

HW2

36-650 – Statistical Computing

Released on Week 3 – Fall 2021

You must submit **your own** HW as series of SQL scripts (and screenshots) stored in a folder on your GitHub repository.

This week’s homework is a little different. Here you will be working with **SQL**, specifically with the variant **postgres**. You will do your work “remotely” in a **postgres** terminal and cut-and-paste your answers into SQL files (files you create that end with **.sql** extension).

So, check the answer for each question in a separate SQL file. Name the SQL file with the question number (e.g. **q1.sql**)

If you were asked to submit a copy of the output, take a screenshot and check in the screenshot into your GitHub repository and name it with the question number (e.g. **q7.png**)

If the output has many rows, it is sufficient to take a screenshot of the first five, unless otherwise instructed.

Check in all your scripts and screenshots under a folder named **HW2** in your GitHub repository. Make sure your GitHub repository is publicly viewable.

On Canvas, only submit a URL to your GitHub repository.

Not following the submission guidelines may result in grading penalties.

Question 1

(10 points)

Create a table dubbed **rdata** that has five columns: **id** (type **serial primary key**), **a** and **b** (consisting of strings that should be no more than five characters), **moment** (which is a date), and **x** (which is a number that should have no more than five digits total and two to the right of the decimal point).

Question 2

(10 points)

Delete the table and create it again, with certain constraints: **a** and **b** cannot be null and must be unique; **moment** should have a default value of **2020-01-01** (); and **x** should be larger than zero.

Question 3

(10 points)

Use **\d** in combination with the name of your table to display its properties. Copy and paste all output here.

Question 4

(10 points)

Insert three rows of data into your table. Do this without explicitly referring to the column `id` or `moment`. Display your table using the command `select * from rdata`. You should see a default date in your `moment` column!

Question 5

(10 points)

Attempt to add another row of data with a negative value for `x`. Show what happens. (If the row is added... that's bad. It means you don't have the constraint `x > 0` defined.) Afterwards, add a valid row of data, and show the table.

Question 6

(10 points)

Change the table `rdata` so as to change the data in `moment` so that two of the dates are in March 2020, while the other two are not in March 2020. Use `where` as shown in the notes, and utilize a logical "or" to update two of the dates to the same date in March at the same time. Show your updated table. (Note that the rows may be rearranged. This is OK.)

Question 7

(10 points)

Add a new column to `rdata` dubbed `y`, and let it be of `boolean` type with default value `false`. Display your updated table.

Question 8

(10 points)

Remove the row of your table with `id` value 2. Display your updated table.

Question 9

(10 points)

Rename the column `moment` to have the name `date`. Display your updated table.