CSE 348 Database Management Systems
SPRING 2018
PROJECT
(Due Date: 13.05.2018 23:59)

A. CREATING TABLES (10 points)

Create appropriate tables and primary and foreign keys of them in a database for an entertainment store (CSE348 Store) management system using a PHP page with criterias explained in Part B. You should derive more normalized database scheme to get high points in overall project. **(This criteria will affect all of your project, not only part A.)**

a) Create tables for holding **DISTRICTs** and **CITYs** for Turkey.
b) Create a table for holding **BRANCHs** of CSE348Store available in a City
c) Create a table for holding information about **SALESMANs** for CSE348Store.
d) Create a table to hold information about **CUSTOMERs** of CSE348Store.
e) Create tables to hold information about **PRODUCTCATEGORYs** and **PRODUCTSUBCATEGORYs.**
f) Create a table to hold information about **PRODUCTs**.
g) Create a **SALE** table which holds **PRODUCT**'s sale information, who bought the product (**CUSTOMER**) and who sold them (**SALESMAN**).
h) Add appropriate primary and foreign keys to the tables with considering the relations between tables and the information explained in Part B.

Prepare an installation page with PHP (can be named as install.php) and when clicked on installation button on it, it should create the tables described above and fill these tables with the information described in Part B.

Warning: Name your database as your name_surname (e.g. yusufcan_semerci), userame of your database : root and password: mysql

Your installation button on PHP page will create tables and the relations between them once when it is clicked. If there will be a data in the database, install button should delete all information, tables and relations between them in the database and then do its creation operation.

B. INSERT OPERATIONS (40 points)

Install button of your PHP GUI inserts data to the tables with the criterias below:

1) **CSE348Store** has 5 Branches in each of 81 Turkish Provinces. So you should save 81 Turkish Provinces and 7 districts in your database. All districts and cities of Turkey should be held in an csv file and imported to the appropriate tables with install action. **(See note 1 & 2)**

2) Each Branch has different **real turkish names**. So you should save different 81*5 branch names in your database. Save each branch with its name and city information (and/or other

information you think crucial for representing a branch). All branch names should be held in a CSV file and imported to the appropriate tables with install action. **(See note 1 & 2)**

3) In each branch, there is 4 salesman with different **real person name and surname.** Salesman names differ in each branch so you should save different 81*5*4 employee names in your database. Store employees with his/her name, surname and branch where s/he works (and/or other information you think crucial for representing a salesman). All salesman names should be held in a CSV file and imported to the appropriate tables with install action. **(See note 1 & 2)**

4) In each branch, there is minimum 3 customers and maximum 5 customers with different **real person name and surname**. Customer names differs in every branches so you should save different 81*5*5 customer names in your database. Keep customers with his/her name, surname (and/or other information you think crucial for representing a customer). All customer names should be held in a CSV file and imported to the appropriate tables with install action. **(See note 1 & 2)**

5) Each branch sales Music CD, Music DVD, Music Vinyl, PC Games, XBox Games, PS Games and books. In every branches of store, there may be at least 40 and at most 500 products with different categories (Music, Game, Book), subcategories (i.e, CD, DVD and Vinyl for Music, PC, XBox and PS for games) and **real product names**. Every branch should have at least one product for all the categories (Music, Game, Book). Keep products with its unique identifier number (ISBN for books, barcode number for other products), name and price (and/or other information you think crucial for representing a product). All item names should be held in a CSV file and imported to the appropriate tables with install action. **(See note 1 & 2)**

6) Each Customer buys at least 15 products with different category, subcategory, name and different amounts from different salesman in a branch of the CSE348Store on a different time. Store a sale operation with salesman information, customer information, product information, sale amount and sale date (and/or other information you think crucial for representing a sale).

Note1: Your Insert Operation should insert all data to tables. If there are data in the database, they should be deleted from the tables in database and then the insertion operation should take place. You should do insertion operation with importing data in the csv files to the appropriate tables.

Note2: For branch name, salesman name, employee name and book name, you should provide **real names** in **CSV files**. (Not Branch1, Branch2, …, Salesman1, Salesman2, … Customer1, … etc.). **If we fetch the case that you do not choose real names, your overall project will be graded out of 20.**

C. REPORTING (50 points)

Show the results of the queries below in Tables in appropriate PHP files.

1. Provide a list of the district names on a PHP file and if user chooses a district, report the below in different tables on the PHP file:
    a. The total income of each branches with their name, their city name, their district name, the salesman name and surname who does biggest number of sale with how many items s/he sold, the salesman name and surname who does least number of sale with how many items s/he sold. The salesmans in the result set should be in the district chosen. Write one SQL query for this report. **(DO NOT USE UNION FOR THIS QUERY)**
    b. The total sales income for each salesman with salesman name, branch name, district name, province name, maximum price of sale done from this salesman to which customer, its sales date, minimum price of sale done from this salesman to which customer and its sales date. The salesmans in the result set should be in the district chosen. Write one SQL query for this report. **(DO NOT USE UNION FOR THIS QUERY)**

2. Provide a textbox to get Customer name and surname on a PHP file and report the detailed invoice of sales for the customer with giving district name, province name, branch name, salesman name done the sale and the details of the sale with prices, VAT, price with vat, sales dates, the total sale price, total VAT price on a table on the PHP file. Write one SQL query for this report. **(WRITE THE SQL QUERY AS YOU DO NOT GET ANY REDUNDANT INFORMATION ON RESULT SET OF QUERY !)**

3. Provide a list of the branch names on your GUI and if a user chooses a branch, report below in different tables on your GUI:
    a. The total amount number and price of sales done from each customer. Write one SQL query for this report.
    b. The total amount number and price of sales performed from each salesman. Write one SQL query for this report.

4. Provide a list of cities of Turkey. With the chosen city draw the below charts:
    a. A pie chart which will compare the total earnings of each branch in that city. Write one SQL query for this report.
    b. A bar chart which will compare the maximum and minimum total sale values of the salesmans who are in the branches in that city. Write one SQL query for this report.

IMPORTANT NOTES FOR THE PROJECT

1) You should do your project using PHP and MySQL. The assignments done with different combinations, frameworks, languages and Database Management Systems will get 0 points. The projects without PHP will get 0 points.

2) Use your PHP Application only to get information to build the proper SQL queries, send the SQL query to DBMS and show any result of the SQL query coming from DBMS. Do not get all information in the database tables to your PHP Application and do filtering in it. If so, you will get 0 points.

Submission: Compress all the files related with the project ***in a single ZIP*** file and name it with your name and surname, attached in front of "**Project**" (e.g. "***KemalCagriSerdaroguProject.zip***"). Submit your work through the "***CSE 348 - Project***" link on the course page at COADSYS.

You will do a demo for the project. The time and the procedure of the demo will be announced later.

Projects will be checked for plagiarism and if a copied assignment will be observed, the copied work will be graded with 0. No excuses will be accepted !!!