# Relational Databases with MySQL Week 7 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:**

Using the employees database you installed, write SQL queries that do the following (the SQL queries you write are what you will turn in for your homework):

1. Show all employees who were born before 1965-01-01

SELECT \* FROM employees WHERE birth\_date < "1965-01-01";

2. Show all employees who are female and were hired after 1990

SELECT \* FROM employees WHERE gender = "F" AND hire\_date > "1990-01-01";

3. Show the first and last name of the first 50 employees whose last name starts with F

SELECT first\_name, last\_name FROM employees WHERE first\_name LIKE "f%";

4. Insert 3 new employees into the employees table. There emp\_no should be 100, 101, and 102. You can choose the rest of the data.

INSERT INTO employees VALUES (100, "2000-01-01", "Tom","Smith","M", "2010-01-01");

INSERT INTO employees VALUES (101, "2000-01-01", "John", "Mackey", "M", "2015-01-01");

INSERT INTO employees VALUES (102, "1998-01-01", "Sandra", "Barbara", "F", "2008-01-01");

5. Change the employee's first name to Bob for the employee with the emp\_no of 10023.

UPDATE employees SET first\_name = "Bob" WHERE emp\_no = 10023;

6. Change all employees hire dates to 2002-01-01 whose first or last names start with P.

UPDATE employees Set hire\_date = "2002-01-01" WHERE first\_name Like "P%" or last\_name Like "P%";

7. Delete all employees who have an emp\_no less than 10000

DELETE FROM employees WHERE emp\_no < 10000;

8. Delete all employee who have an emp\_no of 10048, 10099, 10234, and 20089.

Delete FROM employees WHERE emp\_no = 10048 or emp\_no = 10099 or emp\_no = 10234 or emp\_no = 20089;

**Screenshots of Queries:**

**A picture containing text

Description automatically generated**

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

**Graphical user interface, text, application

Description automatically generated**

**^^^^^Already ran the query a few times, so it says that no rows were affected 331 & 332 but verified with a select query.**

**Below is all results in order from query 1-8**

**1.** Table

Description automatically generated **2.** **Table

Description automatically generated**

**3.** **Graphical user interface, application, table

Description automatically generated** 4. Graphical user interface, application, table

Description automatically generated

5. Graphical user interface, application

Description automatically generated

6. Graphical user interface, table

Description automatically generated

7. 

null when emp\_no < 10000 was selected through query.

8. 

Null when emp\_no 10048, 10099, 10234, or 20089 is selected through query.

**URL to GitHub Repository:**

**https://github.com/mbrooks212/mysqlWeek7**