Relational Databases with MySQL Week 4 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: Using a text editor of your choice, write the queries that accomplishes the objectives listed below. Take screenshots of the queries and results and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

Write 5 stored procedures for the employees database.

Write a description of what each stored procedure does and how to use it.

Procedures should use constructs you learned about from your research assignment and be more than just queries.

Screenshots:

1. This one will show you all employees with the date of birth entered:


```
bin — mysql -u root -p — 104×35
mysql> use employees;
Database changed
mysql> DELIMITER //
mysql> DROP procedure if exists find_birth_dates //
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE procedure find_birth_dates(IN lookup_birthdate DATE)
    -> BEGIN
    -> SELECT * FROM employees WHERE birth_date = lookup_birthdate;
    -> END //
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL find_birth_dates('1960-04-18');
+--------+------------+-------------+-----------------+--------+------------+
| emp_no | birth_date | first_name  | last_name       | gender | hire_date  |
+--------+------------+-------------+-----------------+--------+------------+
|  10359 | 1960-04-18 | Heekeun     | Sambasivam      | F      | 1991-05-09 |
|  11726 | 1960-04-18 | Eldridge    | Reghbati        | M      | 1990-05-29 |
|  14078 | 1960-04-18 | Frederique  | Chimia          | M      | 1987-11-01 |
|  16495 | 1960-04-18 | Collette    | Czaja           | F      | 1994-04-21 |
|  20748 | 1960-04-18 | Janalee     | Baaz            | F      | 1995-04-22 |
|  22026 | 1960-04-18 | Abdulla     | Thorelli        | M      | 1990-03-13 |
|  25896 | 1960-04-18 | Heeju       | Caine           | F      | 1992-01-16 |
|  28866 | 1960-04-18 | Remko       | Farris          | F      | 1986-02-03 |
|  29898 | 1960-04-18 | Subhankar   | Hofting         | M      | 1987-04-01 |
|  36723 | 1960-04-18 | Atreye      | Millington      | M      | 1987-03-05 |
|  54345 | 1960-04-18 | Munir       | Stille          | M      | 1993-08-01 |
|  71371 | 1960-04-18 | Rosalie     | Kemmerer        | M      | 1994-12-20 |
|  73650 | 1960-04-18 | Emdad       | Castellani      | M      | 1986-10-02 |
|  74078 | 1960-04-18 | Adamantios  | Bratten         | M      | 1987-10-11 |
|  88487 | 1960-04-18 | Sumali      | Bain            | M      | 1988-07-07 |
|  89697 | 1960-04-18 | Jeane       | Demian          | F      | 1985-12-08 |
|  93365 | 1960-04-18 | Adhemar     | Lindqvist       | M      | 1993-04-14 |
|  94927 | 1960-04-18 | Torsten     | Stemann         | M      | 1986-11-22 |
```

2. This one will show you all employees with matching salary and hire date input:

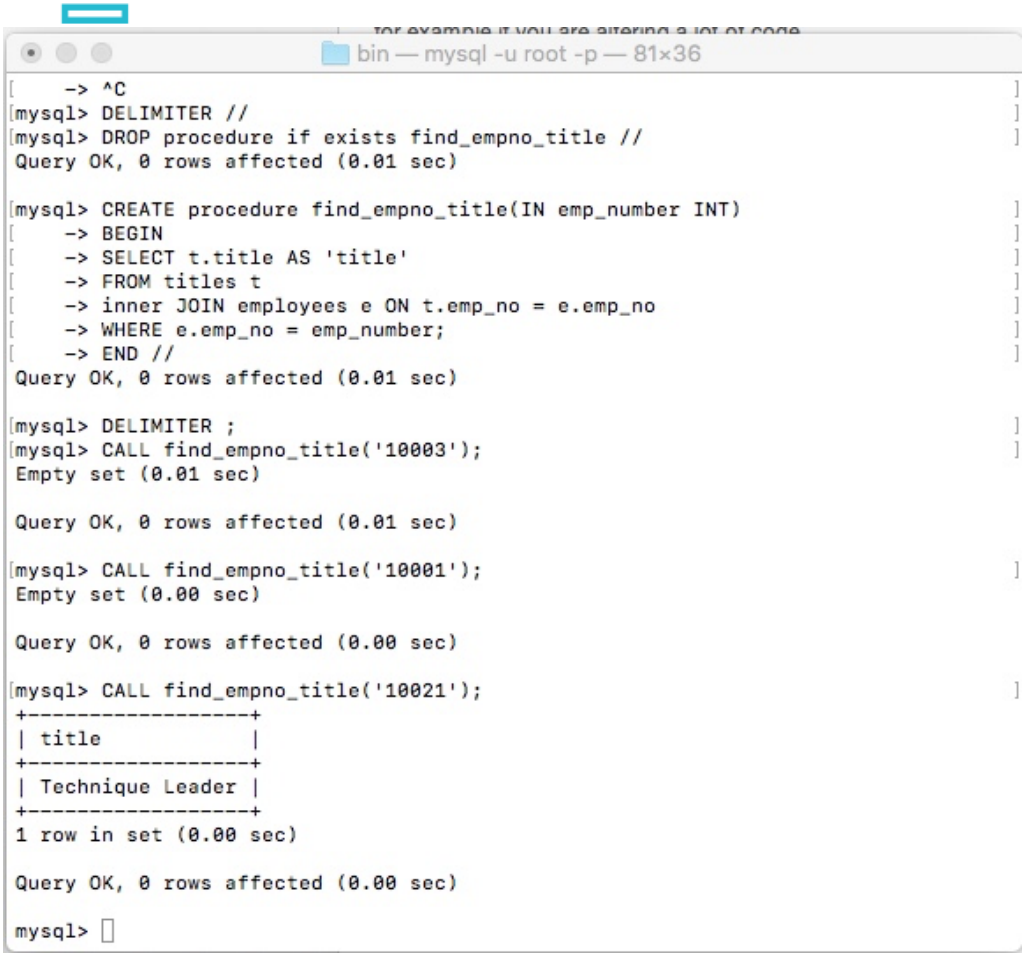
```
bin — mysql -u root -p — 96×27
mysql> DELIMITER //
mysql> DROP procedure if exists salaries_and_hiredates //
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE procedure salaries_and_hiredates(IN salary_given INT, IN date_hired DATE)
    -> BEGIN
    -> SELECT concat(e.first_name, e.last_name) 'Name', e.emp_no 'Emp. No', e.hire_date 'Hire Da
te' FROM employees e
    -> inner JOIN salaries s ON s.emp_no = e.emp_no
    -> WHERE e.hire_date = date_hired AND s.salary >= salary_given;
    -> END //
Query OK, 0 rows affected (0.02 sec)

mysql> DELIMITER ;
mysql> CALL salariessalaries_and_hiredates(60000, '2000-01-01');
ERROR 1305 (42000): PROCEDURE employees.salariessalaries_and_hiredates does not exist
mysql> CALL salaries_and_hiredates(60000, '2000-01-01');
+--------------------+---------+------------+
| Name               | Emp. No | Hire Date  |
+--------------------+---------+------------+
| MariangiolaBoreale |  108201 | 2000-01-01 |
+--------------------+---------+------------+
1 row in set (0.65 sec)

Query OK, 0 rows affected (0.65 sec)

mysql>
```

PROMINEO TECH

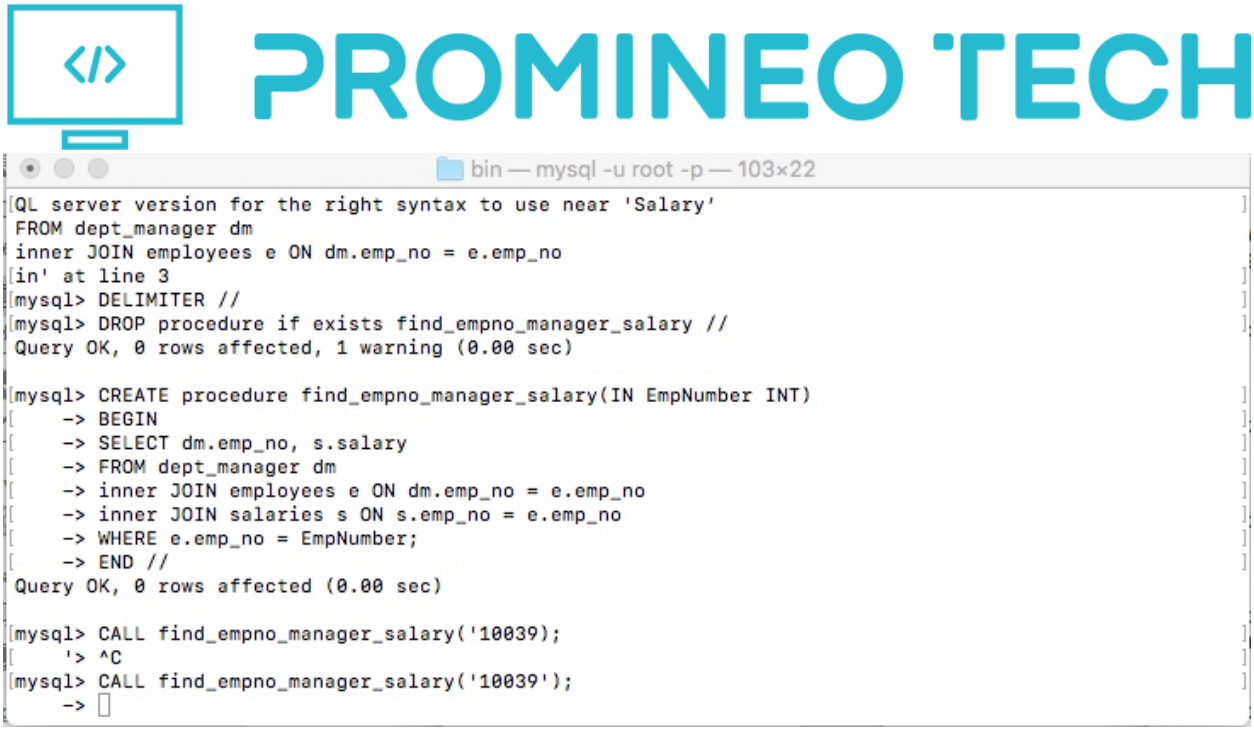
3. This one will show you department salary averages when you enter department number:

```
bin — mysql -u root -p — 98×28
inner JOIN employees e ON e.em' at line 4
mysql> DELIMITER //
mysql> DROP procedure if exists department_salaries //
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> CREATE procedure department_salaries(IN dept_number CHAR(4))
    -> BEGIN
    -> SELECT avg(s.salary), d.dept_name
    -> FROM departments d
    -> inner JOIN dept_emp de ON de.dept_no = d.dept_no
    -> inner JOIN employees e ON e.emp_no = de.emp_no
    -> inner JOIN salaries s ON s.emp_no = e.emp_no
    -> WHERE d.dept_no = dept_number;
    -> END //
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL department_salaries('d008');
+---------------+-----------+
| avg(s.salary) | dept_name |
+---------------+-----------+
|    59664.4027 | Research  |
+---------------+-----------+
1 row in set (0.77 sec)

Query OK, 0 rows affected (0.77 sec)

mysql>
```

4. This one will show you the title of an employee when you enter an employee number:



5. This one will show you the manager salary when you enter an employee number (query was fine so not sure why result is not populating):



URL to GitHub Repository: https://github.com/danielleyokley/MYSQLWEEK4.git