 3) , designation, and salary, Qualification.



**Q 2**:​ Write a command to see the structure of the table.

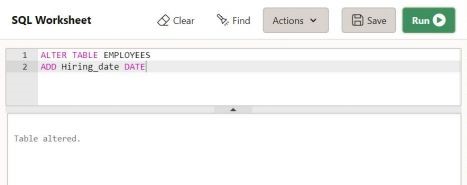
DESCRIBE EMPLOYEES​



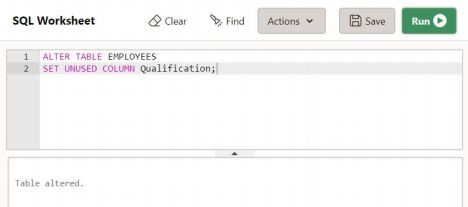
**Q 3:** Write a Query to Alter the column EMPNO NUMBER(4) TO​ EMPNO NUMBER(6).



**Q 4:** Write a query to add a new column Hiring\_date in to employee​



**Q 5:** Delete COLUMN Qualification from the table​

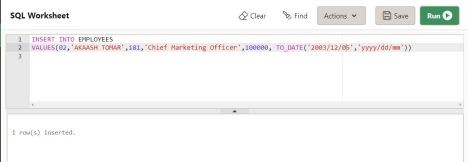


Q 6: Write a query to drop an existing table employee

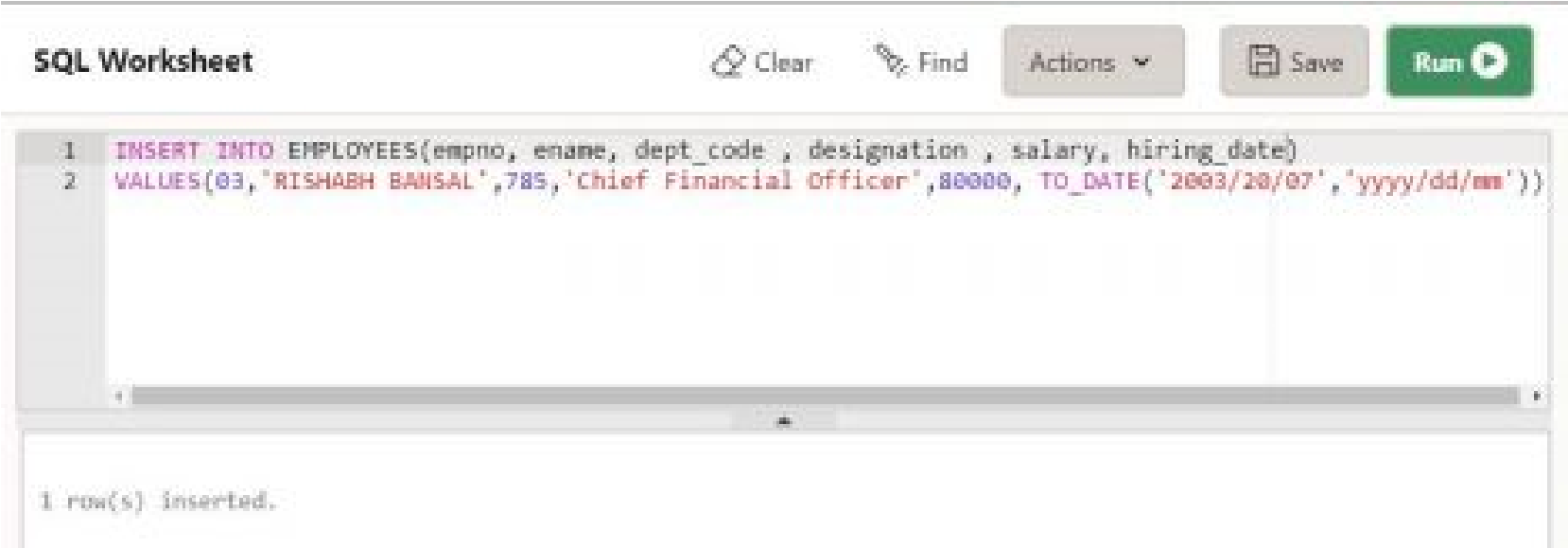
DROP TABLE EMPLOYEE​

**Q 7:** Write a query to insert the records in to employee at least five​ rows with different methods

**Method 1**

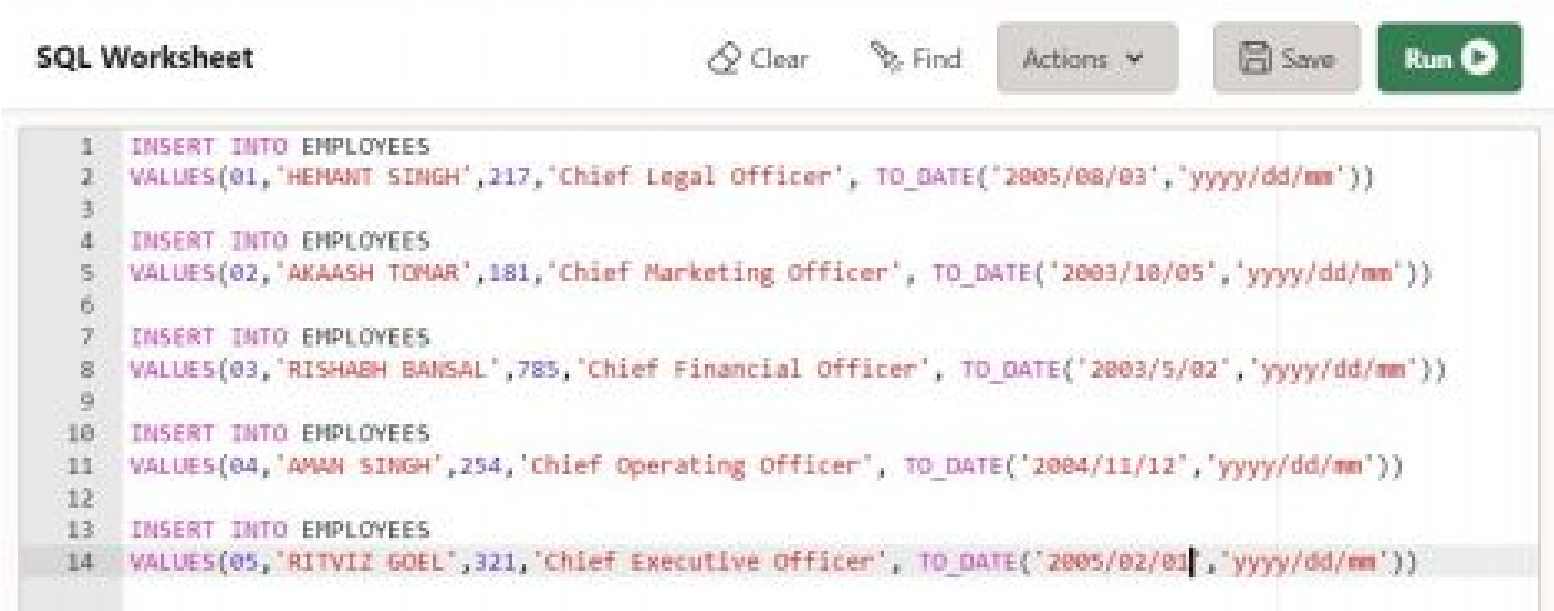


**Method 2**



**Method 3**

Insert by SELECT ​ statement​

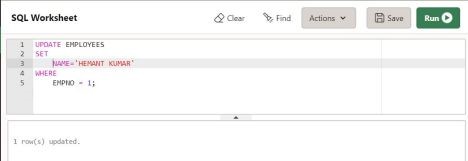


**Q 8**:​ Write a query to display the records from employee

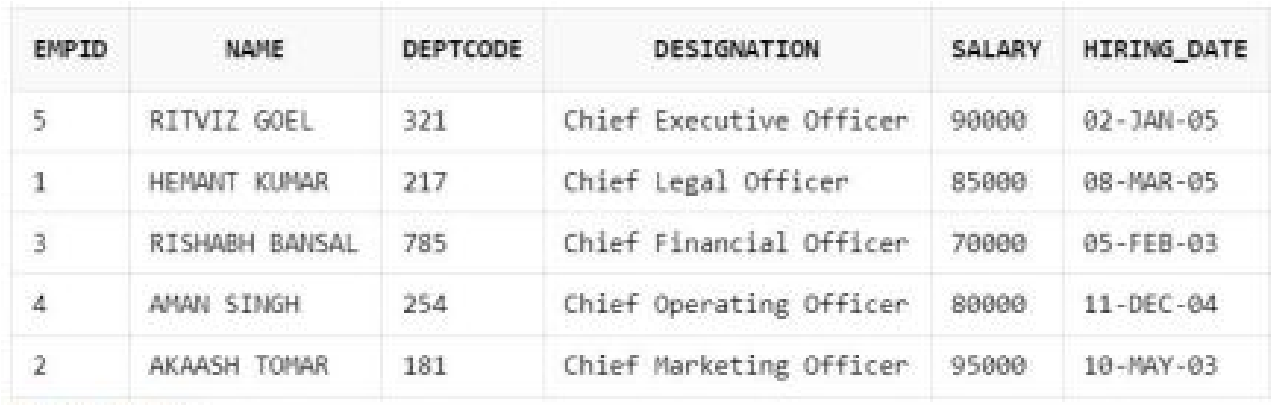
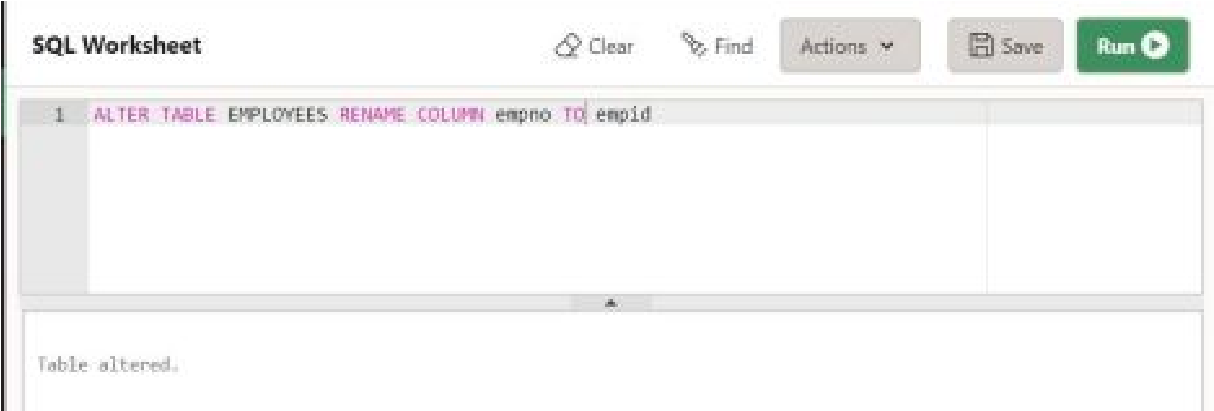


**Q 9**:​ Write a query to update the records from HEMANT SINGH changed to

HEMANT KUMAR



**Q 10**:​ Write a query to change the name of the column name empno to empid



SQL Assignment 2

**The SQL Select Statement**

**Aim**:​ To learn operators

**Theory** :​ The ​ ​**SELECT** ​statement is used to select data from a database. The data returned is stored in a result table, called the result-set.

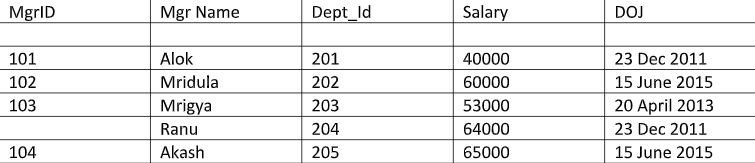
Syntax:

**SELECT** ​column\_list FROM table-name

**[WHERE Clause] [GROUP BY clause]**

**[HAVING clause] [ORDER BY clause];**

Data Table

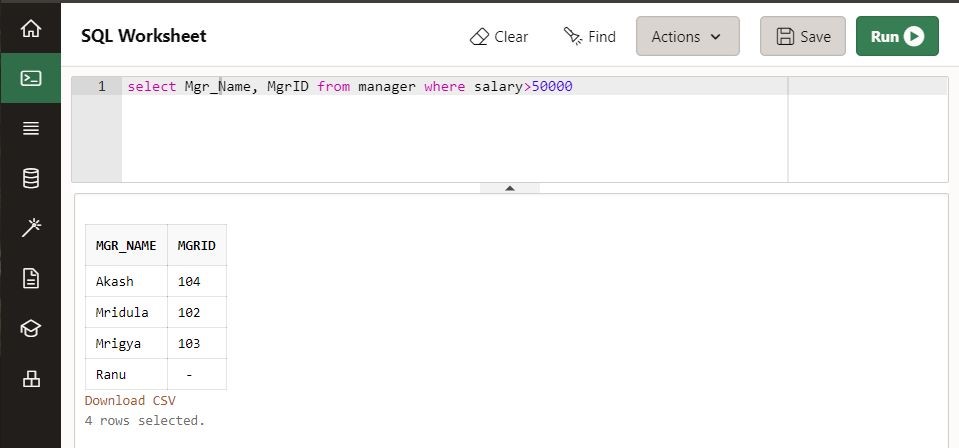


**Q 1**​ :​Write query to retrieve mgr name , mgr id from manager table whose salary is

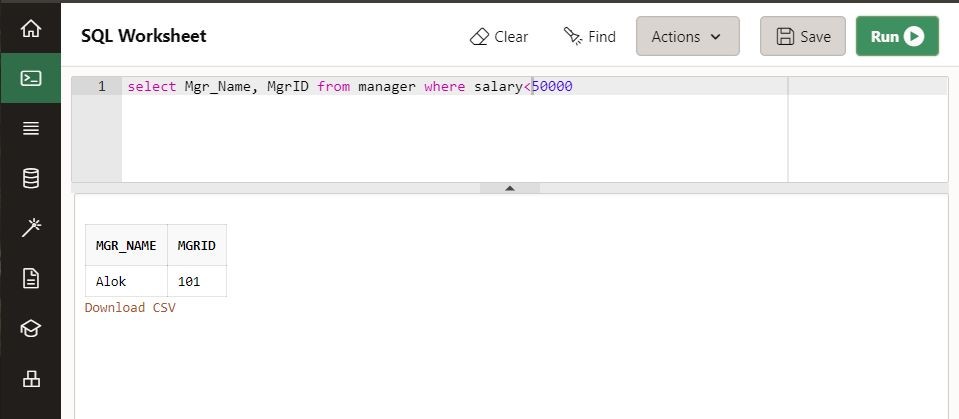
1. Greater than 50000
2. Less than 50000
3. Greater and equal to 64000
4. Less than equal to 53000

Ans

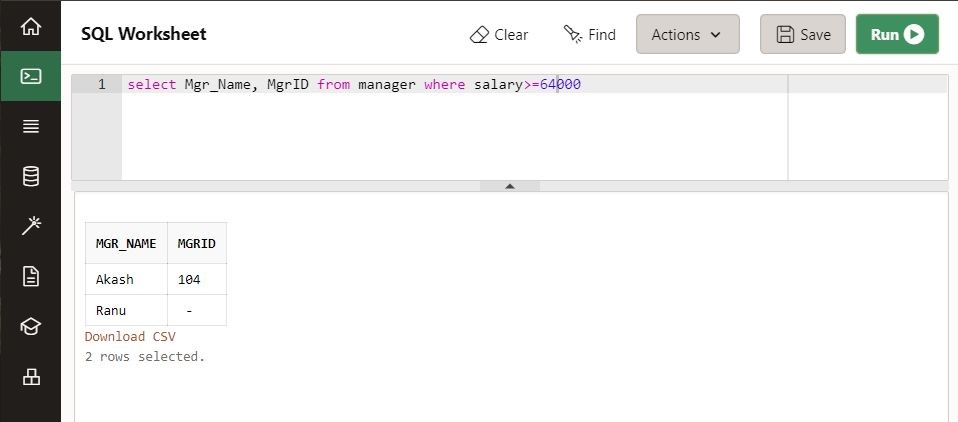
1. ​Greater than 50000



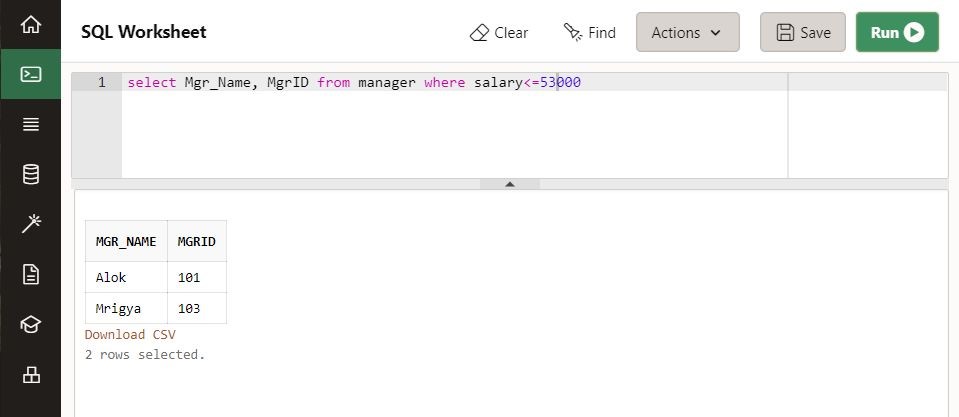
1. Less than 50000​



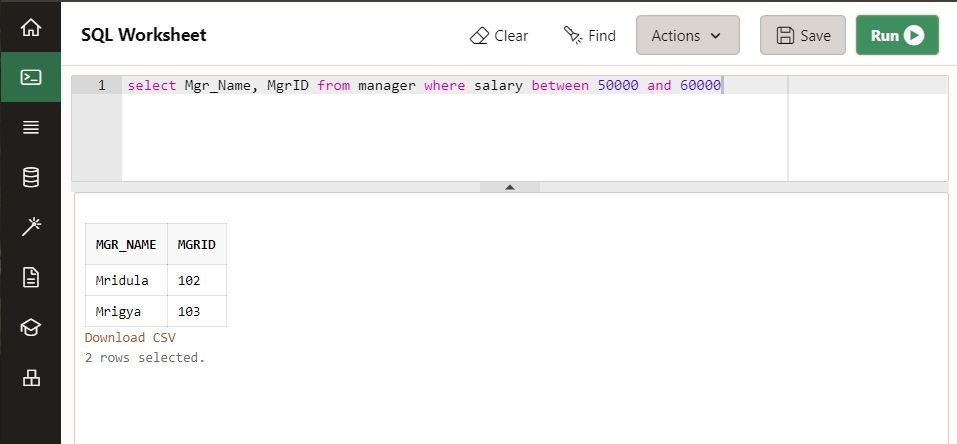
1. Greater and equal to 64000



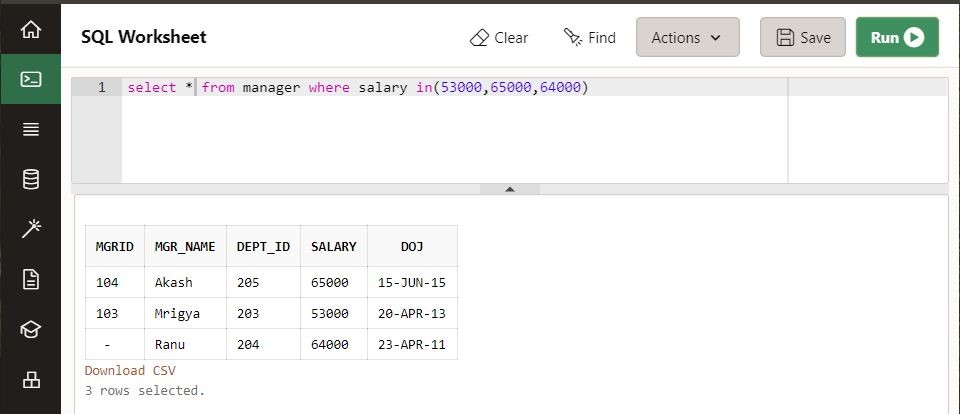
1. Less than equal to 53000



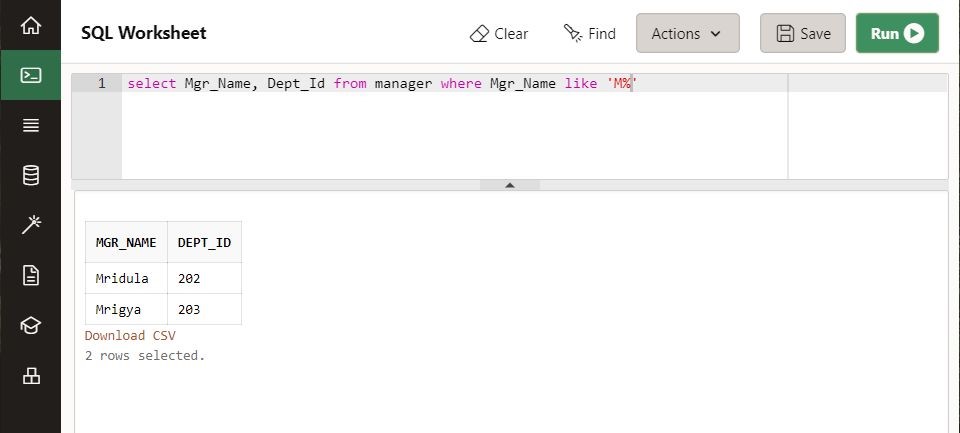
**Q 2**​ :Write query to retrieve mgr name , mgr id from manager table whose salary is between 50000 and 60000



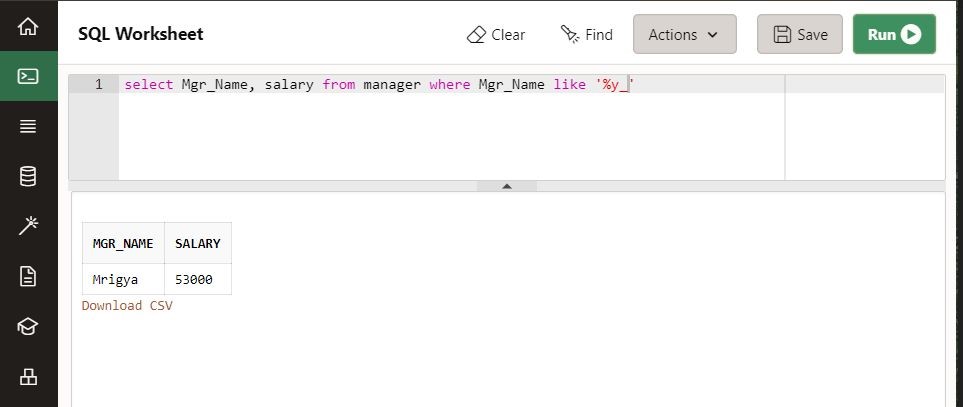
**Q 3**​ : Write query to retrieve all details from manager table whose salary is 53000, 65000, 64000



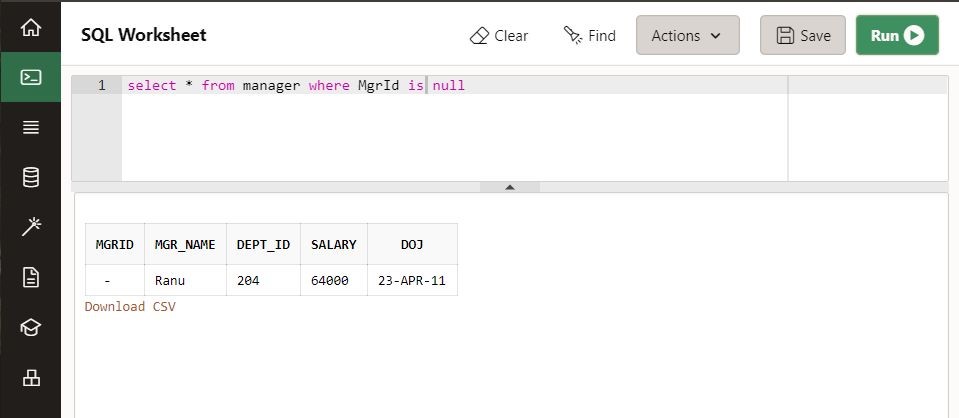
**Q 4:** ​Write query to retrieve mgr name , dept id from manager table whose name starts with capital M



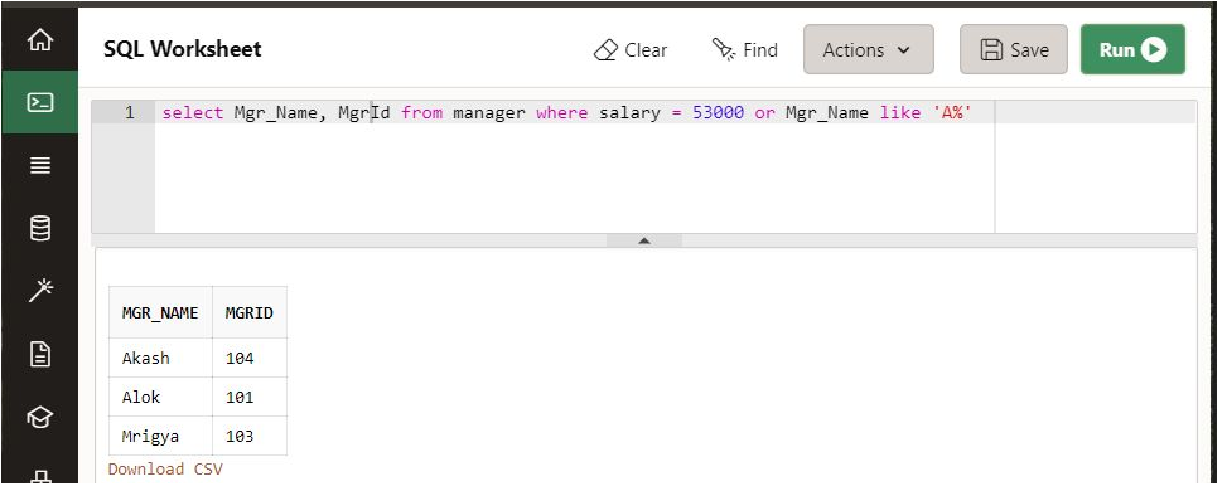
**Q 5:** ​Write query to retrieve mgr name , salary from manager table whose name has y at second last place



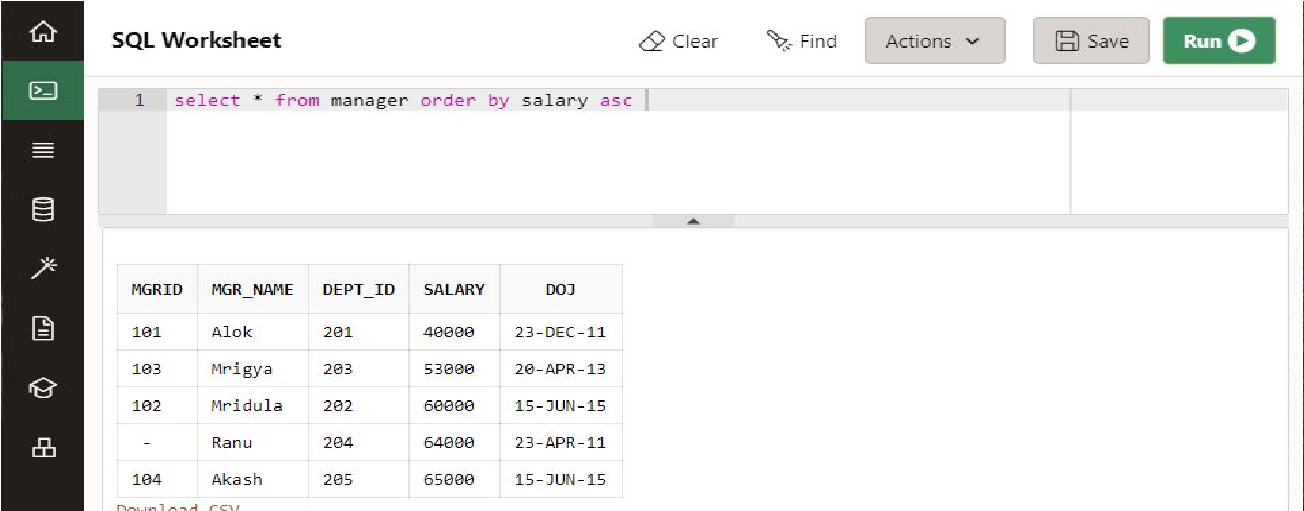
**Q 6:** ​Write a query to retrieve all details from manager table whose mgrid is not known.​



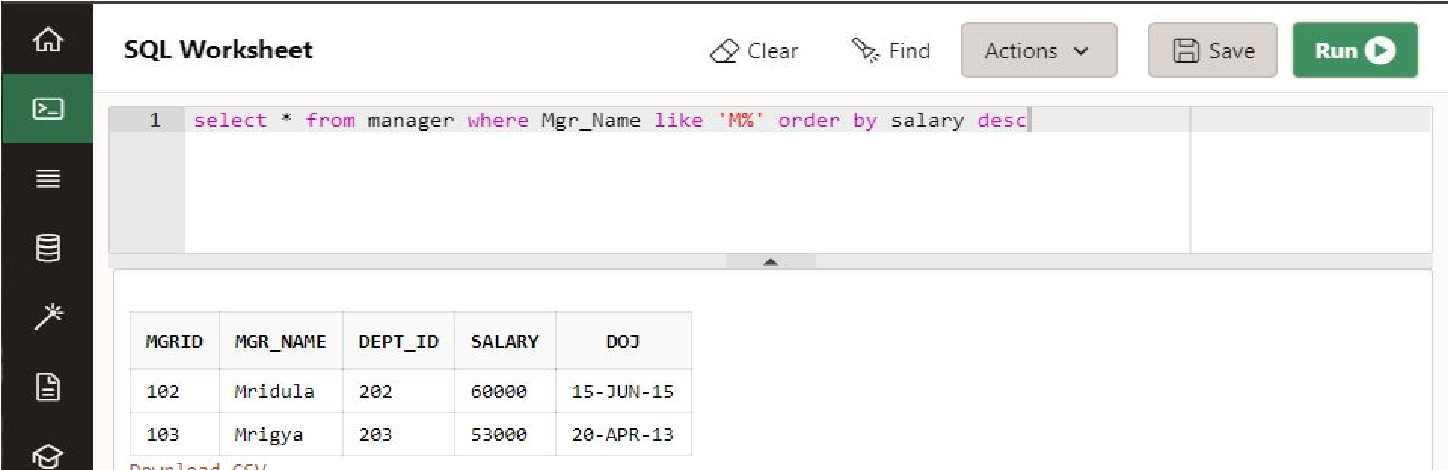
**Q 7:** ​Write query to retrieve mgr name , mgr id from manager table whose salary is 53000 or name starts with A



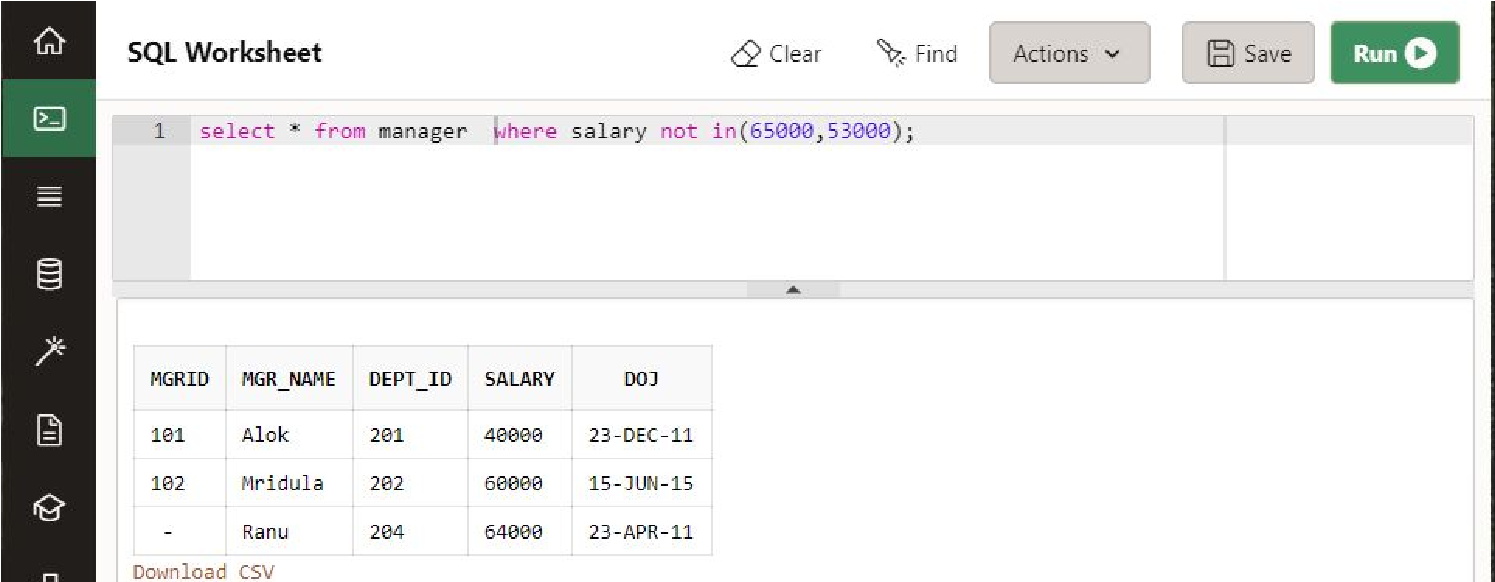
**Q 8:** ​Write query to retrieve all details from manager table and sort the data is ascending order of their salary



**Q 9:**​Write query to retrieve all details from manager table and sort the data is descending order of their salary and having M as start letter of their name



**Q 10:**​Write query to retrieve all details from manager table whose salary not 65000 and 53000

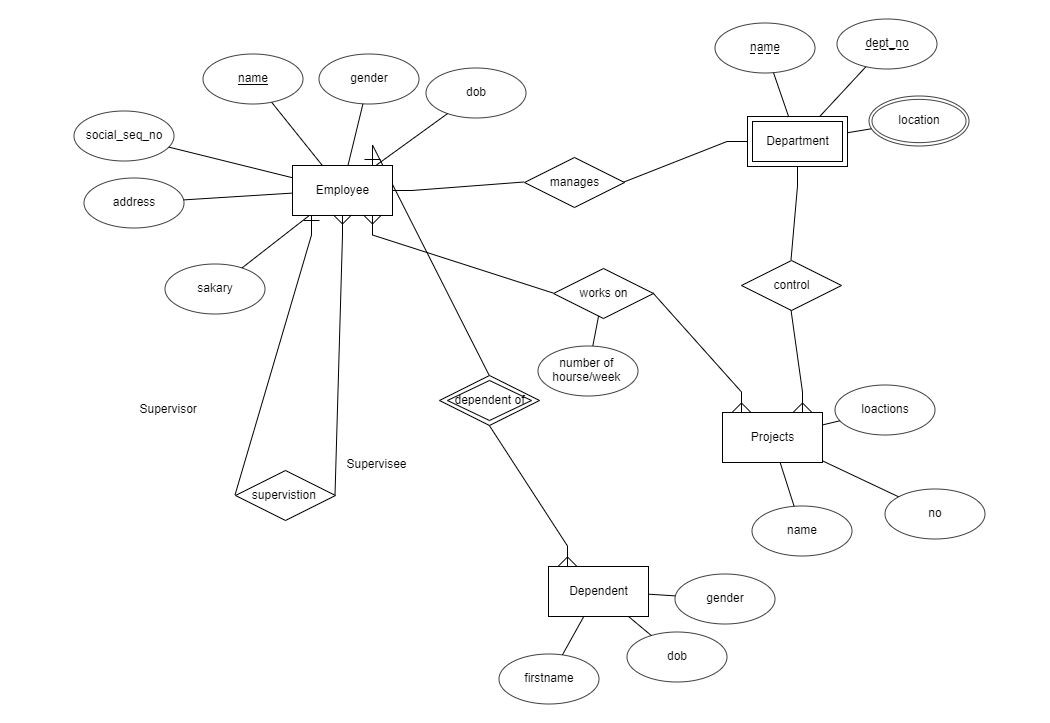


# 

**Assignment 3**

The company is organized into departments. Each department has a unique name, a unique number, and a particular employee who manages the department. We keep track of the start date when that employee began managing the department. A department may have several locations. A department controls a number of projects, each of which has a unique name, a unique number, and a single location. We store each employee’s name, Social Security number, 2 address, salary, sex (gender), and birth date. An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department. We keep track of the current number of hours per week that an employee works on each project. We also keep track of the direct supervisor of each employee (who is another employee).We want to keep track of the dependents of each employee for insurance purposes. We keep each dependent’s first name, sex, birth date, and relationship to the employee.

**Draw ER diagram of above case study using ERD Plus tool.** Answer



**Assignment 4**

**Aim: To implement constraints/restrictions on the Relations**

**Employee**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Empid** | **Emp\_Name** | **location** | **Salary** | **Deptid** | **Doj** |
| **1** | **Akash** | **Del** | **5500** | **1** | **12-Jan-2020** |
| **2** | **Rony** | **Noi** | **20000** | **1** | **18-Aug-201**  **8** |
| **3** | **Ronald** | **Del** | **10000** | **2** | **27-Dec-2009** |
| **4** | **Yukti** | **Mum** | **5000** | **3** | **16-sep-2019** |
| **5** | **Aman** | **Mum** | **16000** |  | **12-May-200**  **6** |

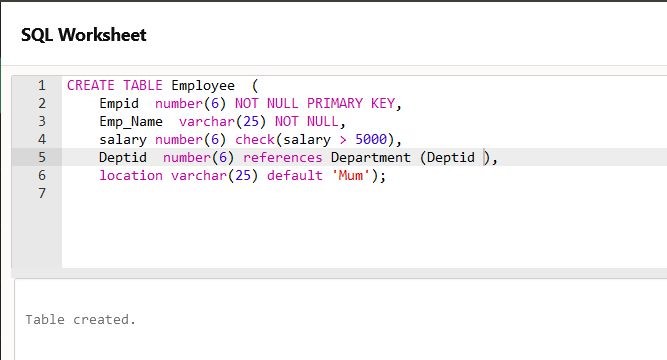
**Department**

|  |  |  |
| --- | --- | --- |
| **Deptid** | **Dname** | **Loc** |
| **1** | **Sales** | **Del** |
| **2** | **Marketing** | **Del** |
| **3** | **HR** | **Mum** |
| **4** | **Product** | **Noi** |

**Ques Create :**

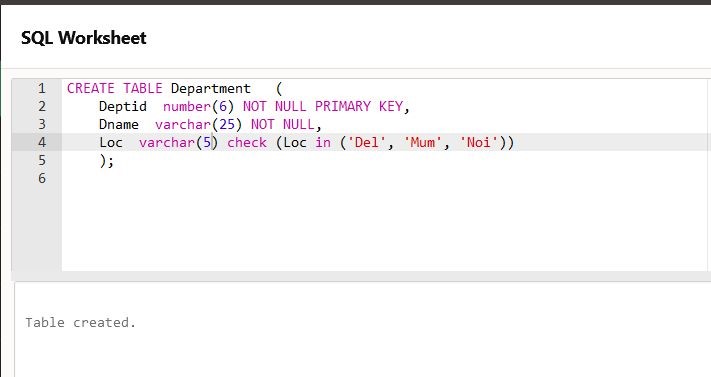
**Employee Table**

* **Empid** ​ ​ **Primary Key**
* **Emp\_Name** ​ ​ **Not null**
* **Salary not less than 5000**
* **Deptid (Employee)** ​ ​ **Foreign Key**
* **Location** ​ ​ **Default Mum**



**Department Table**

* **Deptid** ​ **Primary Key**​
* **Dname** ​ **Not null**​
* **Loc** ​ **only valid values are (Del, Mum, Noi)**​



**Assignment 5**

**Sailor (sid , sname, rating ,Head ,age)**

**Boat (bid , bname, bcolor)**

**Reserve (sid, bid , day)**

## Sailor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sid | Sname | Rating | Head | Age |
| 22 | Ram | 5 | 28 | 30 |
| 29 | Shyam | 7 | 21 | 28 |
| 26 | Sagar | 1 | 28 | 39 |
| 28 | Mohan | 3 | Null | 31 |
| 21 | Ram | 6 | null | 30 |

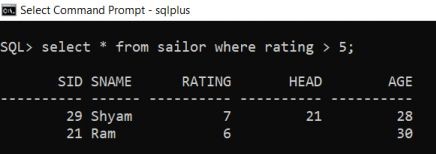
Boat

|  |  |  |
| --- | --- | --- |
| Bid | Bname | Colour |
| 101 | Interlake | Blue |
| 102 | Clipper | Red |
| 103 | Marine | Green |
| 104 | Zubac | Red |

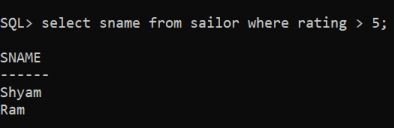
Reserve

|  |  |  |
| --- | --- | --- |
| Sid | Bid | Day |
| 22 | 103 | 1 Feb 2019 |
| 26 | 104 | 2 Aug. 2019 |
| 28 | 102 | 15 Jun 2019 |
| 22 | 104 | 12Aug 2019 |

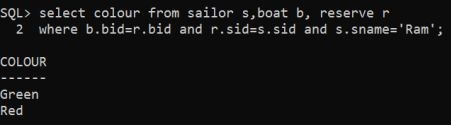
Q1 Find the details of sailor having rating greater than 5.



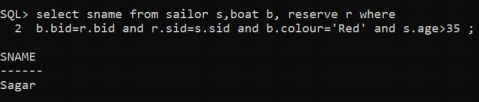
Q2 Find the name of sailor having rating greater than 5.



Q3 Find the colour of boat reserved by ‘RAM’.



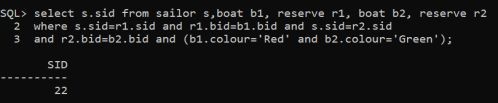
Q4 Find the name of the sailor who have reserved red boat and his age is greater than 35.



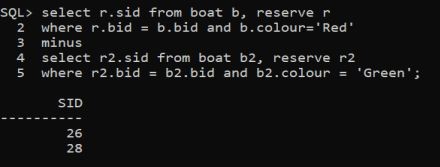
Q5 Find the sid, who have reserved red or green boat



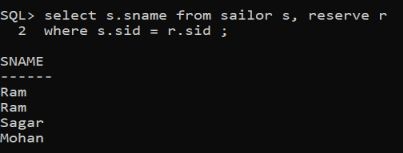
Q6 Find the sid who have reserved red and green boat



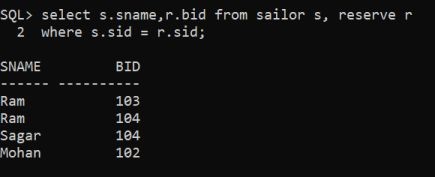
Q7 Find the sid who have reserved red but not green boat



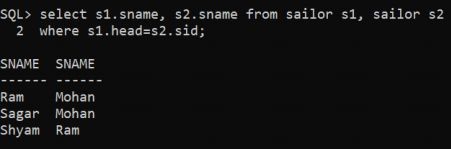
Q8. Find the name of the sailors who have reserved at least one boat.



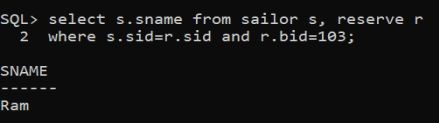
Q9. Find the name of the sailor, along with the boat id, if he has reserved.



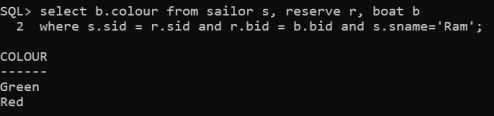
Q10. Display the name of sailor along with his head sailor’s name.



Q11. Find names of sailors who’ve reserved boat #103



Q12. Find the colours of boats reserved by Ram



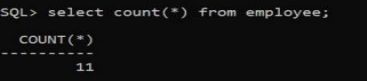
**Assignment Lab 6**

Aggregate Functions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Eno | Ename | Job | Salary | Depno | Mgr |
| 7369 | Smith | Clerk | 800 | 20 | 7782 |
| 7364 | Alen | Sales | 950 | 30 | 7782 |
| 7563 | Ward | Sales | 1100 | 30 | 7563 |
| 7566 | Jones | Manager | 1250 | 20 | 7369 |
| 7939 | Watson | Sales | 1250 | 30 | 7364 |
| 7940 | David | Manager | 1300 | 30 | 7536 |
| 7788 | Rashid | Analyst | 2750 | 10 | 7785 |
| 7876 | Mickel | Sales | 2850 | 20 | 7781 |
| 7956 | Turner | Clerk | 3000 | 10 | 7445 |
| 7800 | Bhanu | Manager | 1500 | 20 | 5656 |
| 7650 | Bhasker | HR | 2500 | 40 | 5645 |

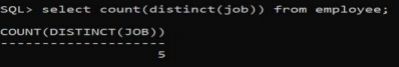
Q1 Calculate the number of employee working in the company

**SQL>** ​select count(\*) from employee;



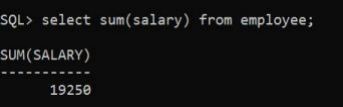
Q2 Calculate the total number of unique jobs in the company

**SQL>**​select count(distinct(job)) from employee;



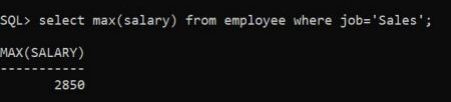
Q3 Calculate the total amount of salary paid to the employees.

**SQL>**select sum(salary) from employee;​



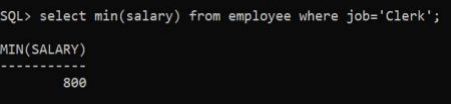
Q4 Find out the maximum salary paid to the employee of Sales Job

**SQL>**select max(salary) from employee where job = ‘Sales’;​



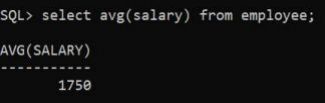
Q5 Find out the minimum salary paid to the employee of Clerk Job

**SQL>**select min(salary) from employee where job = ‘Clerk’;​



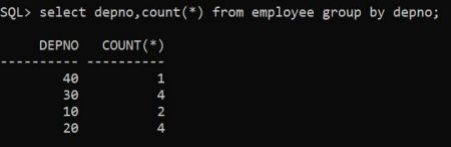
Q6 Calculate the average salary paid in the organization

**SQL>**select avg(salary) from employee;​



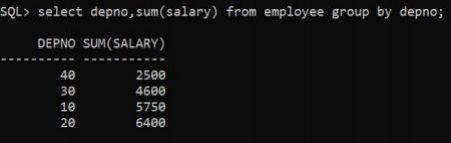
Q7 List the department number along with the number of employee working in that department.

**SQL>**select depno, count(\*) from employee group by depno;​



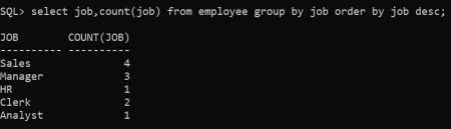
Q8 List the department number and the total salary paid to each department

**SQL>**select depno, sum(salary) from employee group by depno;​



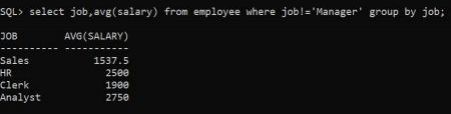
Q9 List the jobs and the number of employee in each job. And the result should be in decreasing order of number of employee .

**SQL>**​select job,count(job) from employee group by job order by job desc;



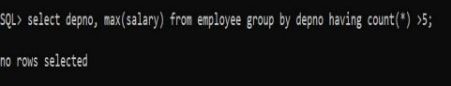
Q10 List the average salary of each job excluding managers

**SQL>**select job,avg(salary) from employee where job != ‘Manager’ group by job;​



Q11 List the maximum salary of each department having more than 5 employee

**SQL>**select depno,max(salary) from employee group by depno having count(\*) > 5;​



# LAB 7

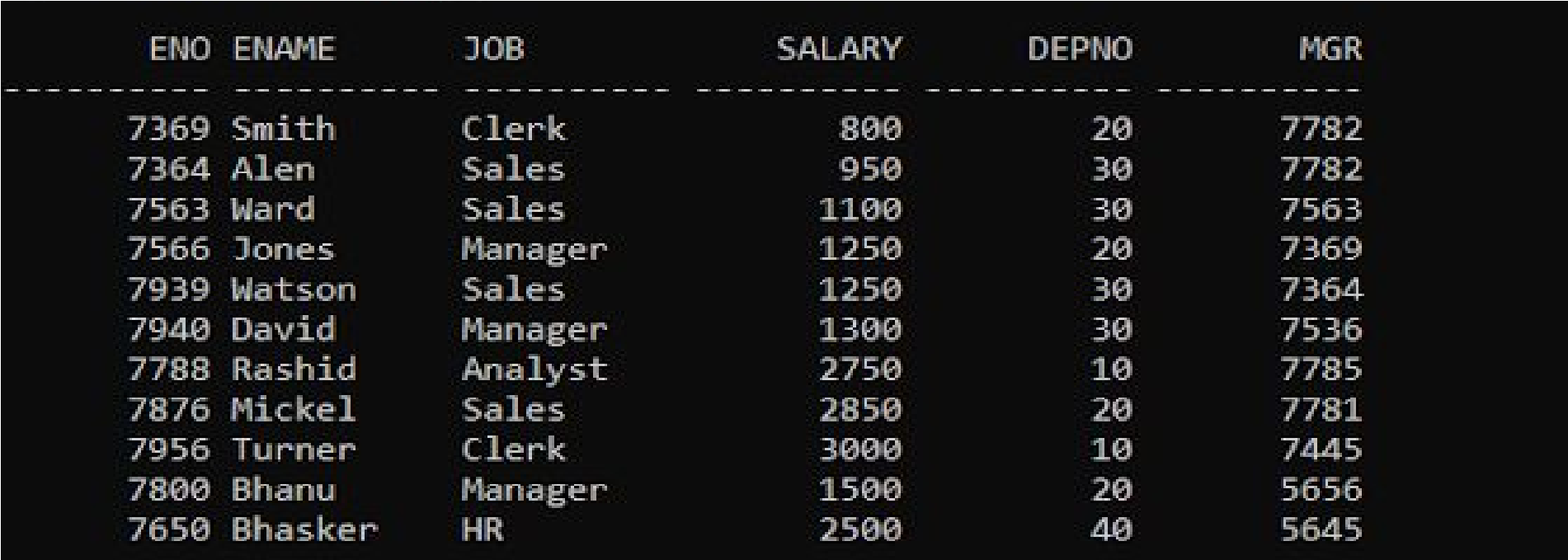
**Employee**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Eno** | **Ename** | **Job** | **Salary** | **Depno** | **Mgr** |
| **7369** | **Smith** | **Clerk** | **800** | **20** | **7782** |
| **7364** | **Alen** | **Sales** | **950** | **30** | **7782** |
| **7563** | **Ward** | **Sales** | **1100** | **30** | **7563** |
| **7566** | **Jones** | **Manager** | **1250** | **20** | **7369** |
| **7939** | **Watson** | **Sales** | **1250** | **30** | **7364** |
| **7940** | **David** | **Manager** | **1300** | **30** | **7536** |
| **7788** | **Rashid** | **Analyst** | **2750** | **10** | **7785** |
| **7876** | **Mickel** | **Sales** | **2850** | **20** | **7781** |
| **7956** | **Turner** | **Clerk** | **3000** | **10** | **7445** |
| **7800** | **Bhanu** | **Manager** | **1500** | **20** | **5656** |
| **7650** | **Bhasker** | **HR** | **2500** | **40** | **5645** |

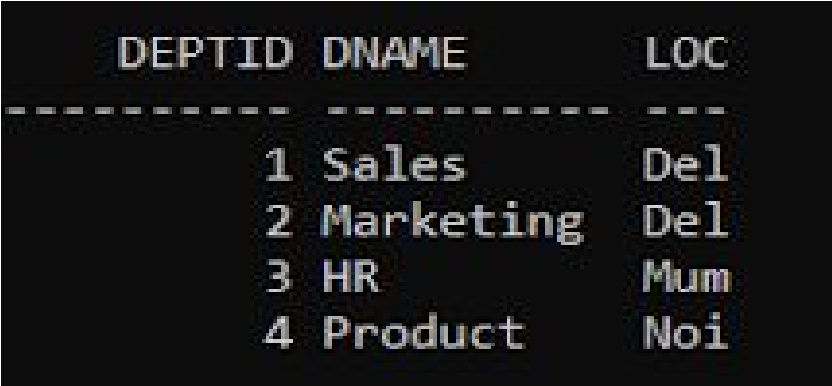
**Department**

|  |  |  |
| --- | --- | --- |
| **Depno** | **Dep\_Name** | **Location** |
| **10** | **PRODUCTION** | **NOI** |
| **20** | **ADMIN** | **DEL** |
| **30** | **REGISTRAR** | **DEL** |
| **40** | **HR** | **GZB** |

**SQL>SELECT \* FROM EMPLOYEE;**

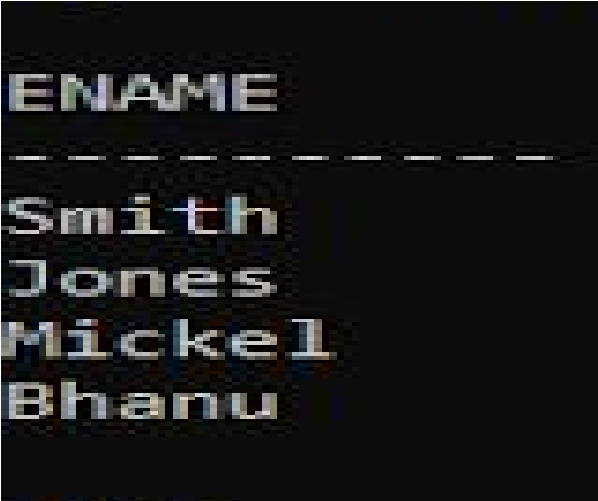


**SQL>SELECT \* FROM DEPARTMENT;**



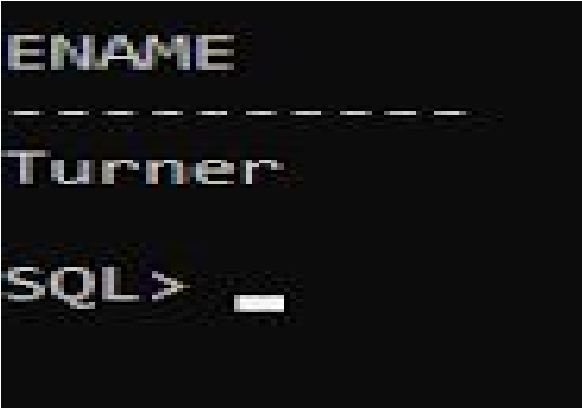
**Q1 list the employees belonging to department of Smith**

**SQL>** select ename from employee where depno = (select from employee where​ ename = ‘Smith’);



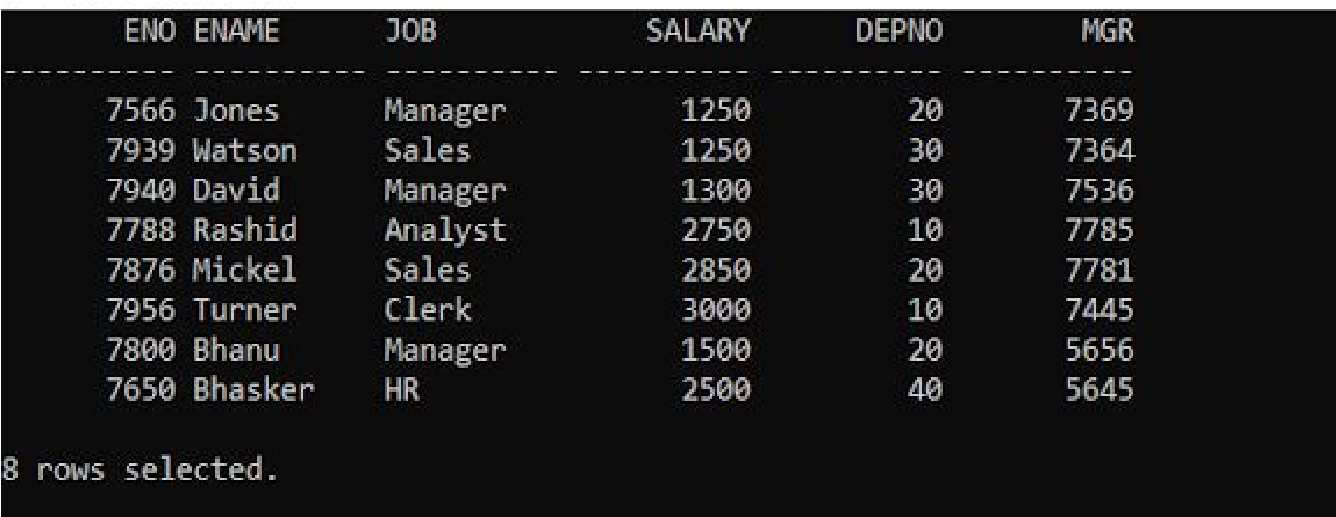
**Q2 list the name of employees drawing highest salary.**

**SQL>**​select ename from employee where salary = (Select max(Salary) from employee);



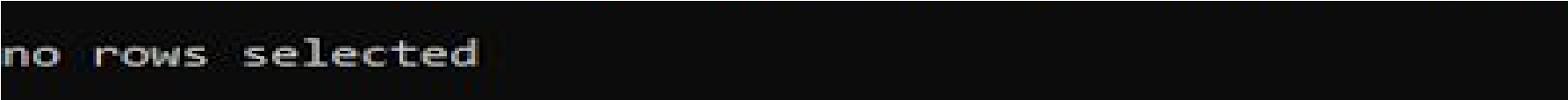
**Q3 list all the details of employee whose salary is greater than average salary of employees where department no is 30.**

**SQL>**​select \* from employee where salary >(select avg(salary) from employees where depno =30);



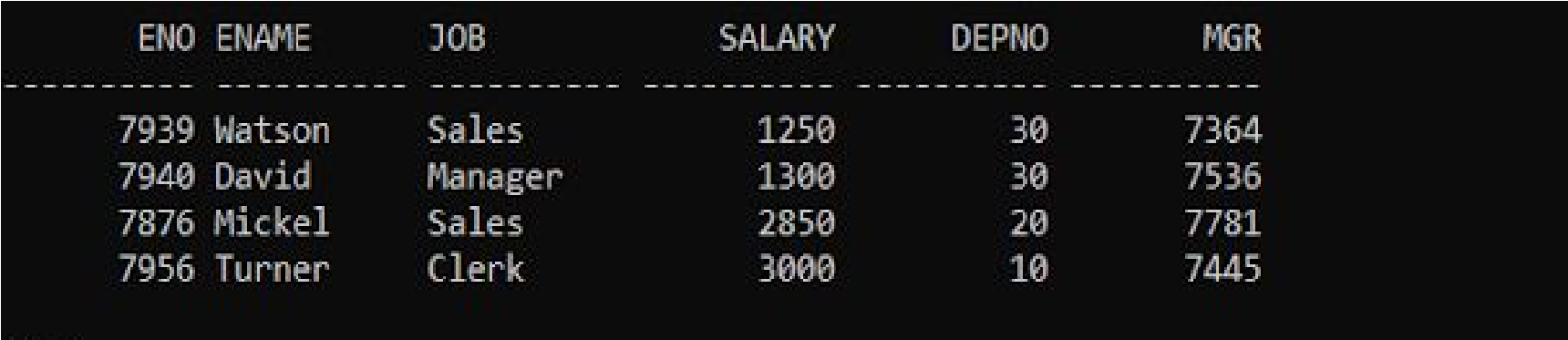
**Q4 list the job with highest average salary.**

**SQL>**​select job, avg(salary) from employee group by job having avg(salary)=(select max(salary) from employee);



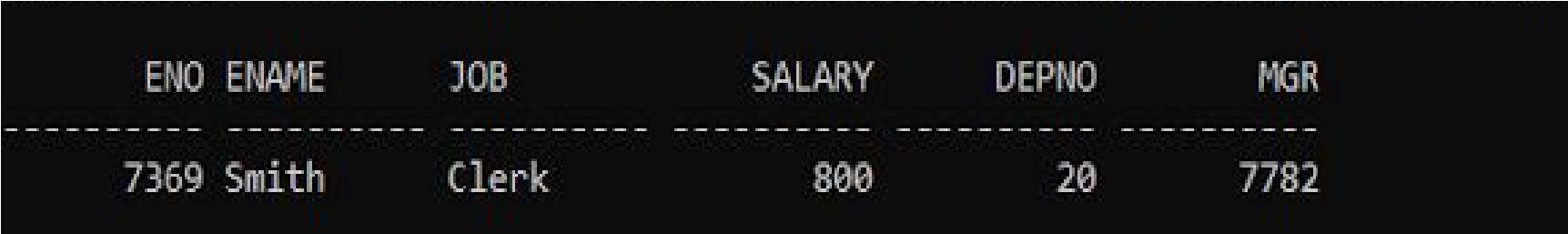
**Q5 list the employee details who earn salary greater than average salary for their department.**

**SQL>**​select \* from employee em where salary>(select avg(salary) from employee where Depno=em.Depno.);



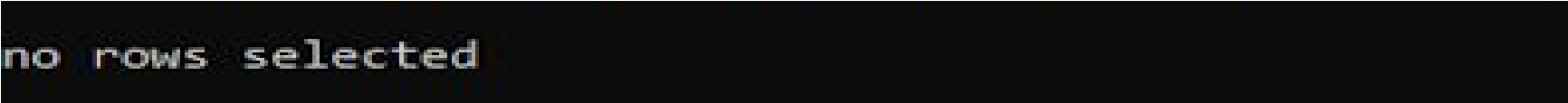
**Q6 list the employee details whose salary is less than equal to the minimum salary of their department.**

**SQL>**​select \* from employee where salary <=(select min(salary) from employee where depno=employee.depno);



**Q7 Retrieve name and salary of employee who work for cse department**

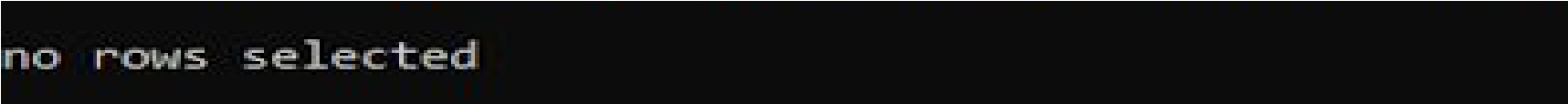
**SQL>**​select name, salary from employee where depno=(select depno from department where​ dep\_name="CSE");



**Q8 list all employee names working under the same manager of Ford.**

**SQL>**​select ename from employee where mgr =(select mgr from employee where ename =​

‘Ford’);



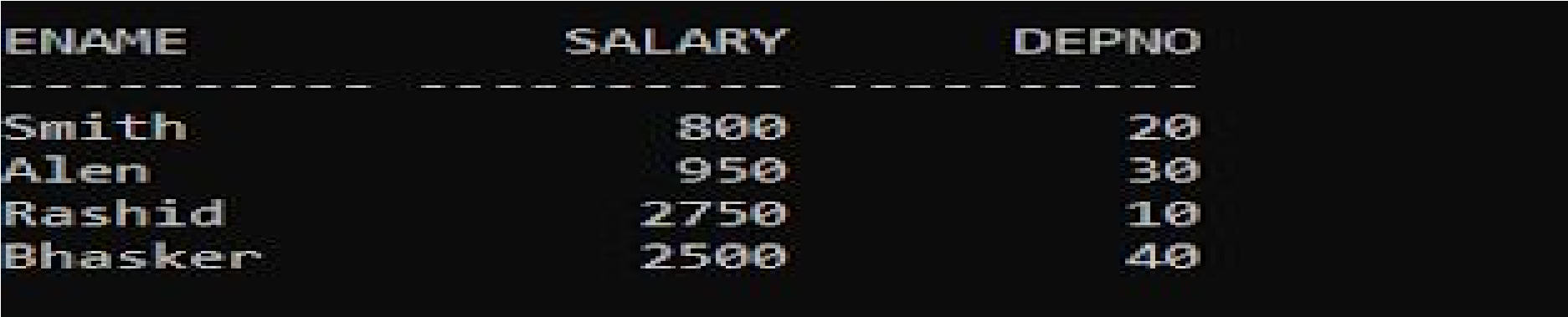
**Q9 list all the employee name belonging to department of turner or king**

**SQL>**​select ename from employee where depno in (select depno from employee where ename="turner " or ename="King");



**Q10 write a query to display name, salary , deptno, of those employees who earn such an amount of salary which is smallest of any department**

**SQL>**Select ename ,salary, depno from employee where salary in(select min(salary)​ from employee group by depno);



**Lab 8**

Emp\_poj1

|  |  |  |
| --- | --- | --- |
| **Eno** | **Ename** | **Job Salary Depno Mgr** |
| 7369 | Smith | Clerk 11800 20 7782 |
| 7364 | Alen | Sales 950 30 7782 |
| 7563 | Ward | Sales 1100 30 7563 |
| 7566 | Jones | Manager 1250 20 7369 |
| 7939 | Watson | Sales 1250 30 7364 |
| 7940 | David | Manager 1300 30 7536 |
| 7788 | Rashid | Analyst 3000 10 7785 |
| 7876 | Mickel | Sales 2850 20 7781 |
| 7956 | Turner | Clerk 3000 10 7445 |
| 7800 | Bhanu | Manager 1500 20 5656 |
| 7650 | Bhasker | HR 2500 40 5645 |

Emp\_proj2

|  |  |
| --- | --- |
| 7245 | Mary |
| 7650 | Bhasker |

|  |  |
| --- | --- |
| **Eno** | **Ename** |
| 7939 | Watson |
| 7940 | David |
| 7788 | Rashid |
| 7876 | Mickel |
| 7878 | Smith |
| 7900 | Alex |
| 7126 | Jony |

Department

**Job Salary Depno**​​Sales 1250 30

Manager 1300 30​ ​Analyst 12750 10

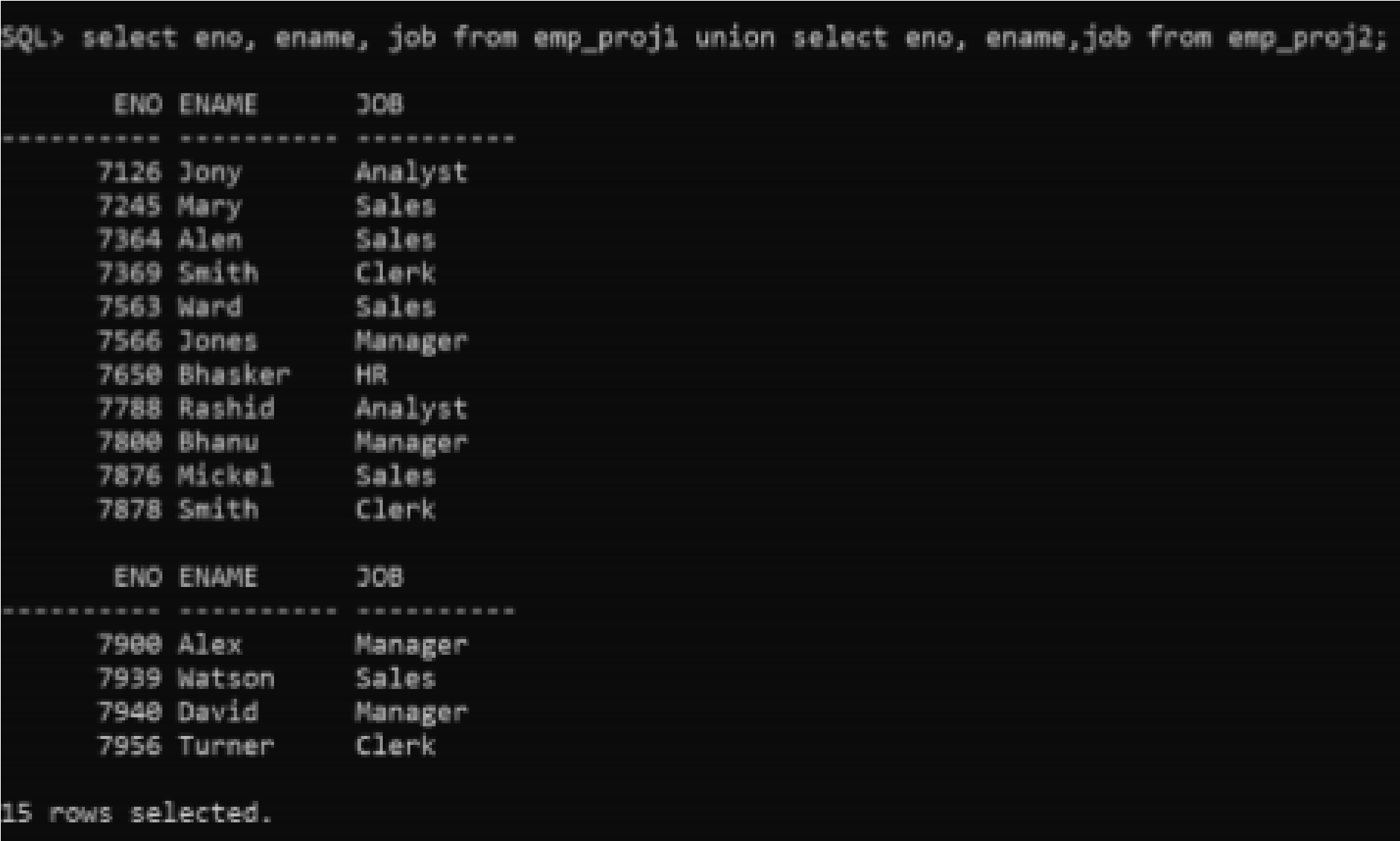
Sales 2850 20​ ​Clerk 5500 10​ ​Manager

7800 20​ ​Analyst 6500 20 ​ ​Sales 5000 30​ ​HR 2500 40

|  |  |
| --- | --- |
| Depno | Dep\_Name Location |
| 10 | PRODUCTION NOI |
| 20 | ADMIN DEL |
| 30 | REGISTRAR DEL |
| 40 | HR GZB |

**Q1 Display the Eno. Ename, Job of those employee, who are in Emp\_proj1 or Emp\_proj2.**

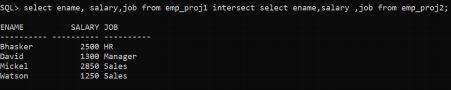
SQL> select eno, ename, job from emp\_proj1 union select eno, ename,job from emp\_proj2;

OUTPUT> 

**Q2 Display the Ename,salary ,Job of those employee, who are in Emp\_proj1 and Emp\_proj2.**

SQL> select ename, salary,job from emp\_proj1 intersect select ename,salary ,job from emp\_proj2;

OUTPUT>

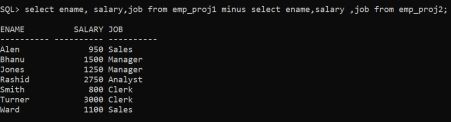


**Q3 Display the Eno. Ename of those employee, who are in Emp\_proj1 but not** ​ **in**​ **Emp\_proj2.**

SQL> select ename, salary,job from emp\_proj1 minus select

ename,salary ,job from emp\_proj2;

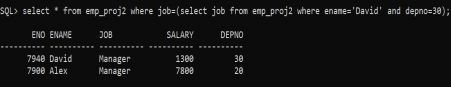
OUTPUT>



**Q4 Display the details from Emp\_proj2 of those employee , whose job is same** ​ **as**​  **that of David of depno 30**

SQL> select \* from emp\_proj2 where job=(select job from emp\_proj2 where ename=’David’ and depno=30);

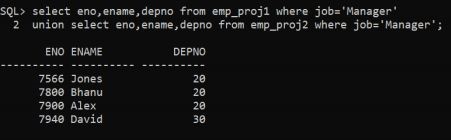
OUTPUT>



**Q5 Display all the details of manager from Both the tables using both set operators and subqueries**

SQL> select eno,ename,depno from Emp\_proj1 where job='Manager' union select eno,ename,depno from Emp\_proj2 where job='Manager' ;

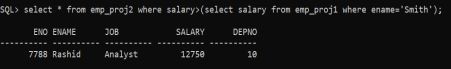
OUTPUT>



**Q6 Display details of the employees whose salary is more than the salary of Smith in Emp\_proj1.**

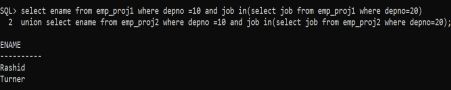
SQL> select \* from emp\_proj1 where salary>(select salary from emp\_proj1 where ename='smith');

OUTPUT>



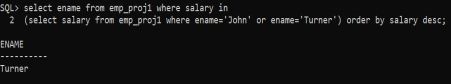
**Q7 list the names of employees of deptid 10 who works in the designation same as deptid 20 from both the tables.**

SQL> select ename from emp\_proj1 where depno =10 and job in(select j​ ob​ from emp\_proj1 where depno=20) union select ename from ​ emp\_proj2​ where depno =10 and job in(select job from emp\_proj2 where ​ depno=20)​ OUTPUT>



**Q8 list the name of employees whose salary is same as the salary of John or Turner. List the result in descending order of salary.**

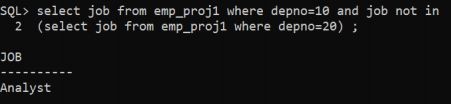
SQL> select ename from emp\_proj1 where salary in (select salary from emp\_proj1 where ename='John' or ename='Turner') order by salary ​ desc;​ OUTPUT>



**Q9 Write a query in SQL to list any job of deptid 10 those that are not found in deptid 20.**

SQL> select job from emp\_proj1 where depno=10 and job not in (select job from emp\_proj1 where depno=20) ;​

OUTPUT>

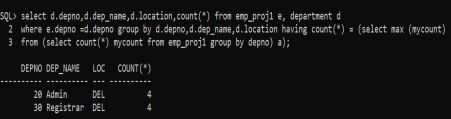


**Q10 list the details of the departments where maximum number of employees are working.**

SQL> select d.depno,d.dep\_name,d.location,count(\*) from emp\_proj1 e, department d where e.depno =d.depno group by

d.depno,d.dep\_name,d.location having count(\*) = (select max (mycount) from

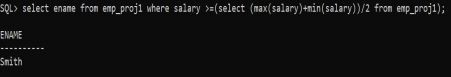
(select count(\*) mycount from emp\_proj1 group by depno) a); OUTPUT>



**Q11 list the employees whose salary is equal or more to the average of**​  **maximum and minimum salary.**

SQL> select ename from emp\_proj1 where salary >=(select (max(salary)+min(salary))/2 from emp\_proj1) ;

OUTPUT>



**Q12. Write a query in SQL to list the name of departments where atheist** ​ **2**​ **employees are working in that department.**

SQL> select dep\_name , count(eno) from department,emp\_proj1 where department.depno = emp\_proj1.depno group by department.dep\_name having count(eno) >= 2;

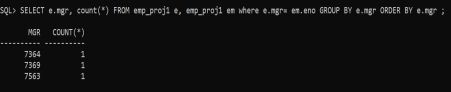
OUTPUT>



**Q**​**13.**​ **Write a query in SQL to list the manager no and the number of**​  **employees working for those managers in ascending order on manager id.**

SQL> SELECT e.mgr, count(\*) FROM emp\_proj1 e, emp\_proj1 em where e.mgr= em.eno GROUP BY e.mgr ORDER BY e.mgr;

OUTPUT>



Q.14 Write a query in SQL to list the employee id, name, location, department ​ of​ all the departments 10 and 30

SQL> select eno,ename,location,dep\_name from emp\_proj1,department where department.depno = emp\_proj1.depno and emp\_proj1.depno in(10,30);

OUTPUT>

