OPIM 3221 **Name: Daniel Lesh**

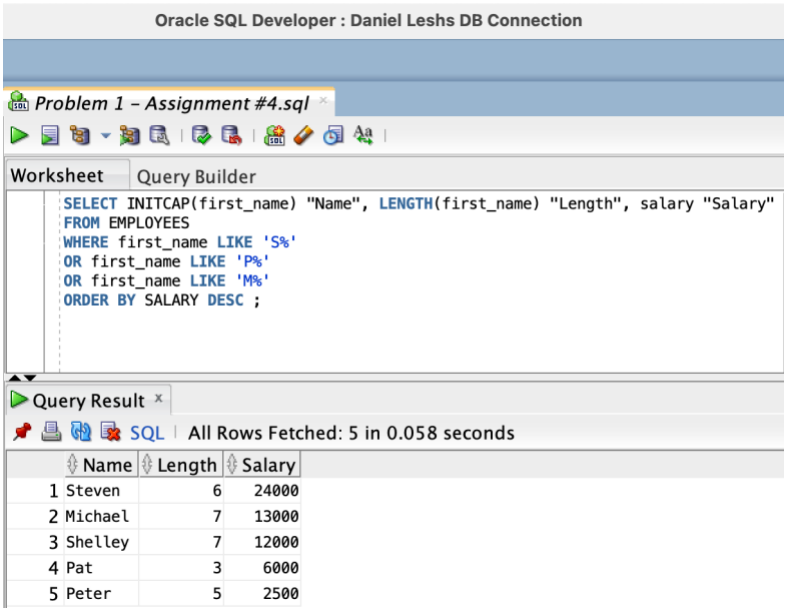
Assignment #4 - Single Row and Group Functions

Problem 1 - For the following problems screenshot your code and output (View)

Grading Rubric: **(Total 50 points)**

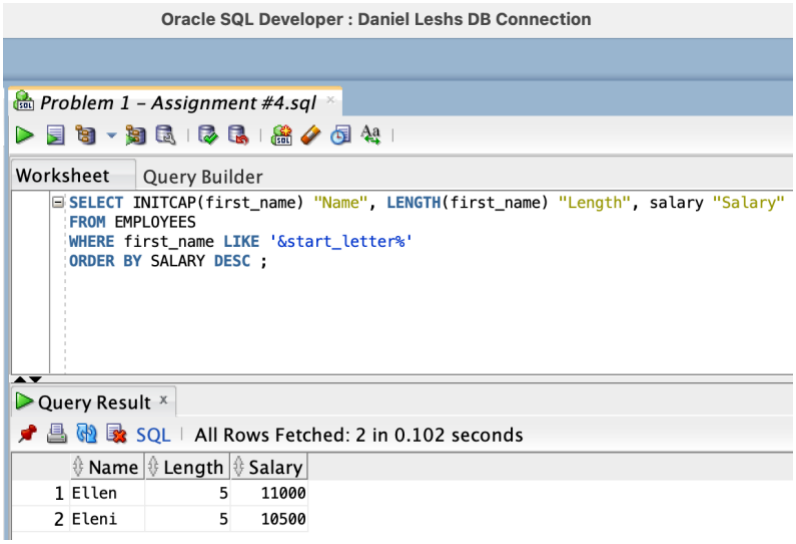
- Q1 – Q4 5 points for each question
  - 1 point for posting view and output correctly
  - 4 points for parts of the query (partial points awarded)
- Q5 – Q7 10 points for each question
  - 2 points for posting view and output correctly
  - 8 points for parts of the query (partial points awarded)

1. Write a query that displays the first name (with the first letter in uppercase and all the other letters in lowercase) and the length of the first name for all employees whose name starts with the letters “S” “P” or “M” salary. Give each column an appropriate label. Sort the results by the descending order of employees’ salary.

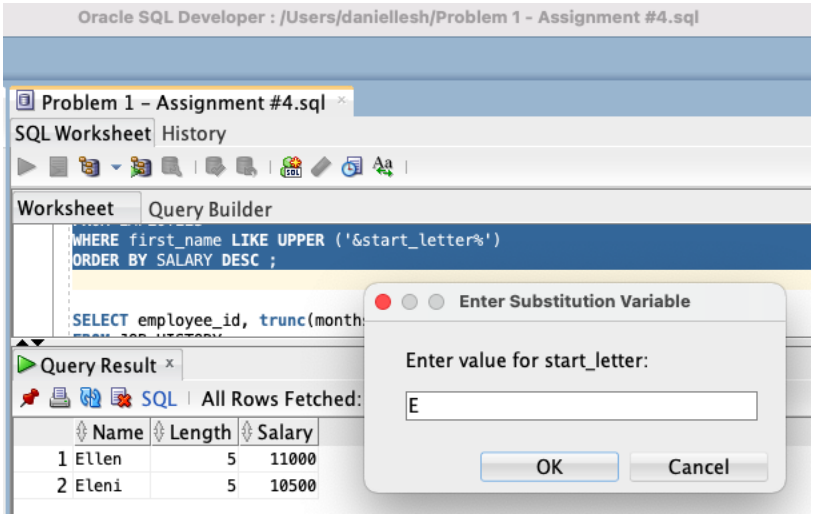


	Name	Length	Salary
1	Steven	6	24000
2	Michael	7	13000
3	Shelley	7	12000
4	Pat	3	6000
5	Peter	5	2500

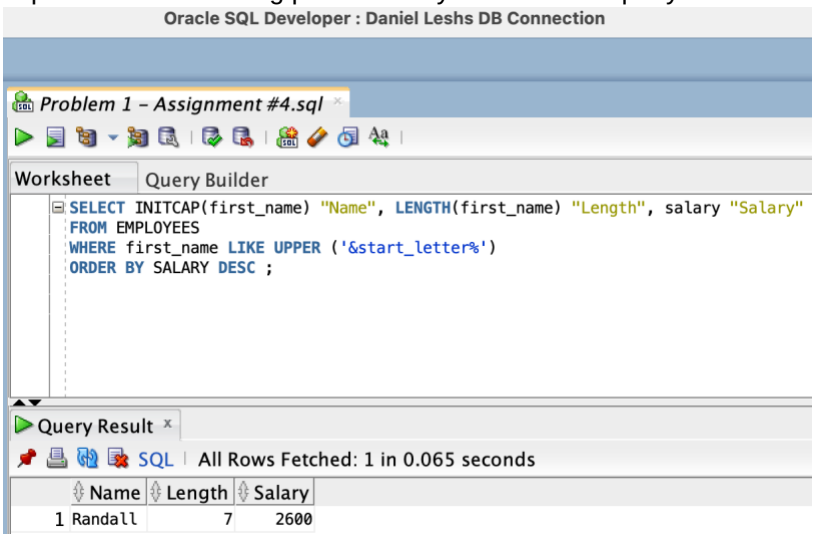
2. Rewrite the query so that the user is prompted to enter a letter that the first name starts with. For example, if the user enters “E” (capitalized) when prompted for a letter, then the output should show all employees whose first name starts with the letter “E.”

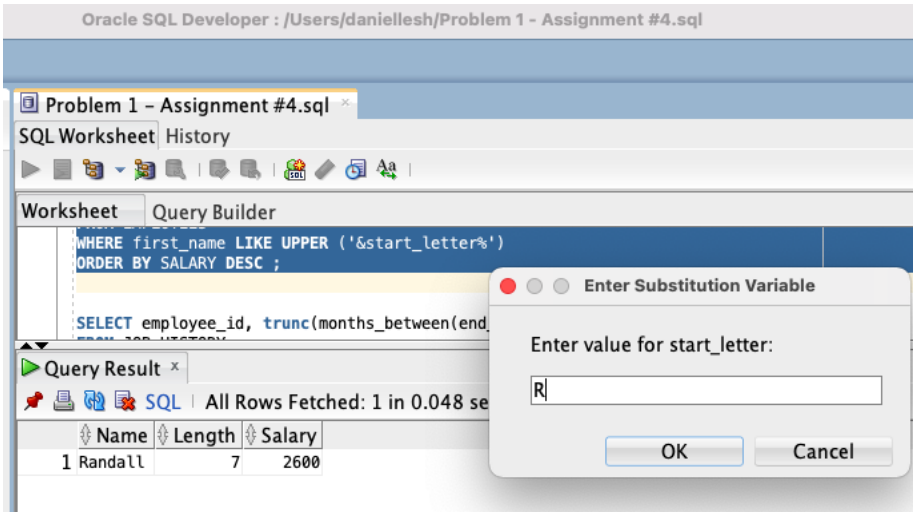


	Name	Length	Salary
1	Ellen	5	11000
2	Eleni	5	10500



3. Modify the query such that the case of the entered letter does not affect the output. The entered letter must be capitalized before being processed by the SELECT query.





4. From the job_history table, the HR department wishes to find out the number of full years an employee has worked. Exclude any employees who have worked for less than a year. Display the employee_id and the number of full years

Oracle SQL Developer : Daniel Leshs DB Connection

Problem 1 – Assignment #4.sql

Worksheet | Query Builder

```
SELECT employee_id, trunc(months_between(end_date,start_date)/12) "Years"
FROM JOB_HISTORY
WHERE trunc(months_between(end_date,start_date)/12) > 1
```

Query Result

SQL | All Rows Fetched: 6 in 0.088 seconds

	EMPLOYEE_ID	Years
1	102	5
2	101	4
3	101	3
4	201	3
5	200	5
6	200	4

5. Create a query to display the first name and email for all employees. Format the email to be 12 characters long, right-padded with the @ symbol. Label the column Email Address

Oracle SQL Developer : /Users/daniellesh/Problem 1 - Assignment #4.sq

Problem 1 – Assignment #4.sql

SQL Worksheet | History

Worksheet | Query Builder

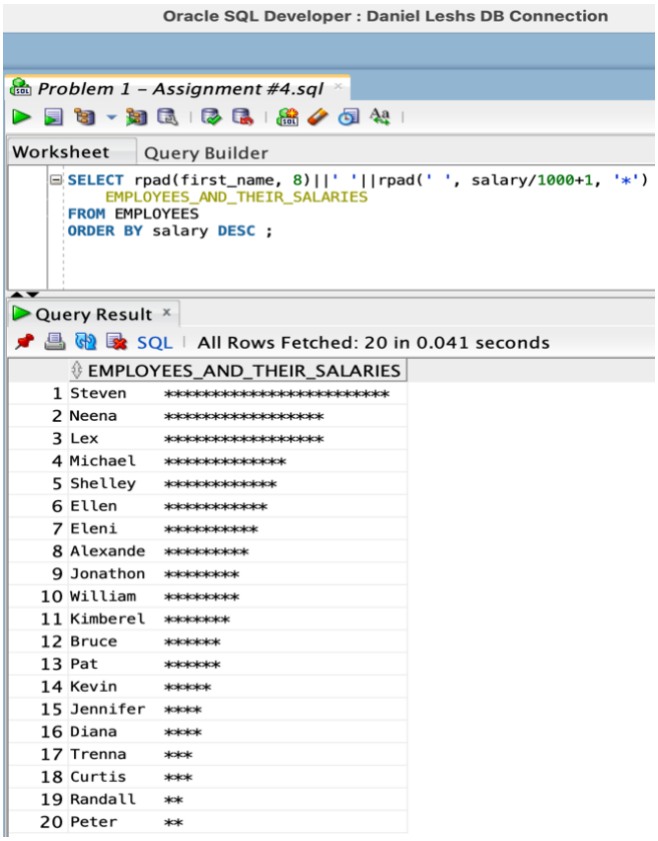
```
SELECT first_name, RPAD(email, 12, '@') "Email Address"
FROM EMPLOYEES ;
```

Query Result

SQL | All Rows Fetched: 20 in 0.043 seconds

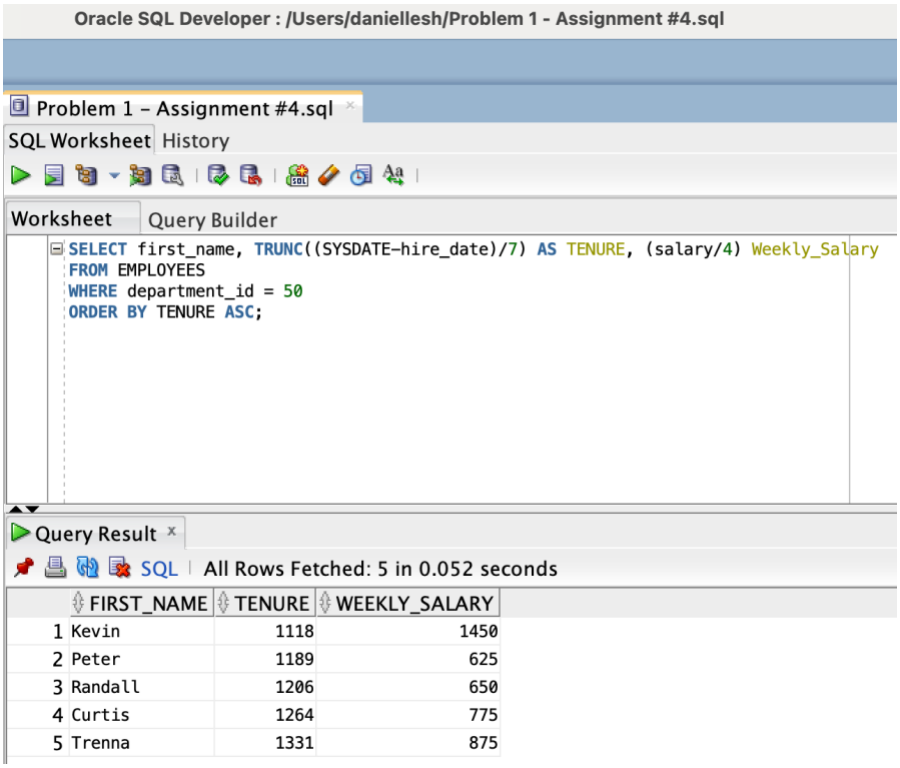
	FIRST_NAME	Email Address
1	Steven	SKING@@@@@@
2	Neena	NKOCHHAR@@@@
3	Lex	LDEHAAN@@@@@
4	Alexander	AHUNOLD@@@@@
5	Bruce	BERNST@@@@@@
6	Diana	DLORENTZ@@@@
7	Kevin	KMOURGOS@@@@
8	Trenna	TRAJS@@@@@@
9	Curtis	CDAVIES@@@@@
10	Randall	RMATOS@@@@@@
11	Peter	PVARGAS@@@@@
12	Eleni	EZLOTKEY@@@@
13	Ellen	EABEL@@@@@@
14	Jonathon	JTAYLOR@@@@@
15	Kimberely	KGRANT@@@@@@
16	Jennifer	JWHALEN@@@@@
17	Michael	MHARTSTE@@@@
18	Pat	PFAY@@@@@@@@
19	Shelley	SHIGGINS@@@@
20	William	WGIETZ@@@@@@

6. Create a query that displays the first eight characters of the employees' first names and indicates the amounts of their salaries with asterisks. Each asterisk signifies a thousand dollars. Sort the data in descending order of salary. Label the column EMPLOYEES_AND_THEIR_SALARIES.



7. Create a query to display the first name, the number of weeks employed, and weekly Salary (Salary divided by 4) for all employees in department 50. Label the number of weeks column as TENURE. Truncate the number of weeks value to 0 decimal places. Show the records in ascending order of the employee's tenure.

Note: The TENURE value will differ as it depends on the date on which you run the query.



Problem 2 - For the following problems screenshot your code and output (View) – exclude Q1 – Q3

Grading Rubric: **(Total 50 points)**

- Q1 – Q3 2 points for each question
- Q4 – Q9 5 points for each question
    - 1 point for posting view and output correctly
    - 4 points for parts of the query (partial points awarded)
- Q10 9 points
    - 2 points for posting view and output correctly
    - 7 points for parts of the query (this is not a trick question; it should have results)

Determine the validity of the following three statements. Circle either True or False.

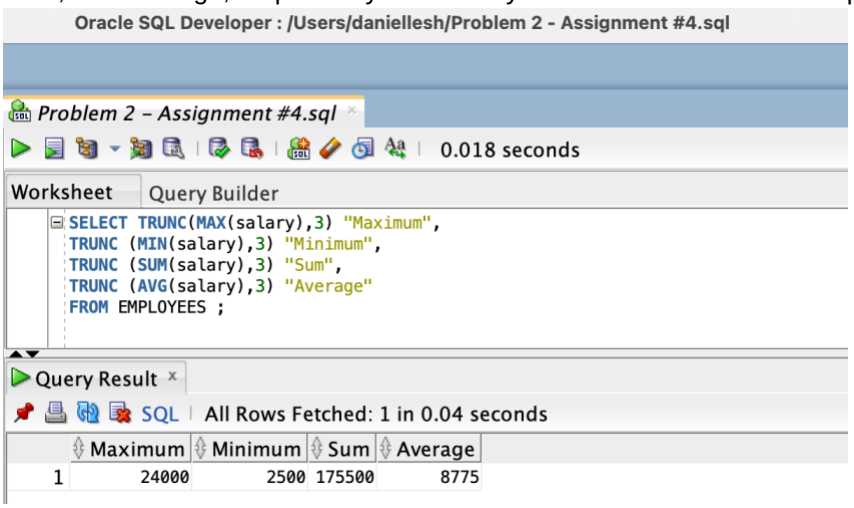
1. Group functions work across many rows to produce one result per group.
   True

2. Group functions include nulls in calculations.
   False

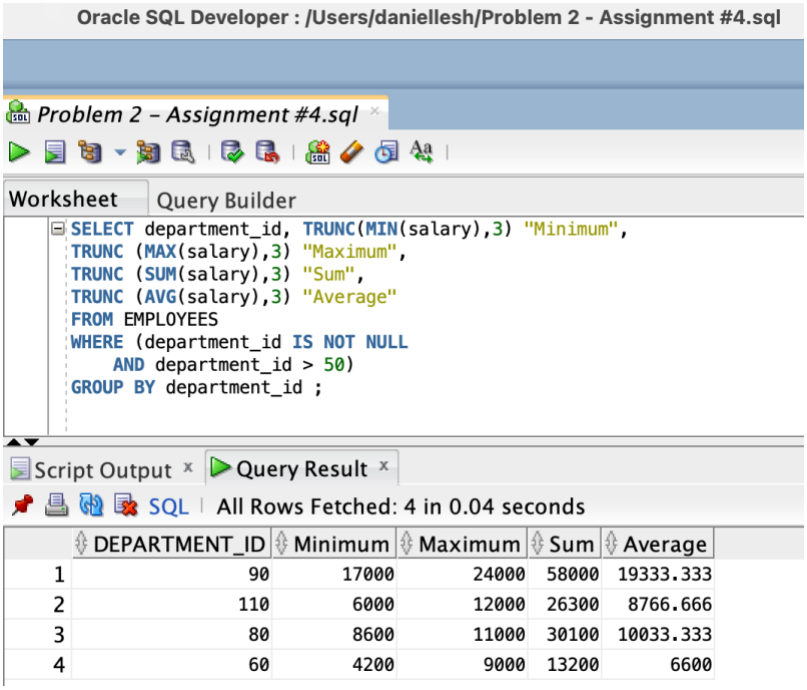
3. You can include a having clause that restricts records without a group by clause
   False

The HR department needs the following reports:

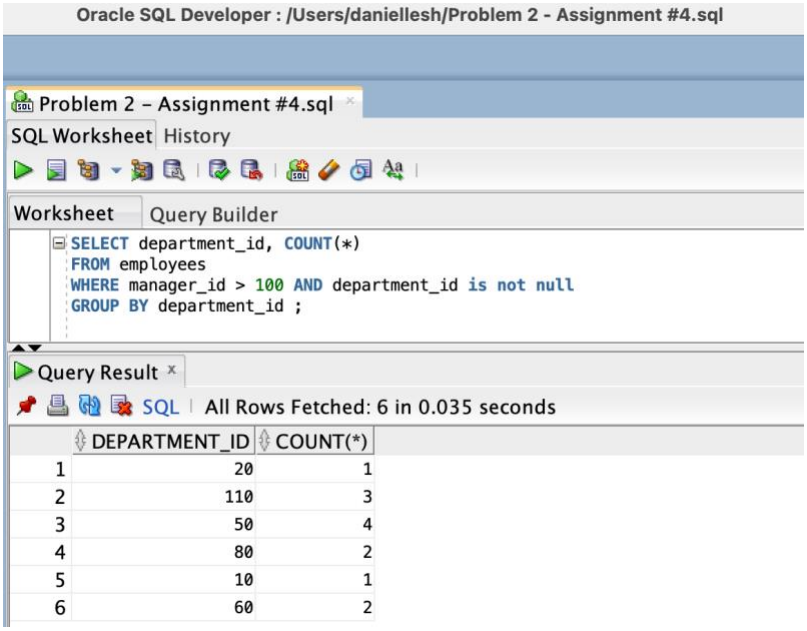
4. Find the highest, lowest, sum, and the average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Truncate your results to the 3 decimal places. Run the query.



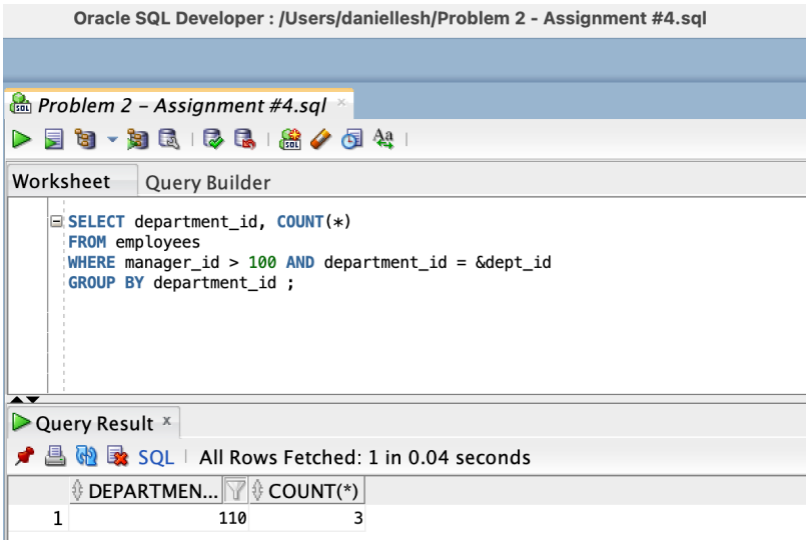
5. Modify the query in Question 4 to display the minimum, maximum, sum, and average salary for each department. Again, run the query. Note: We do not want to see the null department ids and any departments with ids less than or equal to 50.

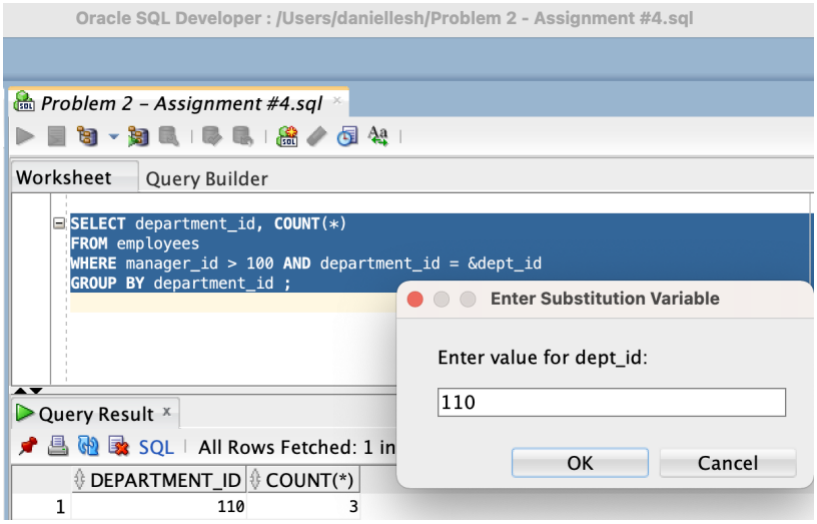


6. Write a query to display the number of people working in a given department having manager_id greater than 100. Note: we do not want to see any null values for department_id in the results.

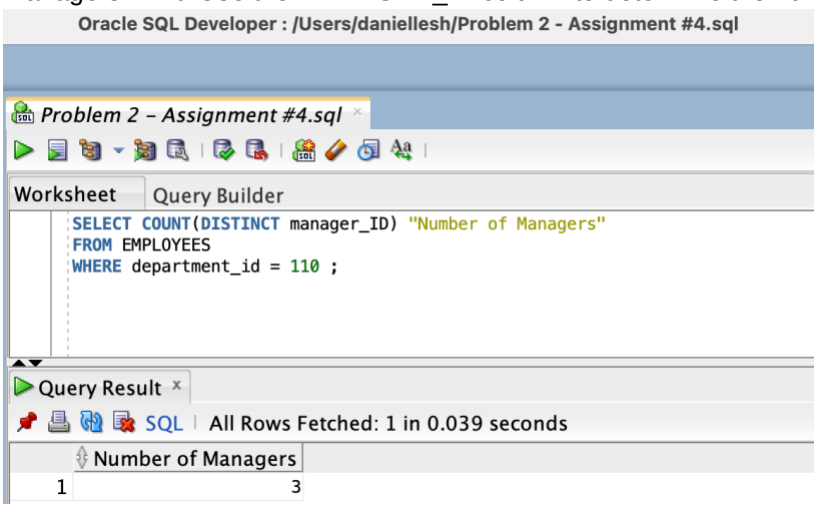


7. Generalize the query in Question 6 so that the user in the HR department is prompted for a department id. Run the query. Enter 110 when prompted. Screenshot the view for this question as well as the code.





8. Determine the numbers of managers without listing them in department number 110. Label the column Number of Managers. Hint: Use the MANAGER_ID column to determine the number of managers.



9. Find the difference between the highest and lowest salaries for each job id. Label the column Wage Gap

Oracle SQL Developer : /Users/daniellesh/Problem 2 - Assignment #4.sql

Problem 2 – Assignment #4.sql

Worksheet | Query Builder

```
SELECT job_id, MAX(salary) - MIN(salary) "Wage Gap"
FROM EMPLOYEES
GROUP BY job_id ;
```

Query Result

SQL | All Rows Fetched: 12 in 0.068 seconds

	JOB_ID	Wage Gap
1	AC_ACCOUNT	0
2	AC_MGR	0
3	AD_ASST	0
4	AD_PRES	0
5	AD_VP	0
6	IT_PROG	4800
7	MK_MAN	0
8	MK_REP	0
9	SA_MAN	0
10	SA_REP	4000
11	ST_CLERK	1000
12	ST_MAN	0

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

Oracle SQL Developer : /Users/daniellesh/Problem 2 - Assignment #4.sql

Problem 2 – Assignment #4.sql

Worksheet | Query Builder

```
SELECT manager_id, MAX(salary)
FROM EMPLOYEES
WHERE manager_id IS NOT NULL
GROUP BY manager_id
HAVING MAX (salary) > 9000
ORDER BY MAX (salary) DESC ;
```

Query Result

SQL | All Rows Fetched: 3 in 0.059 seconds

	MANAGER_ID	MAX(SALARY)
1	100	17000
2	101	12000
3	149	11000