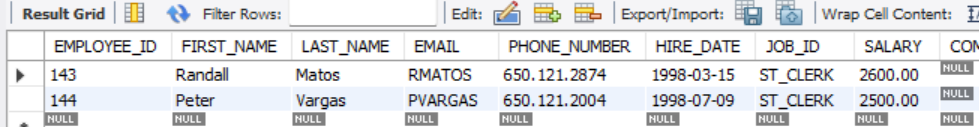
These exercises can be used for extra practice after you have discussed the following topics: basic SQL SELECT statement, basic *i*SQL\*Plus commands, and SQL functions.

1. Show all data of the clerks who have been hired after the year 1997.

SELECT \* FROM employees whereJOB\_ID = "ST\_CLERK" AND YEAR(HIRE\_DATE) > 1997



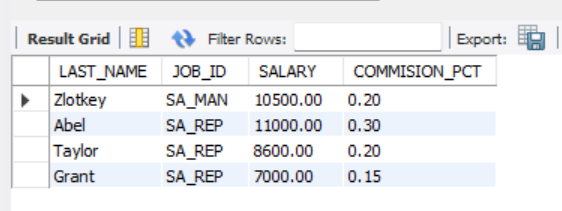
2. Show the last name, job, salary, and commission of those employees who earn commission. Sort the data by the salary in descending order.

SELECT employees.LAST\_NAME, employees.JOB\_ID, employees.SALARY, employees.COMMISION\_PCT

FROM employees

where COMMISION\_PCT IS NOT NULL

ORDER BY LAST\_NAME DESC



3. Show the employees that have no commission with a 10% raise in their salary (round off the  
salaries).

SELECT LAST\_NAME, JOB\_ID, SALARY, COMMISION\_PCT,

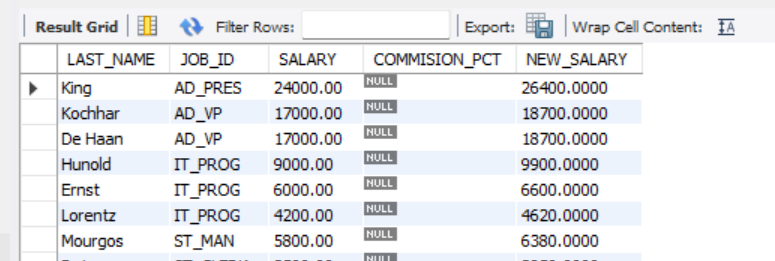
CASE

WHEN COMMISION\_PCT IS NULL THEN SALARY \* 1.10

ELSE SALARY

END AS NEW\_SALARY

FROM employees



4. Show the last names of all employees together with the number of years and the number of  
completed months that they have been employed.

5. Show those employees that have a name starting with J, K, L, or M.

select \* FROM employees where (

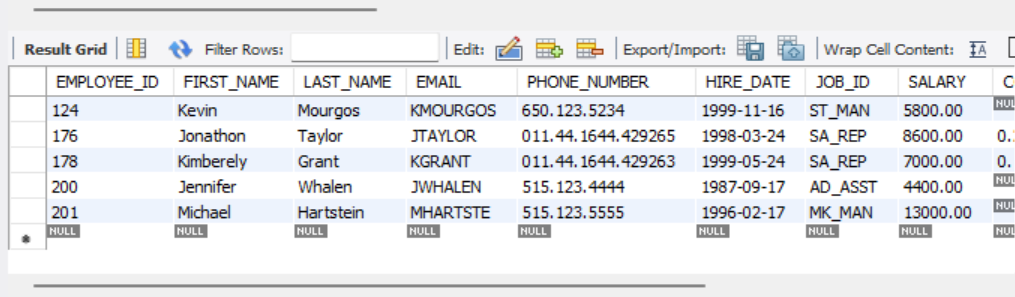
first\_name LIKE 'J%' OR

first\_name LIKE 'K%' OR

first\_name LIKE 'M%' OR

first\_name LIKE 'L%'

)



6. Show all employees, and indicate with “Yes” or “No” whether they receive a commission.

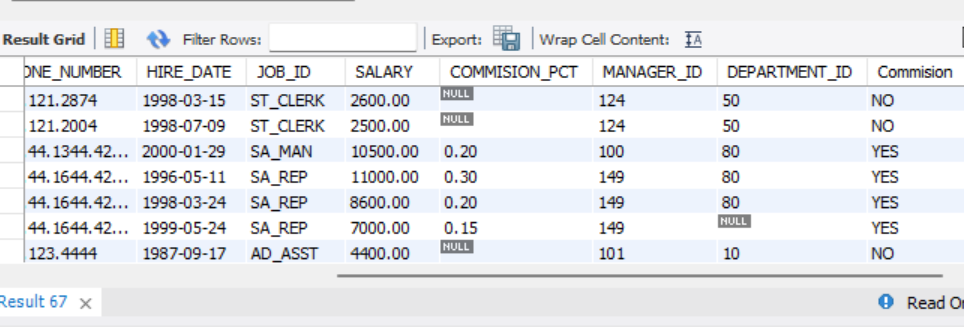
SELECT \*,

CASE WHEN COMMISION\_PCT IS NULL THEN 'NO'

ELSE 'YES'

END AS Commision

FROM employees;



7. Show the department names, locations, names, job titles, and salaries of employees who work  
in location 1800.

SELECT employees.LAST\_NAME, employees.JOB\_ID, employees.SALARY,

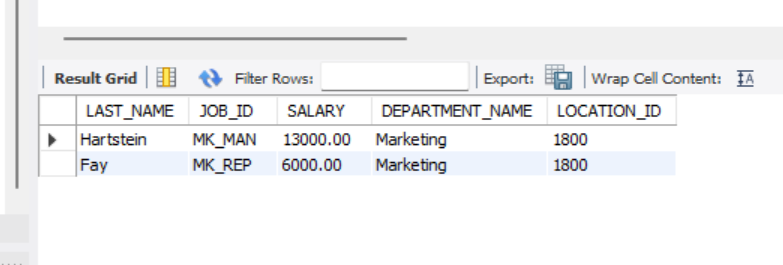
departments.DEPARTMENT\_NAME, locations.LOCATION\_ID

FROM employees

JOIN departments ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

JOIN locations ON departments.LOCATION\_ID = locations.LOCATION\_ID

WHERE locations.LOCATION\_ID = 1800

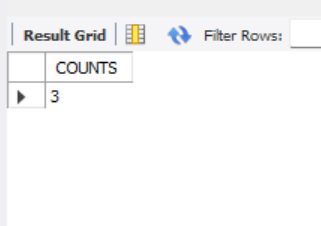


8. How many employees have a name that ends with an n? Create two possible solutions.

SELECT COUNT(\*) AS COUNTS

FROM employees

WHERE LAST\_NAME LIKE '%n'



9. Show the names and locations for all departments, and the number of employees working in each department. Make sure that departments without employees are included as well.

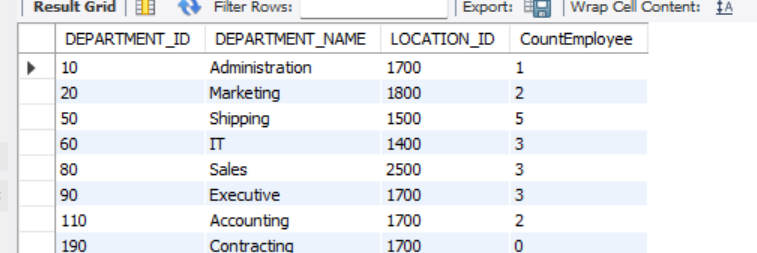
SELECT departments.DEPARTMENT\_ID,departments.DEPARTMENT\_NAME, departments.LOCATION\_ID,

COUNT(EMPLOYEE\_ID) AS COUNT\_EMPLOYEE

FROM departments

LEFT JOIN employees ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

GROUP BY departments.DEPARTMENT\_ID, departments.LOCATION\_ID

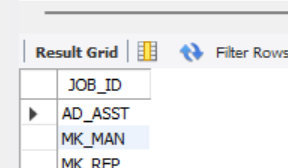


10. Which jobs are found in departments 10AD and 20?

select employees.JOB\_ID

FROM employees

where DEPARTMENT\_ID IN (10, 20)



11. Which jobs are found in the Administration and Executive departments, and how many  
employees do these jobs? Show the job with the highest frequency first.

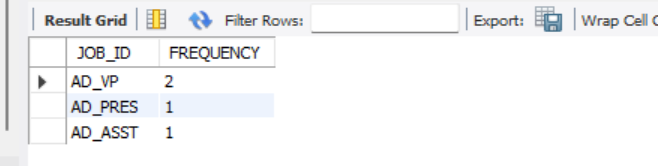
SELECT JOB\_ID, COUNT(\*) AS FREQUENCY

FROM employees WHERE JOB\_ID IN (

'AD\_VP', 'AD\_ASST', 'AD\_PRES'

)

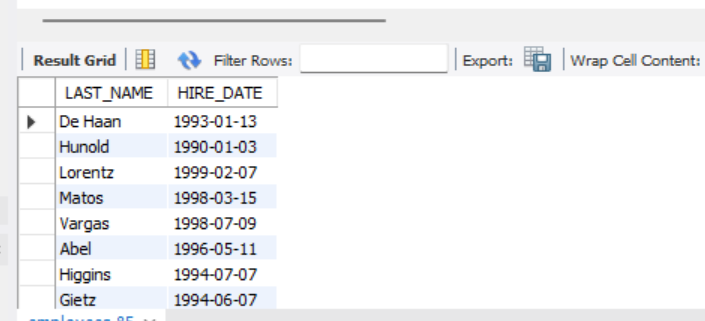
GROUP BY JOB\_ID ORDER BY FREQUENCY DESC



12. Show all employees who were hired in the first half of the month (before the 16th of the month).

SELECT LAST\_NAME, HIRE\_DATE FROM employees

WHERE DAY(HIRE\_DATE) <= 15

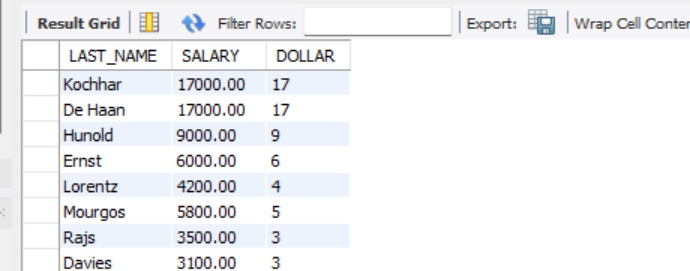


13. Show the names, salaries, and the number of dollars (in thousands) that all employees earn.

SELECT LAST\_NAME, SALARY,

floor(SALARY / 1000) AS DOLLAR

FROM employees



14. Show all employees who have managers with a salary higher than $15,000. Show the  
following data: employee name, manager name, manager salary, and salary grade of the manager.

15. Show the department number, name, number of employees, and average salary of all departments, together with the names, salaries, and jobs of the employees working in each department.

SELECT departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME,

COUNT(employees.EMPLOYEE\_ID) OVER

(partition by departments.DEPARTMENT\_ID) AS NUMBER\_EMPLOYEE,

AVG(employees.SALARY) OVER (partition by departments.DEPARTMENT\_ID) AS AVG\_SALARY,

employees.LAST\_NAME,

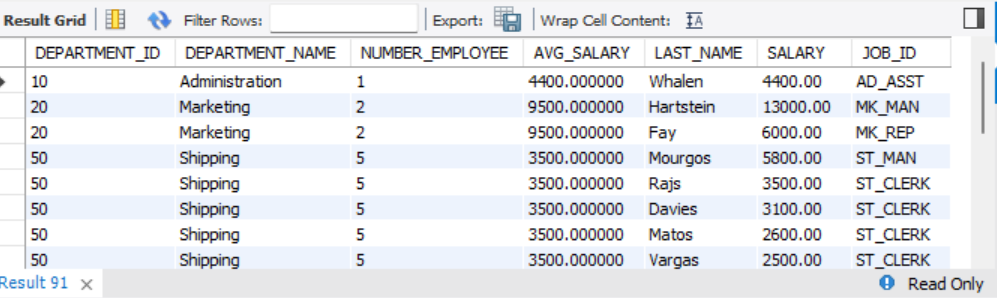
employees.SALARY,

employees.JOB\_ID

FROM departments

LEFT JOIN employees ON departments.DEPARTMENT\_ID = employees.DEPARTMENT\_ID

ORDER BY departments.DEPARTMENT\_ID, employees.SALARY DESC;



16. Show the department number and the lowest salary of the department with the highest average salary.

SELECT departments.DEPARTMENT\_ID,

min(employees.SALARY) as MIN\_SALARY

from departments

JOIN employees ON departments.DEPARTMENT\_ID = employees.DEPARTMENT\_ID

WHERE departments.DEPARTMENT\_ID = (

select DEPARTMENT\_ID

FROM(

SELECT DEPARTMENT\_ID,

AVG(SALARY) AS AVG\_SALARY

FROM employees

group by DEPARTMENT\_ID

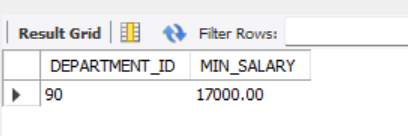
order by AVG\_SALARY DESC

LIMIT 1

) AS HIGEST

)

GROUP BY departments.DEPARTMENT\_ID



17. Show the department numbers, names, and locations of the departments where no sales representatives work.

SELECT departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME, departments.LOCATION\_ID

FROM departments

WHERE NOT exists (

SELECT 1

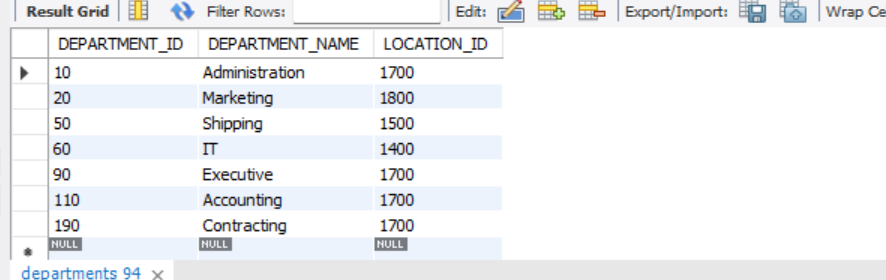
FROM employees

JOIN jobs ON employees.JOB\_ID = jobs.JOB\_ID

WHERE employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

and jobs.JOB\_TITLE = 'Sales Representative'

)



18. Show the department number, department name, and the number of employees working in each department that.

SELECT departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME,

COUNT(\*) FROM departments

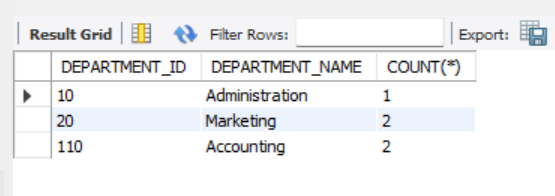
INNER JOIN employees

ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

GROUP BY departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME

HAVING COUNT(\*) < 3

ORDER BY COUNT(\*) ASC



SELECT departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME,

COUNT(\*) AS TotalCount

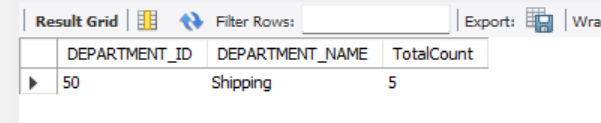
FROM departments

INNER JOIN employees

ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

GROUP BY departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME

ORDER BY TotalCount DESC LIMIT 1;



SELECT departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME,

COUNT(\*) AS TotalCount

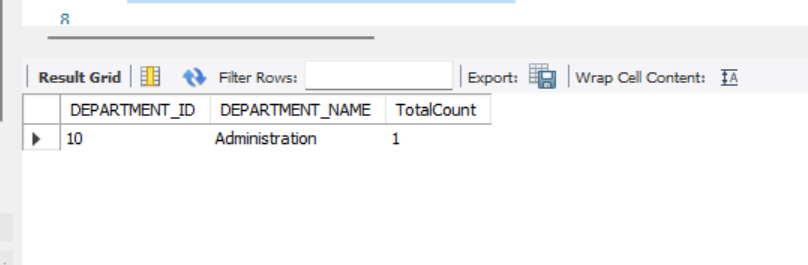
FROM departments

INNER JOIN employees

ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

GROUP BY departments.DEPARTMENT\_ID, departments.DEPARTMENT\_NAME

ORDER BY TotalCount ASC LIMIT 1;



19. Show the employee number, last name, salary, department number, and the average salary in their department for all employees.

SELECT employees.EMPLOYEE\_ID, employees.LAST\_NAME,

departments.DEPARTMENT\_ID,

departments.SALARY

from employees

join

(SELECT avg(SALARY) SALARY, DEPARTMENT\_ID

FROM employees

group by DEPARTMENT\_ID)

departments ON employees.DEPARTMENT\_ID = departments.DEPARTMENT\_ID

20. Show all employees who were hired on the day of the week on which the highest number of employees were hired.

21. Create an anniversary overview based on the hire date of the employees. Sort the anniversaries in ascending order.

SELECT employees.LAST\_NAME, month(HIRE\_DATE) AS MONTH\_HIRE, employees.HIRE\_DATE

FROM employees

WHERE MONTH\_HIRE = 'January'

22. Find the job that was filled in the first half of 1990 and the same job that was filled during the same period in 1991.

23. Write a compound query to produce a list of employees showing raise percentages, employee IDs, and old salary and new salary increase. Employees in departments 10, 50, and 110 are given a 5% raise, employees in department 60 are given a 10% raise, employees in departments 20 and 80 are given a 15% raise, and employees in department 90 are not given a raise.

29. Write a query to display the top three earners in the EMPLOYEES table. Display their last names and salaries.

SELECT employees.LAST\_NAME, max(employees.SALARY)

FROM employees

GROUP BY LAST\_NAME

limit 3