

Homework #3: SQL

Thought of the week: If Google can't help you with the homework, believe me "GOD is testing you".

Overview

This homework is worth 60 points. It is due on Sunday, November 11, at 11:55 p.m. Late submissions will be penalized 20% during a 3-day grace period up until Wednesday, November 14, 11:55 p.m. After that time, no late work will be accepted. Your homework submission should be via the link found in the Homework section of the week where you got this file.

This homework will give you hands-on practice in working with SQL (Structured Query Language.) In this homework you will create a database and populate it using scripts provided. The database you create will then be used for various queries/problems in this homework.

Objectives

- Become familiar with SQL language & syntax for SELECT queries, DDL and DML
- Become familiar with a tool of your choice for building and submitting queries (whether command mode or GUI.)
- Successfully run the scripts necessary to create a sample database consisting of 8 tables, verify that your database is correctly built.
- Use SQL your database to answer the assigned problems.

Pre-Processing

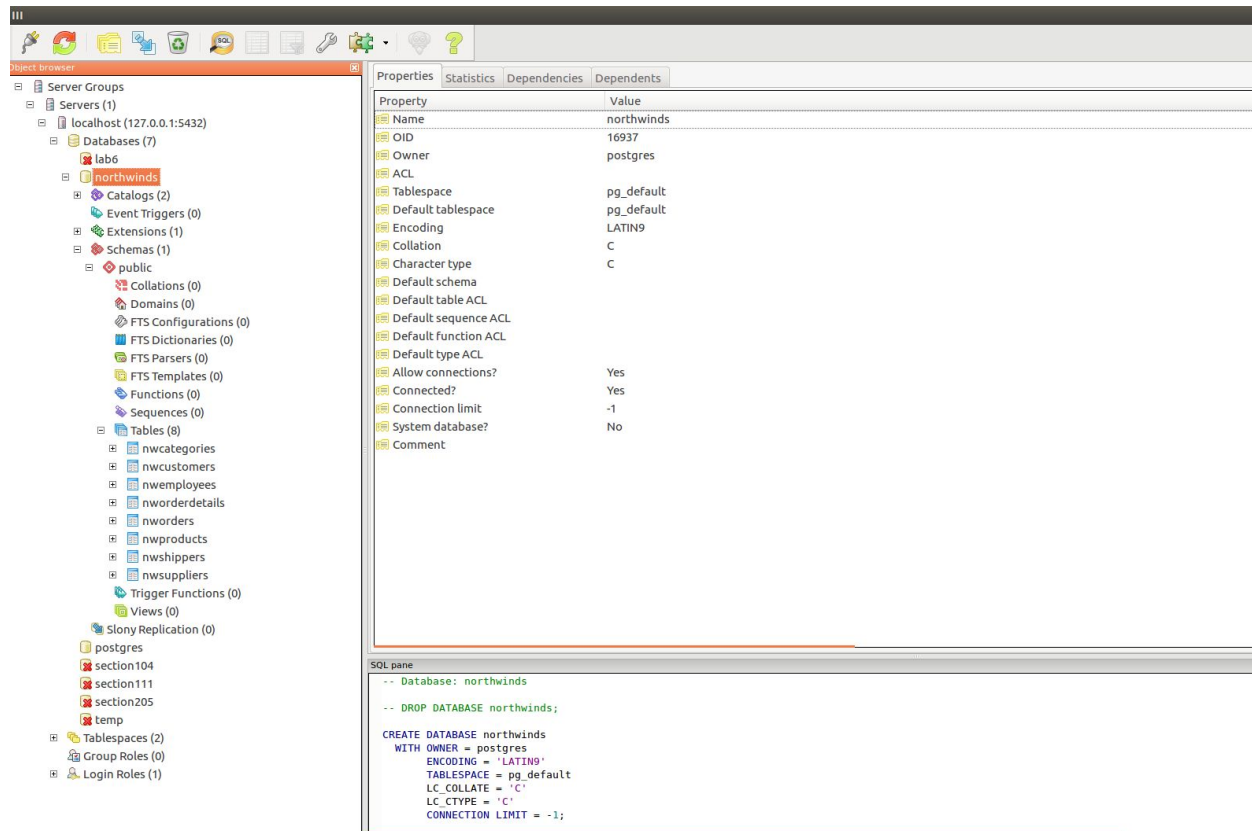
Step1: Download and Install PostgreSQL/PostgresAdmin

Please follow the installation guide, if you haven't installed it

https://moodle.cs.colorado.edu/pluginfile.php/112893/mod_resource/content/1/Cla ss_%20PostgreSQL%20Installation.pdf

Homework #3: SQL

In this homework, if you want to use console or the Admin tool, You are free to use either one. The guide is for your VM or Ubuntu system. If you have any other operating system, please download the respective installer for it.



Step2: Creating the tables

Download a Script folder from Moodle under HW3. There are 8 scripts files to create 8 tables and insert values inside these tables. Before creating these tables, you have to create on database. Please read the below instruction carefully.

Before you can create your database, you need to make sure that your instance of PostgreSQL is running.

To check your instance is running or not, Open a new terminal and type “**service postgresql status**”, if the status is **Active** then your postgresQL is running. If the status is **inactive** or **dead**, then type “**service postgresql start**” and hit enter and type your system password when the window is pop-up.

Homework #3: SQL

Follow this writeup to start your postgresSQL:
https://moodle.cs.colorado.edu/pluginfile.php/112893/mod_resource/content/1/Cla ss_%20PostgreSQL%20Installation.pdf

Do the following:

1. **Create a Database:** create database **company** WITH ENCODING 'LATIN9' LC_COLLATE = 'C' LC_CTYPE='C' TEMPLATE=template0;
2. Open a new terminal and go to the directory where you download your script files from Moodle. Go inside this directory and use **ls** command and it will list out all the files. You will get below files in the folder.
 - a. categories.sql
 - b. employees.sql
 - c. orders.sql
 - d. order_details.sql
 - e. customers.sql
 - f. products.sql
 - g. suppliers.sql
 - h. shipper.sql
3. Import all the tables from your SQL files inside your database named **“company”**.
 - a. To import, please type in your terminal **sudo su postgres** and it will ask your system password and then type

psql company<categories.sql

Do the same thing for the rest of the sql files.

Note: If you get any **error** while importing the files regarding unsupported character set or UTF8, then please go to **/etc/postgresql/9.5/main/**

“9.5” is my postgresql version, likewise you may have different version. So, please follow the version and find **postgresql.conf** file. First, check that the file is writeable format or not. If not, then convert it into writeable format which you learned in your script Lab class.

Homework #3: SQL

Go inside this file and search for “**client_encoding**” and comment it out if it’s not and restart your postgres server which is given in your “**Creating the table**” section. Drop the database “**company**” and start from Step 1.

The company database consists of 8 tables. The HW3_scripts folder contains 9 script files, one for creating each table, and one for generating this one. Please run **psql company<verify.sql** likewise you run your script files. It will show you below results:

table_schema	table_name	count_rows
public	orderdetails	2155
public	orders	830
public	customers	87
public	products	77
public	suppliers	29
public	employees	9
public	categories	8
public	shippers	3

(8 rows)

Note: Please make sure you are getting the same count of rows as above pictures. If you are not getting then drop the table and insert it. If you are getting the errors while inserting the values then follow the guide patiently.

Preparing Your Homework Submission

Your results for this homework assignment should be captured in a document (such as a .txt file, MS Word or similar tool.) Use the link found in the Homework Assignment section in the Moodle site to submit your work for grading. If you are doing PAIR PROGRAMMING on this assignment, please be sure to identify the name of your programming partner on your submission. **You must EACH submit your own results document for this homework.**

You must turn in BOTH your SQL and your ANSWER SET (One file with only SQL queries. The other with queries and answer set). Save your SQL Query files as HW3.sql and change permissions to make it executable.

Homework #3: SQL

For each problem where a multi-row answer set is created, the number of rows you should expect in your answer set is listed. Some queries will produce NO answer set.

Query Problems

For this homework you must create and execute queries against the company database to fulfill the requirements listed below. For each query requirement the number of rows to expect in your answer set is listed in parentheses.

1. Create an alphabetical listing (First Name, Last Name) of all employees not living in the USA who have been employed with company for at least 5 years as of today. (4)
2. Prepare a Reorder List for products currently in stock. (Products in stock have at least one unit in inventory.) Show Product ID, Name, Quantity in Stock and Unit Price for products whose inventory level is at or below the reorder level. (17)
3. What is the name and unit price of the most and least expensive product sold by company? Use a sub-query. (2)
4. Create a list of the products in stock which have an inventory value (the number of units in stock multiplied by the unit price) over \$1000. Show the answer set columns as Product ID, Product Name and “Total Inventory Value” in order of descending inventory value (highest to lowest.) (25)
5. List the country and a count of Orders for all the orders that shipped outside the Germany during October 2013 in descending country sequence. (12)
6. List the CustomerID and ShipName of the customers who have more than or equal to 10 orders. (37)
7. Create a Supplier Inventory report (by Supplier ID) showing the total value of their inventory in stock. (“value of inventory” = UnitsInStock * UnitPrice.) List only those suppliers from whom company receives more than or equal to 5 different items. (2)
8. Create a SUPPLIER PRICE LIST showing the Supplier CompanyName, ProductName and UnitPrice for all products from suppliers located in the United

Homework #3: SQL

States of America or Germany. Sort the list in order from HIGHEST price to LOWEST price. (21)

9. Create an EMPLOYEE ORDER LIST showing, in reverse alphabetical order (LastName, FirstName), the LastName, FirstName, Title, Extension and Number of Orders for each employee who has more than 50 orders. (7)

10. Create an ORDERS EXCEPTION LIST showing the CustomerID and the CompanyName of all customers who have no orders on file. (2)

11. Create an OUT OF STOCK LIST showing the Supplier CompanyName, Supplier ContactName, Product CategoryName, CategoryDescription, ProductName and UnitsOnOrder for all products that are out of stock (UnitsInStock = 0). (5)

12. List the productname, suppliername, supplier country and UnitsInStock for all the products that come in a bags or bottles. (16)

13. Create a NEW table named “Top_Items” with the following columns: ItemID (integer), ItemCode (integer), ItemName (varchar(40)), InventoryDate (timestamp), SupplierID (integer), ItemQuantity (integer) and ItemPrice (decimal (9,2)) . None of these columns can be NULL. Include a PRIMARY KEY constraint on ItemID. (No answer set needed.)

14. Populate the new table “Top_Items” using these columns from the products table.

ProductID → ItemID

CategoryID → ItemCode

ProductName → ItemName

Today's date → Inventory Date

UnitsInStock → ItemQuantity

UnitPrice → ItemPrice

SupplierID → SupplierID

for those products whose inventory value is greater than \$1,500. (No answer set needed.)

(HINT: the inventory value of an Item is ItemPrice times ItemQuantity.) (16 rows inserted)

Homework #3: SQL

15. Delete the rows in Top_Items for suppliers from USA or Canada. (6 rows deleted. No answer set needed.)
16. Add a new column to the Top_Items table called InventoryValue ((decimal (9,2))). No answer set needed.
17. Update the Top_Items table, setting the InventoryValue column equal to ItemPrice times ItemQuantity. (No answer set needed.)
18. Drop the Top_Items table. No answer set needed.