Relational Databases with MySQL Week 8 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 to the repository. Additionally, push an .sql file with all your queries to the same 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 queries to address the following business needs.

1. I want to know how many employees with each title were born after 1965-01-01.

```
SELECT titles.title, COUNT(titles.emp_no)
FROM titles
INNER JOIN employees ON titles.emp_no = employees.emp_no
WHERE employees.birth_date > '1965-01-01'
GROUP BY titles.title;
```

2. I want to know the average salary per title.

```
SELECT titles.title, AVG(salaries.salary)
FROM titles
INNER JOIN salaries ON salaries.emp_no = titles.emp_no
GROUP BY titles.title;
```

3. How much money was spent on salary for the marketing department between the years 1990 and 1992?

```
SELECT dept_emp.dept_no, department.dept_name, SUM(salaries.salary) AS
    'Total_salary_cost_1990-1992'
FROM salaries
INNER JOIN dept_emp ON dept_emp.emp_no = salaries.emp_no
LEFT JOIN departments ON departments.dept_no = dep_emp.dept_no
WHERE salaries.from_date >= 1990-01-01 AND salaries.to_date <= 1992-12-31
GROUP BY dept_emp.dept_no;
```

**Note that this is probably going to return inaccurate data because there would have been salary figures that were initiated in the year before 1990-01-01 and ran into the year 1990. Likewise, any salary amount that was initiated in the year 1992 but spilled into 1993 will have too much money accounted for in this total. The way their data is structured, (I THINK) it's difficult to figure out exact amounts (in SQL) for what was actually spent in the given years. I may be wrong, so please let me know if I am. :) I would think that normally a company would have a table with department budgets indicating much went to salary for the fiscal year. That would be the better data to use for this.

Screenshots of Queries and Screenshots of Query Results (only include the last 20 rows):

I put them together, on the next page.

Github repository:

https://github.com/lindsey406/week8homework.git

```
mysql> SELECT titles.title, AVG(salaries.salary)
    -> FROM titles
    -> INNER JOIN salaries ON salaries.emp_no = titles.emp_no
    -> GROUP BY titles.title;
+--------------------+----------------------+
| title              | AVG(salaries.salary) |
+--------------------+----------------------+
| Senior Engineer    |           60543.2191 |
| Staff              |           69309.1023 |
| Engineer           |           59508.0397 |
| Senior Staff       |           70470.8353 |
| Assistant Engineer |           59304.9863 |
| Technique Leader   |           59294.3742 |
| Manager            |           66924.2706 |
+--------------------+----------------------+
7 rows in set (3.58 sec)
```

```
mysql> SELECT departments.dept_name, SUM(salaries.salary) AS 'Total_salary_cost_1990-1992'
    -> FROM salaries
    -> INNER JOIN dept_emp ON dept_emp.emp_no = salaries.emp_no
    -> LEFT JOIN departments ON departments.dept_no = dept_emp.dept_no
    -> WHERE departments.dept_name = 'Marketing' AND
    -> salaries.from_date >= '1990-01-01' AND salaries.to_date <= '1992.12.31'
    -> GROUP BY dept_emp.dept_no;
+-----------+----------------------------+
| dept_name | Total_salary_cost_1990-1992 |
+-----------+----------------------------+
| Marketing |                 1096824732 |
+-----------+----------------------------+
1 row in set, 1 warning (0.13 sec)
```

```
mysql> SELECT titles.title, COUNT(titles.emp_no)
    -> FROM titles
    -> INNER JOIN employees ON titles.emp_no = employees.emp_no
    -> WHERE employees.birth_date > '1965-01-01'
    -> GROUP BY titles.title;
+--------------------+----------------------+
| title              | COUNT(titles.emp_no) |
+--------------------+----------------------+
| Senior Staff       |                  612 |
| Staff              |                  703 |
| Technique Leader   |                   95 |
| Senior Engineer    |                  589 |
| Engineer           |                  657 |
| Assistant Engineer |                   97 |
+--------------------+----------------------+
6 rows in set (0.11 sec)
```

-