1) Group functions work across many rows to produce one result per group.

**True**

2) Group functions include nulls in calculations.

**False**

3) The WHERE clause restricts rows before inclusion in a group calculation.

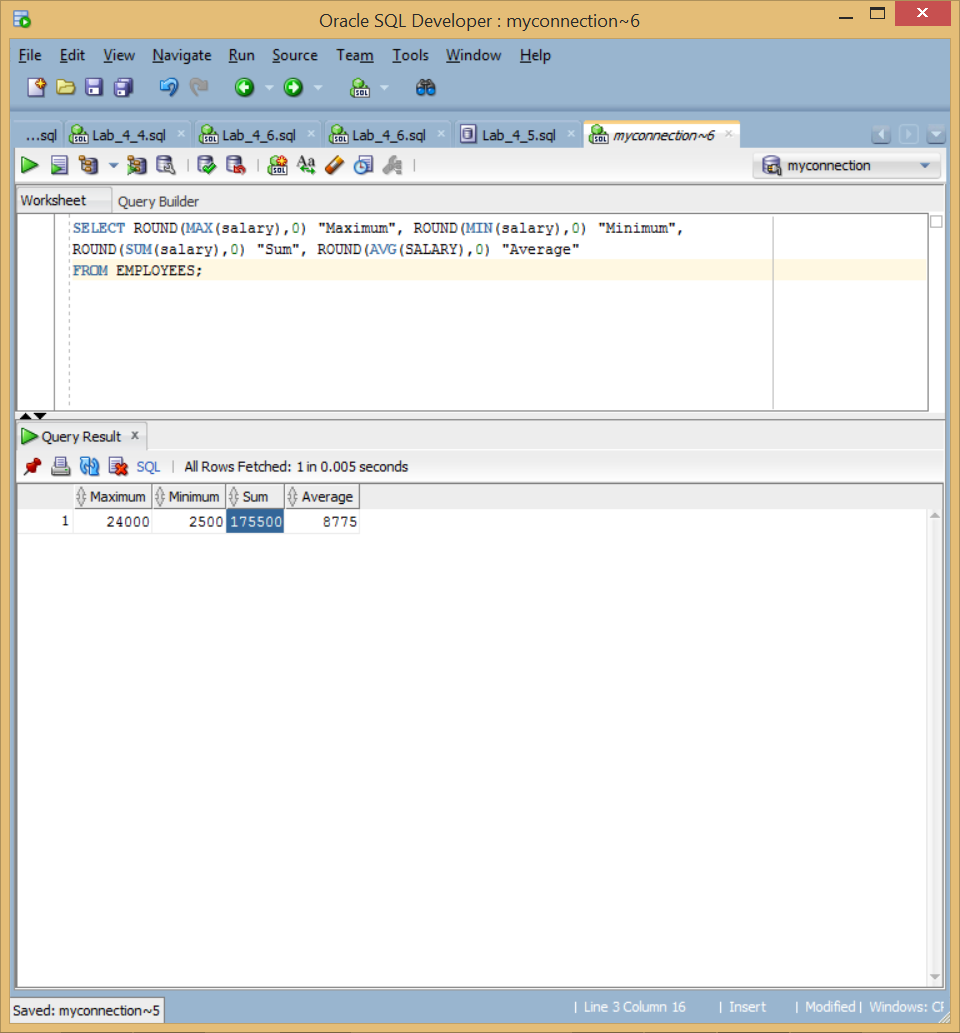
**True**

4) Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number. Save your SQL statement as lab\_05\_04.sql. Run the query.

SELECT ROUND(MAX(salary),0) "Maximum", ROUND(MIN(salary),0) "Minimum",

ROUND(SUM(salary),0) "Sum", ROUND(AVG(SALARY),0) "Average"

FROM EMPLOYEES;



5) Modify the query in lab\_05\_04.sql to display the minimum, maximum, sum, and average salary for each job type. Save lab\_05\_04.sql as lab\_05\_05.sql again. Run the statement in lab\_05\_05.sql.

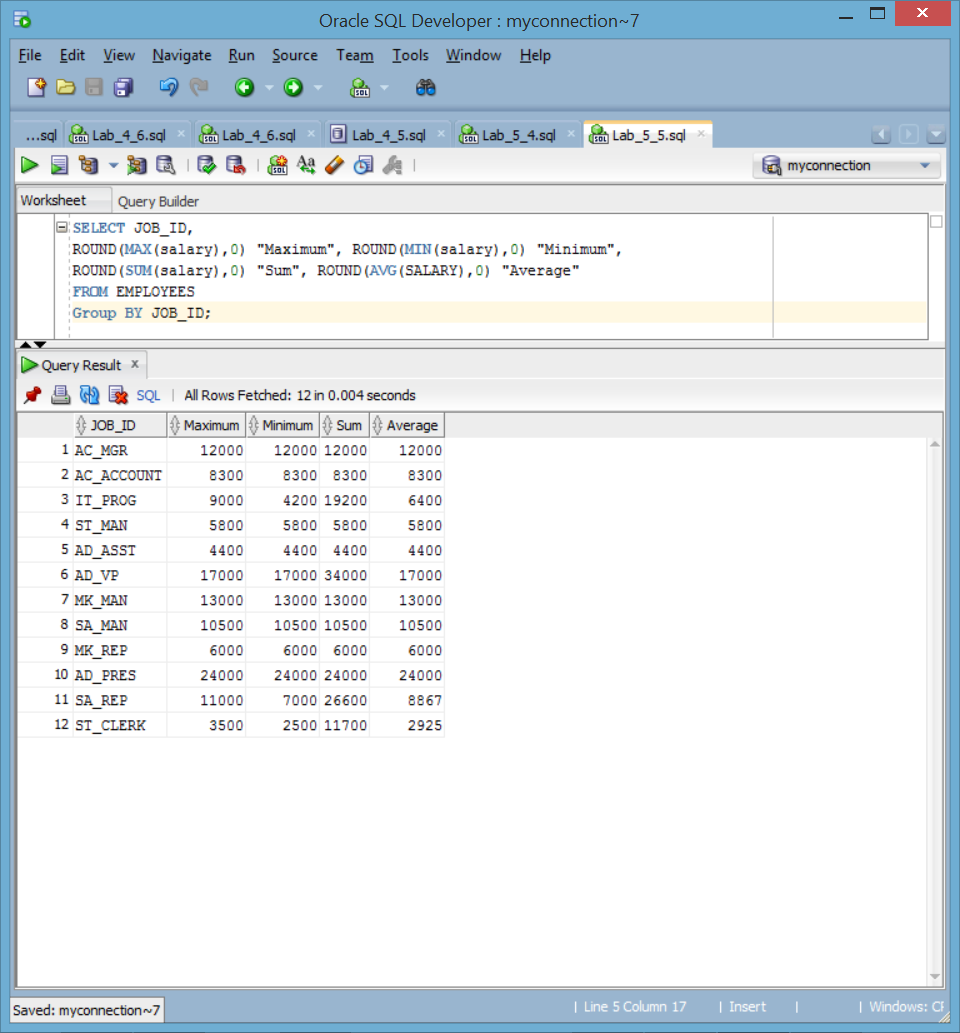
SELECT JOB\_ID,

ROUND(MAX(salary),0) "Maximum", ROUND(MIN(salary),0) "Minimum",

ROUND(SUM(salary),0) "Sum", ROUND(AVG(SALARY),0) "Average"

FROM EMPLOYEES

Group BY JOB\_ID;

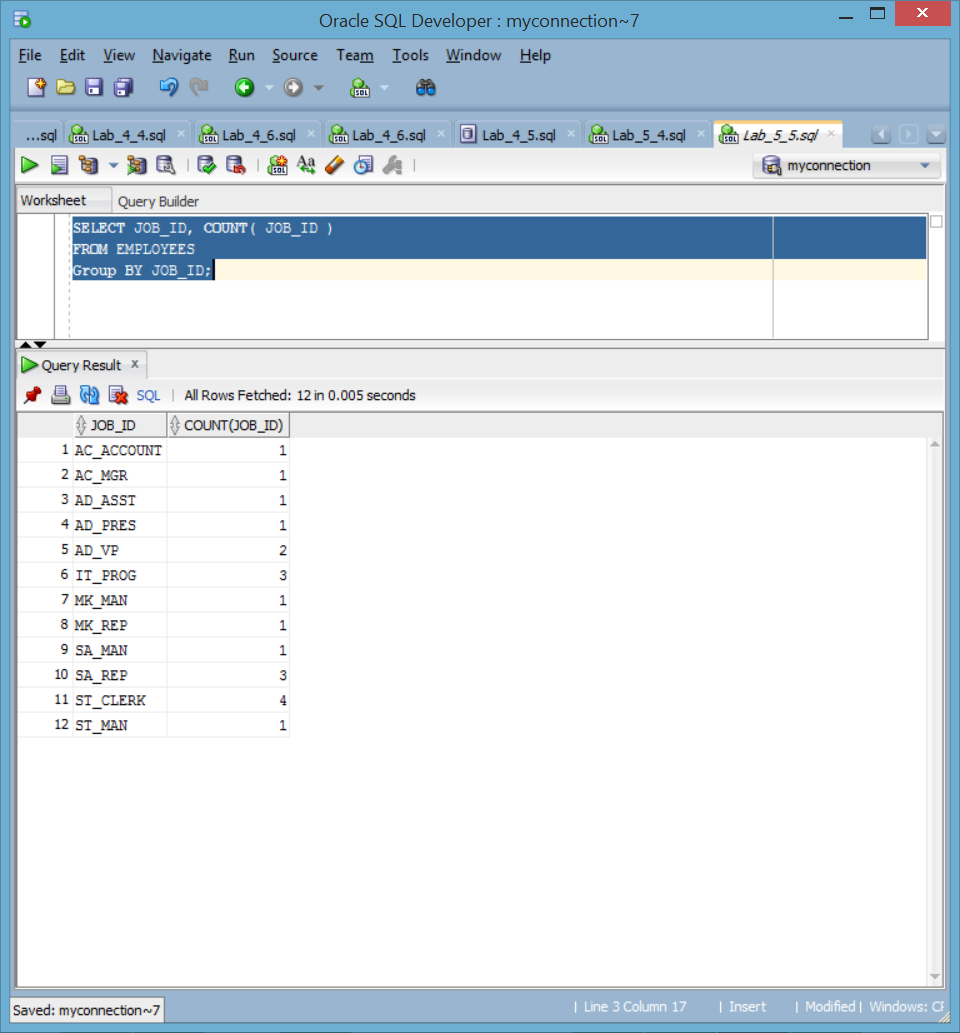


6) Write a query to display the number of people with the same job.

SELECT JOB\_ID, COUNT( JOB\_ID )

FROM EMPLOYEES

Group BY JOB\_ID;



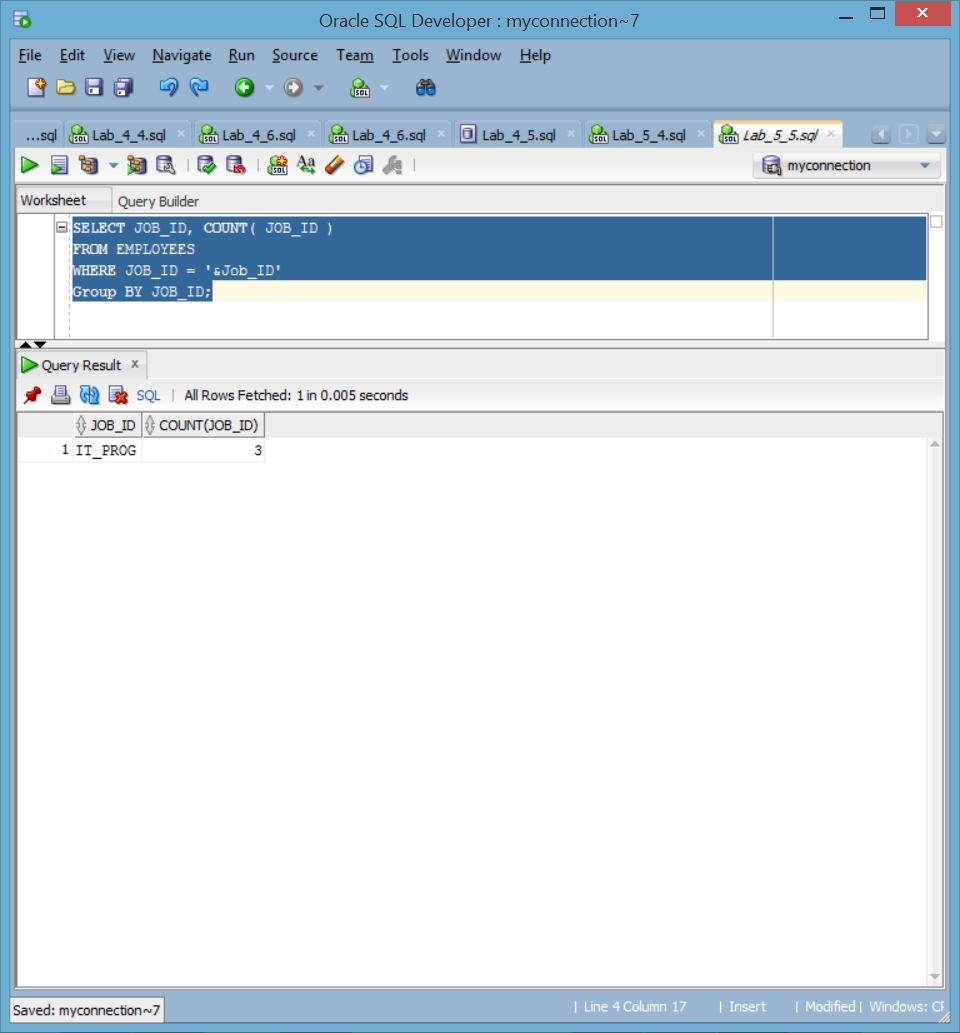
Generalize the query so that the user in the HR department is prompted for a job title. Save the script to a file named lab\_05\_06.sql. Run the query. Enter IT\_PROG when prompted.

SELECT JOB\_ID, COUNT( JOB\_ID )

FROM EMPLOYEES

WHERE JOB\_ID = '&Job\_ID'

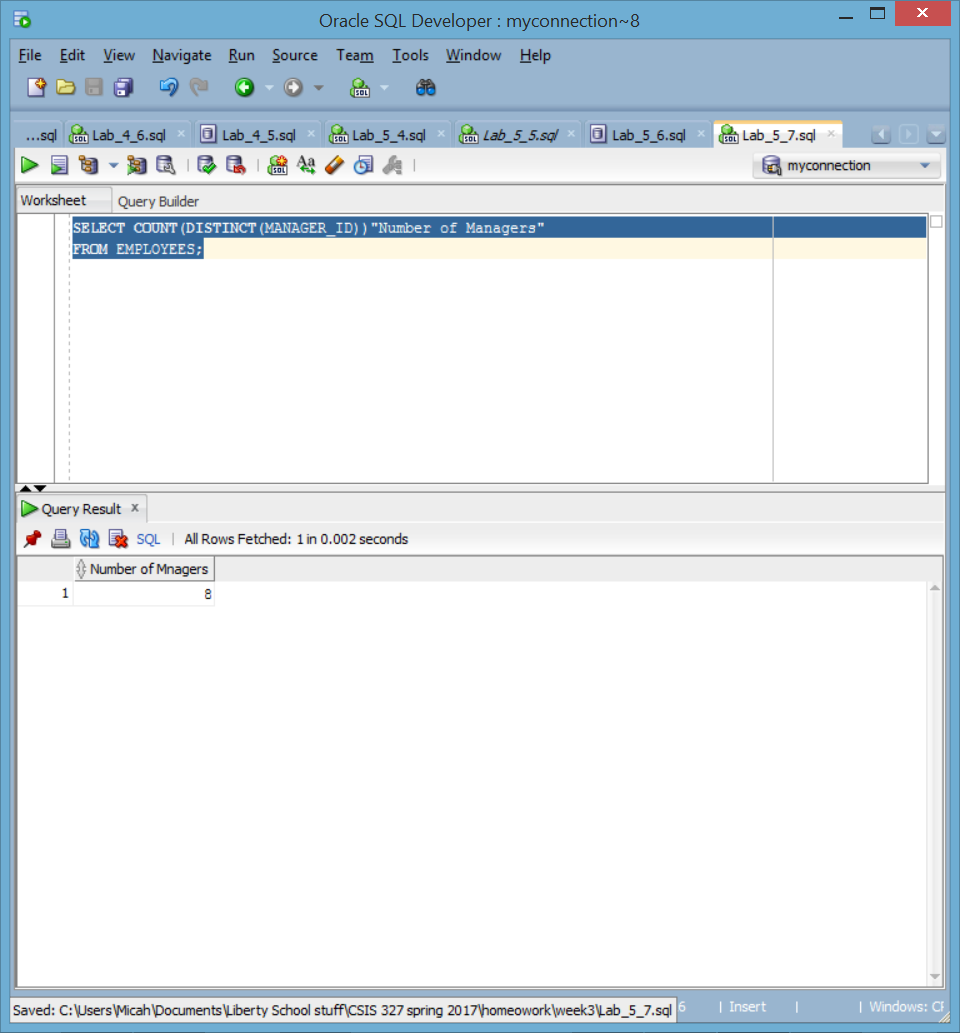
Group BY JOB\_ID;



7) Determine the number of managers without listing them. Label the column Number of Managers . Hint: Use the MANAGER\_ID column to determine the number of managers.

SELECT COUNT(DISTINCT(MANAGER\_ID))"Number of Managers"

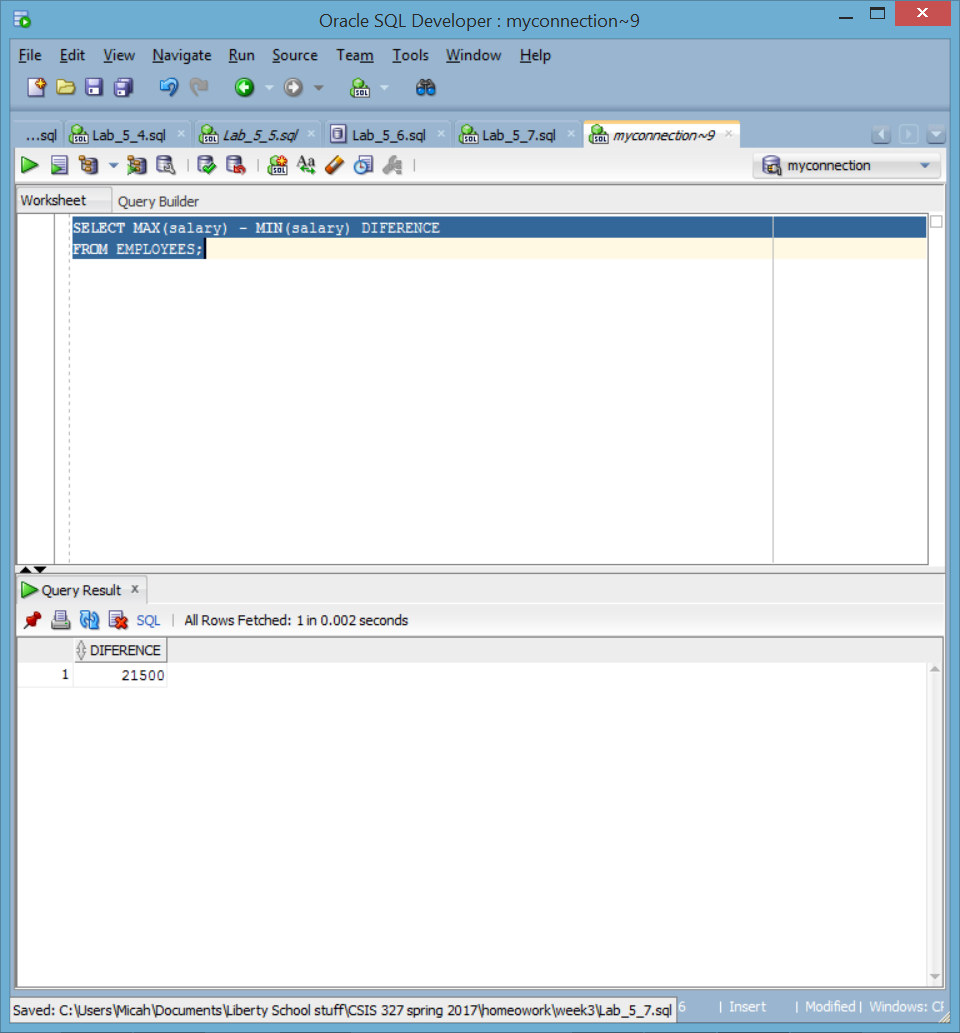
FROM EMPLOYEES;



8) Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.

SELECT MAX(salary) - MIN(salary) DIFERENCE

FROM EMPLOYEES;



9) Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is $6,000 or less. Sort the output in descending order of salary.

SELECT manager\_id, MIN(salary)

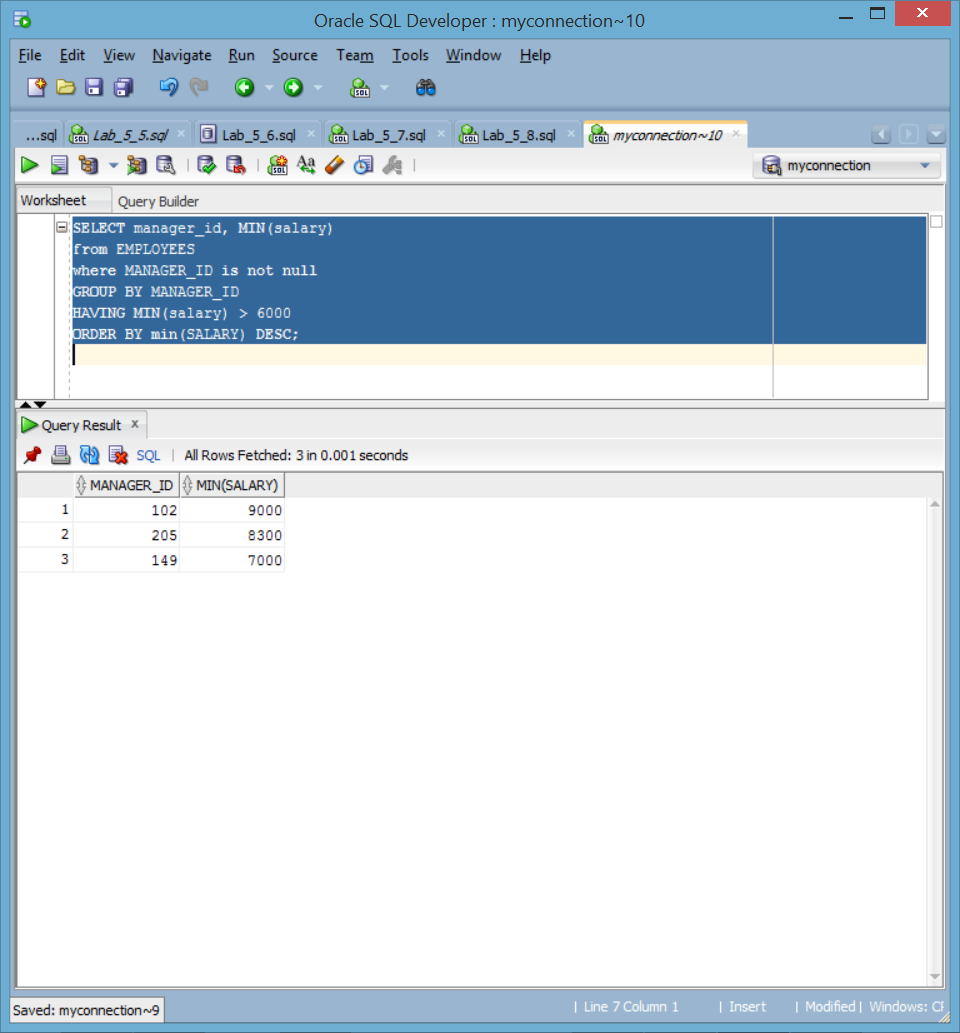
from EMPLOYEES

where MANAGER\_ID is not null

GROUP BY MANAGER\_ID

HAVING MIN(salary) > 6000

ORDER BY min(SALARY) DESC;



10) Create a query to display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

SELECT COUNT(\*),

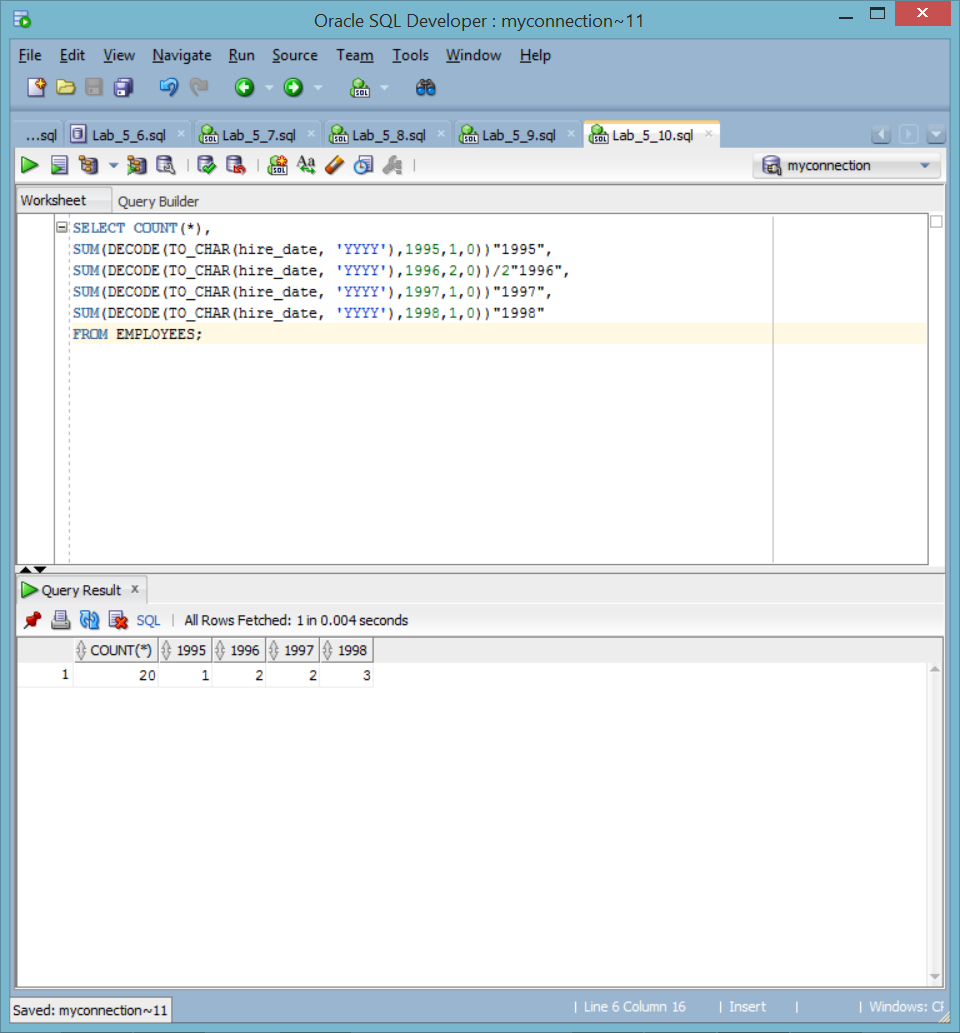
SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1995,1,0))"1995",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1996,1,0))"1996",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1997,1,0))"1997",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1998,1,0))"1998"

FROM EMPLOYEES;



11) Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.

SELECT job\_id "Job",

SUM(DECODE(department\_id , 20, salary)) "Department 20",

SUM(DECODE(department\_id , 50, salary)) "Department 50",

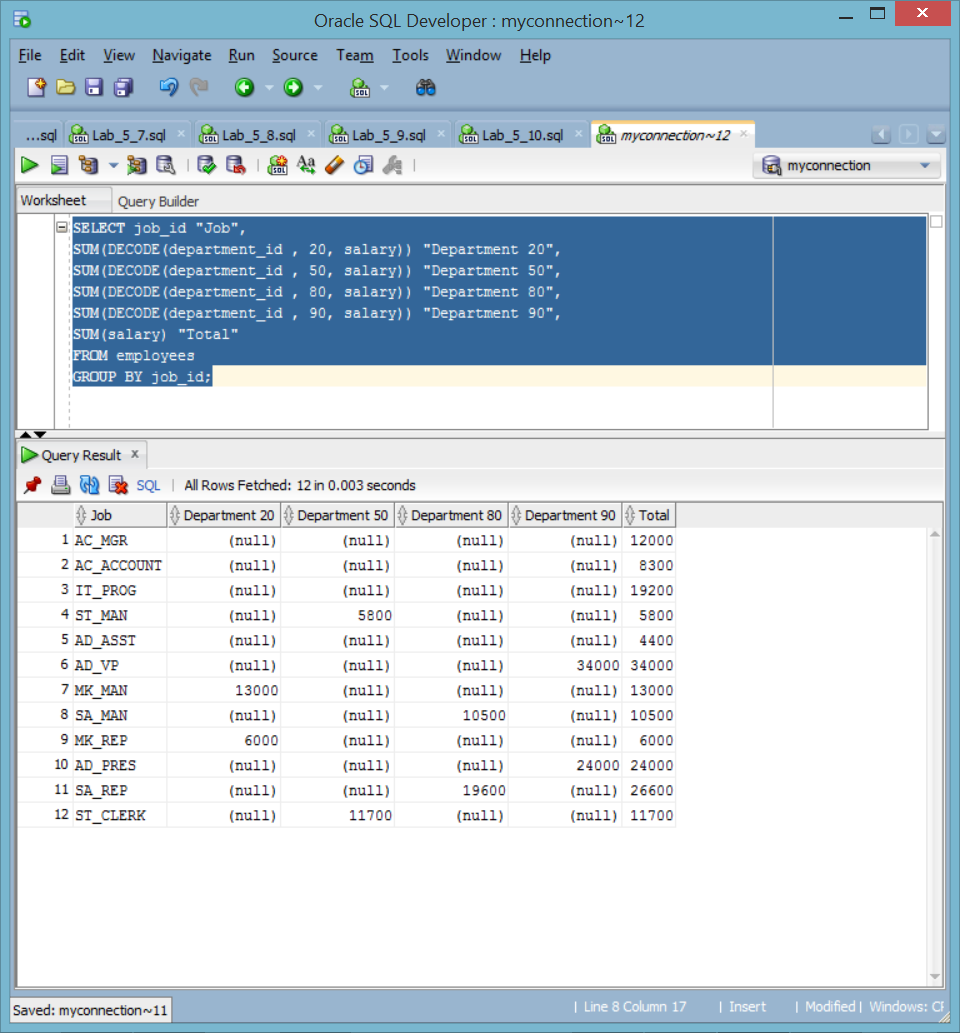
SUM(DECODE(department\_id , 80, salary)) "Department 80",

SUM(DECODE(department\_id , 90, salary)) "Department 90",

SUM(salary) "Total"

FROM employees

GROUP BY job\_id;

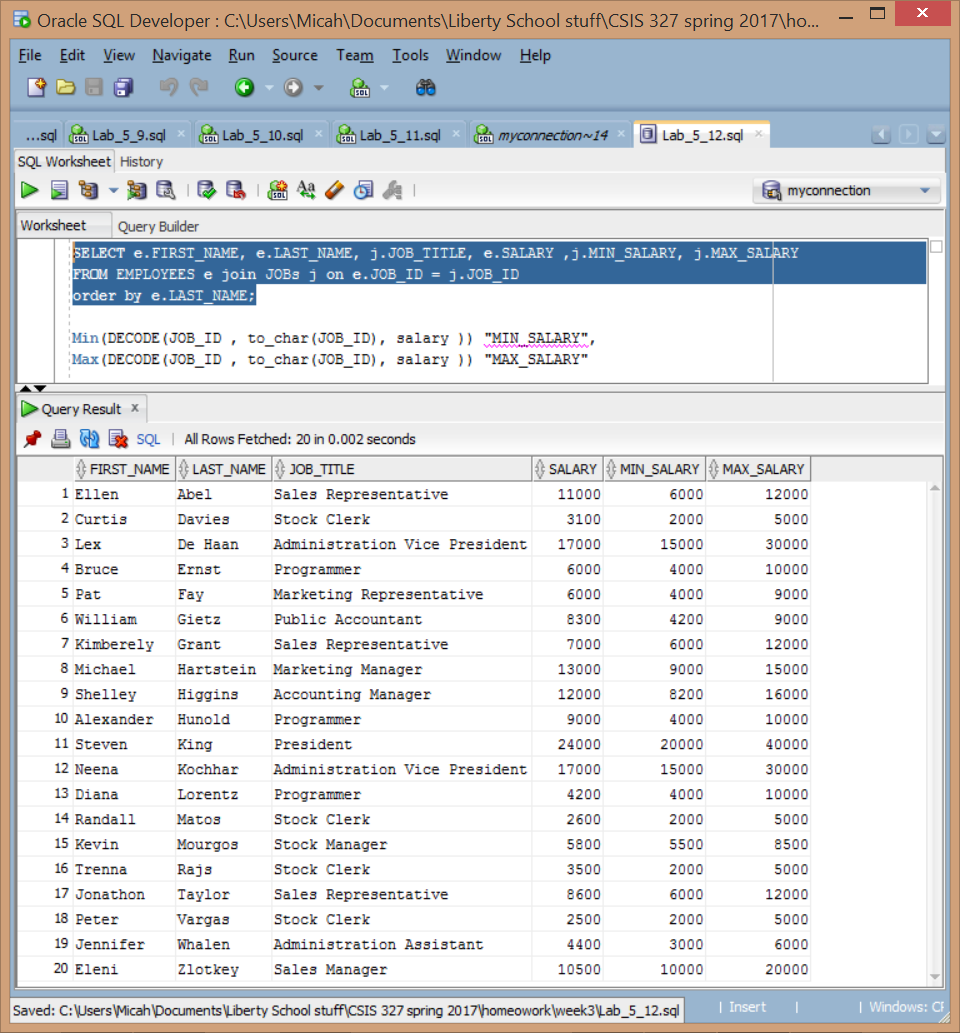


12) Create a query to display the salary of each employee and the subsequent minimum and maximum salary for their position.

SELECT e.FIRST\_NAME, e.LAST\_NAME, j.JOB\_TITLE, e.SALARY ,j.MIN\_SALARY, j.MAX\_SALARY

FROM EMPLOYEES e join JOBs j on e.JOB\_ID = j.JOB\_ID

order by e.LAST\_NAME;



13) Using your own version, create a matrix query that shows whether an employee’s salary is within the job title range that they occupy. The output should have at least one column named “Within range” that displays a “yes” or “no” to depict whether the employee’s salary is within the range.

SELECT e.FIRST\_NAME, e.LAST\_NAME, j.JOB\_TITLE, e.SALARY ,j.MIN\_SALARY, j.MAX\_SALARY,

CASE WHEN e.salary BETWEEN j.MIN\_SALARY and j.MAX\_SALARY THEN 'yes'

ELSE 'no' END Within\_range

FROM EMPLOYEES e join JOBs j on e.JOB\_ID = j.JOB\_ID

order by e.LAST\_NAME;

