

COMP 4754

Database Systems

Assignment 2: SQL and Normalization

Instructions: Check the instructions in the syllabus. The university policy on academic dishonesty and plagiarism (cheating) will be taken very seriously in this course. *Everything submitted should be your own writing.* You must not let other students copy your work.

Note: Submit the answer of part A (1-9) in a separate SQL file, make sure the syntax you used is valid for MySQL, as for marking, the file will be tested on MySQL workbench, for the other parts, just submit the answer in separate pdf file. Also, after you create the tables, you can insert any dummy data in each table for testing your queries, but you don't need to include the insert statement in your submitted SQL file.

Video

<u>Catalog #</u>	<u>Video #</u>	Title	Status	Daily rental fees	Category Name
------------------	----------------	-------	--------	-------------------	---------------

Rental

<u>Catalog #</u>	<u>Video #</u>	<u>Customer #</u>	RentDate	Returndate
------------------	----------------	-------------------	----------	------------

Customer

<u>Customer #</u>	Name	RegDate	Street	City	Province	PostalCode
-------------------	------	---------	--------	------	----------	------------

A – SQL (70 marks)

Suppose we simplified the assignment 1 schema to be just 3 tables as shown, and each video will have only one category, and we skipped the actors. So, we ended by having the category to the video table

- 1- Write the SQL statement to create videoRental database for those 3 tables only, you should add any primary/ foreign keys constraint, also, catalog number, and customer number in video and customer tables should be auto incremented Integer. The video number represents the copy number (6 digits number represent the serial number for each copy), so, it doesn't need to be autoincremented. all columns shouldn't be null except the return date it is null until the customer returns the video, it will be with the value of that date, also the status can have only one value of (available, rented) or (true for available and false for rented), the video category can only be one of (Action, Adult, Children, Drama, Horror, or Sci-Fi)

- 2- Write SQL query that lists the Catalog number, video number, video title, category, and its status
- 3- Write SQL query that list only the video title and the number of copies we have regardless its rental status, sorted ascending by the video title
- 4- Update Query of number 3 to display only the available videos for rent, so we will display the title and the number of copies available for rent
- 5- Create SQL view “availableVideos” for the query of number 4
- 6- For the query of number 4, update that query to display the same result but only for action or Drama videos
- 7- Using sub/nested queries (don’t use join), write SQL statement that displays customer name, customer number, video title for any customer who rented a video and still hasn’t returned it back.
- 8- Create SQL trigger that updates the video status to be available when we update the returned date value
- 9- Write SQL statement to create stored Procedure “returnBackVideo” that take Catalog number, video number, customer number as input and it updates the return date for corresponding rental record for the inputs data to the current date, then it returns the total rental fess which is daily rental fees times the number of rented days. We don’t need to update the status video as it will be updated by the trigger
- 10- Write the relational algebra for queries in number 2, 7

B – Functional Dependency (10 marks)

Consider the relation **R** with attributes A, B, C, D, E, F, and G and with the following functional dependencies (FDs):

$AD \rightarrow E$ $BE \rightarrow F$ $B \rightarrow C$ $AF \rightarrow G$

Based on the given FDs, extract only one candidate key for the relation R. You don’t need to show the steps to find all available keys, just show the proof for the only one you chosen to be the candidate key.

C- Normalization (20 marks)

- 1- Suppose the customer’s name usually has only first name and last name, and we have the following FDs in the customer table

Postal code \rightarrow (Street, City, Prov)

Explain any changes in that table to be normalized in 3NF?

- 2- Suppose in the video table we have the following FD

Catalog number \rightarrow (title, daily rental fees, category),

explain any changes in the design of that table to normalize in 2NF?