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. Lastly, in the Learning Management System, click the “Add Submission” button and paste the URL to your GitHub repository.

**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:**

**This proc takes a year and returns the average salery from the saleries table for that year.**

**mysql> delimiter $$ ;**

**mysql> create procedure avg\_annual\_sal(in input\_year int, out avg\_sal int) begin select avg(salary) into avg\_sal from salaries where year(from\_date) = input\_year and year(to\_date) = input\_year; end $$**

**Query OK, 0 rows affected (0.00 sec)**

**mysql> delimiter ; $$**

**mysql> call avg\_annual\_sal(1990, @average);**

**Query OK, 1 row affected (0.00 sec)**

**mysql> select@average;**

**+----------+**

**| @average |**

**+----------+**

**| 57751 |**

**+----------+**

**1 row in set (0.00 sec)**

**This proc takes a dept\_no and returns the catagory of size depending on how many employees are in the that department.**

**mysql> delimiter $$ ;**

**mysql> create procedure department\_size(in input\_dept\_no varchar(8)) begin declare size varchar(15); declare emp\_count int; select count(\*) into emp\_count from dept\_emp where dept\_no = input\_dept\_no; if emp\_count < 100 then set size = 'small'; elseif emp\_count < 500 then set size = 'medium'; else set size = 'large'; end if; select emp\_count, size; end $$**

**Query OK, 0 rows affected (0.00 sec)**

**mysql> delimiter ; $$**

**mysql> call department\_size('d008');**

**+-----------+-------+**

**| emp\_count | size |**

**+-----------+-------+**

**| 21126 | large |**

**+-----------+-------+**

**1 row in set (0.00 sec)**

**Query OK, 0 rows affected (0.00 sec)**

**URL to GitHub Repository:**

[**https://github.com/mikefranzen/mysql\_week4**](https://github.com/mikefranzen/mysql_week4)