

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.
2. I want to know the average salary per title.
3. How much money was spent on salary for the marketing department between the years 1990 and 1992?

URL to GitHub Repository:

<https://github.com/ladymanj/Week8CodingAssignment.git>

Screenshots of Queries:

Screenshot from .sql file:

```
SELECT t.title AS "Title", COUNT(*) AS "# of Employees Born After Jan 1, 1965" FROM employees e
INNER JOIN titles t ON e.emp_no = t.emp_no
WHERE e.birth_date > '1965-01-01'
GROUP BY t.title
ORDER BY t.title;
```

```
SELECT t.title AS "Title", AVG(s.salary) AS "Average Salary" FROM salaries s
INNER JOIN titles t
ON s.emp_no = t.emp_no
GROUP BY t.title
ORDER BY t.title;
```

```
SELECT d.dept_name AS "Department Name", SUM(s.salary) AS "Total of Salaries" FROM salaries s
INNER JOIN dept_emp de ON s.emp_no = de.emp_no
INNER JOIN departments d ON de.dept_no = d.dept_no
WHERE s.from_date >= '1990-01-01' AND s.to_date <= '1992-12-31'
GROUP BY d.dept_name
HAVING d.dept_name = 'Marketing';
```

Screenshots from CMD:

1.

```
mysql> SELECT t.title AS "Title", COUNT(*) AS "# of Employees Born After Jan 1, 1965" FROM employees e
-> INNER JOIN titles t ON e.emp_no = t.emp_no
-> WHERE e.birth_date > '1965-01-01'
-> GROUP BY t.title
-> ORDER BY t.title;
```
2.

```
mysql> SELECT t.title AS "Title", AVG(s.salary) AS "Average Salary" FROM salaries s
-> INNER JOIN titles t
-> ON s.emp_no = t.emp_no
-> GROUP BY t.title
-> ORDER BY t.title;
```
3.

```
mysql> SELECT d.dept_name AS "Department Name", SUM(s.salary) AS "Total of Salaries" FROM salaries s
-> INNER JOIN dept_emp de ON s.emp_no = de.emp_no
-> INNER JOIN departments d ON de.dept_no = d.dept_no
-> WHERE s.from_date >= '1990-01-01' AND s.to_date <= '1992-12-31'
-> GROUP BY d.dept_name
-> HAVING d.dept_name = 'Marketing';
```

// NOTE FOR ABOVE: I based the query that addressed the third business need on these specifications:

- “between the years 1990 and 1992” – inclusive; from the very beginning of 1990 to the very end of 1992
- “spent on salary” – I decided to only sum the salaries from the yearly salary periods that began starting anytime after the beginning of 1990 AND ended before the end of 1992. I wanted to be inclusive for partial salaries from salary periods that overlapped into 1993 (and likewise for salary periods that started in 1989), but I didn’t want to make assumptions on salary disbursement (“paycheck periods”), so I ultimately decided to exclude those from the query.

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

1.

Title	# of Employees Born After Jan 1, 1965
Assistant Engineer	97
Engineer	657
Senior Engineer	589
Senior Staff	612
Staff	703
Technique Leader	95

6 rows in set (0.19 sec)

2.

Title	Average Salary
Assistant Engineer	59304.9863
Engineer	59508.0751
Manager	66924.2706
Senior Engineer	60543.2191
Senior Staff	70470.5013
Staff	69308.7124
Technique Leader	59294.3742

7 rows in set (4.58 sec)

3.

Department Name	Total of Salaries
Marketing	1096824732

1 row in set (1.77 sec)

*****As an experiment, I added a slight change to the third query to make the sum*****
*****of the salaries easier to read, using knowledge from completing the research*****
*****assignment*****

Query:

```
mysql> SELECT d.dept_name AS "Department Name", FORMAT(SUM(s.salary), 0) AS "Total of Salaries" FROM salaries s  
-> INNER JOIN dept_emp de ON s.emp_no = de.emp_no  
-> INNER JOIN departments d ON de.dept_no = d.dept_no  
-> WHERE s.from_date >= '1990-01-01' AND s.to_date <= '1992-12-31'  
-> GROUP BY d.dept_name  
-> HAVING d.dept_name = 'Marketing';
```

// NOTE FOR ABOVE: I used the FORMAT function to make the final sum easier to read

Query Result:

```
+-----+-----+  
| Department Name | Total of Salaries |  
+-----+-----+  
| Marketing      | 1,096,824,732    |  
+-----+-----+  
1 row in set (1.76 sec)
```