National University of Computer and Emerging Sciences, Lahore Campus

Course:	Database Systems (Lab)	Course Code:	CL203
Program:	BS (Computer Science)	Semester:	Fall 2018
Duration:	2 Hours	Total Marks:	40
Paper Date:	12-12-2018	Weight	35%
Section:	All	Page(s):	3
Exam:	Lab Final Exam	Reg. No	

EXAM_NO_1

Important Instructions (Please read them before attempting the exam):

- Submit **ONLY .sql File** in this format (All parts of a question in one SQL File **named** with your **Roll Number** e.g., L13-4152). Do not zip your file.
- Copy the schema from the following path \\sandata\xeon\Ishaq\Database Systems- Fall 2018\Lab\final_schema.sql. Put the sql file in your D drive, and then **unplug the ethernet cable.**
- **Plagiarism** will result in **F grade** in lab.
- No cell phones are allowed. Sharing of **USBs** or any other items is **not allowed.**
- Submission path will be announced soon.
- **Write your roll number on this paper, and submit this paper to invigilator before leaving the lab.**

SQL Server Login Details:

- **Server: localhost or SQLExpress**
- **username: sa**
- **password: fstky2e4mdt**

Question#1 [5+5+5]

a) List the **art_id, art_name,** and **first_name** of the creator of that art piece whose total quantity sold is less than the total quantity sold of the art piece whose id is 1.

b) Create a view to list all the art pieces, along with the first and last name of their creators, which have never been sold using card payment mode.

c) List the first name and email of those creators who are trainers as well and have made an art piece of type ‘scenery’ as well as ‘portrait’ but not ‘sculpture’.

Question#2 [7]

Create a trigger such that whenever that trigger is fired as a result of insert in creators table, the trigger will insert the data itself in the creators table (the original insert statement will not insert data). However, the trigger will first check whether the first_name and last_name are different. If they are the same, data will not be inserted. Also print an appropriate message on successful as well as unsuccessful insertion.

Question#3 [3+7]

a) Create a UDF that takes **Creator_id** and returns the total quantity of art pieces of that creator sold.

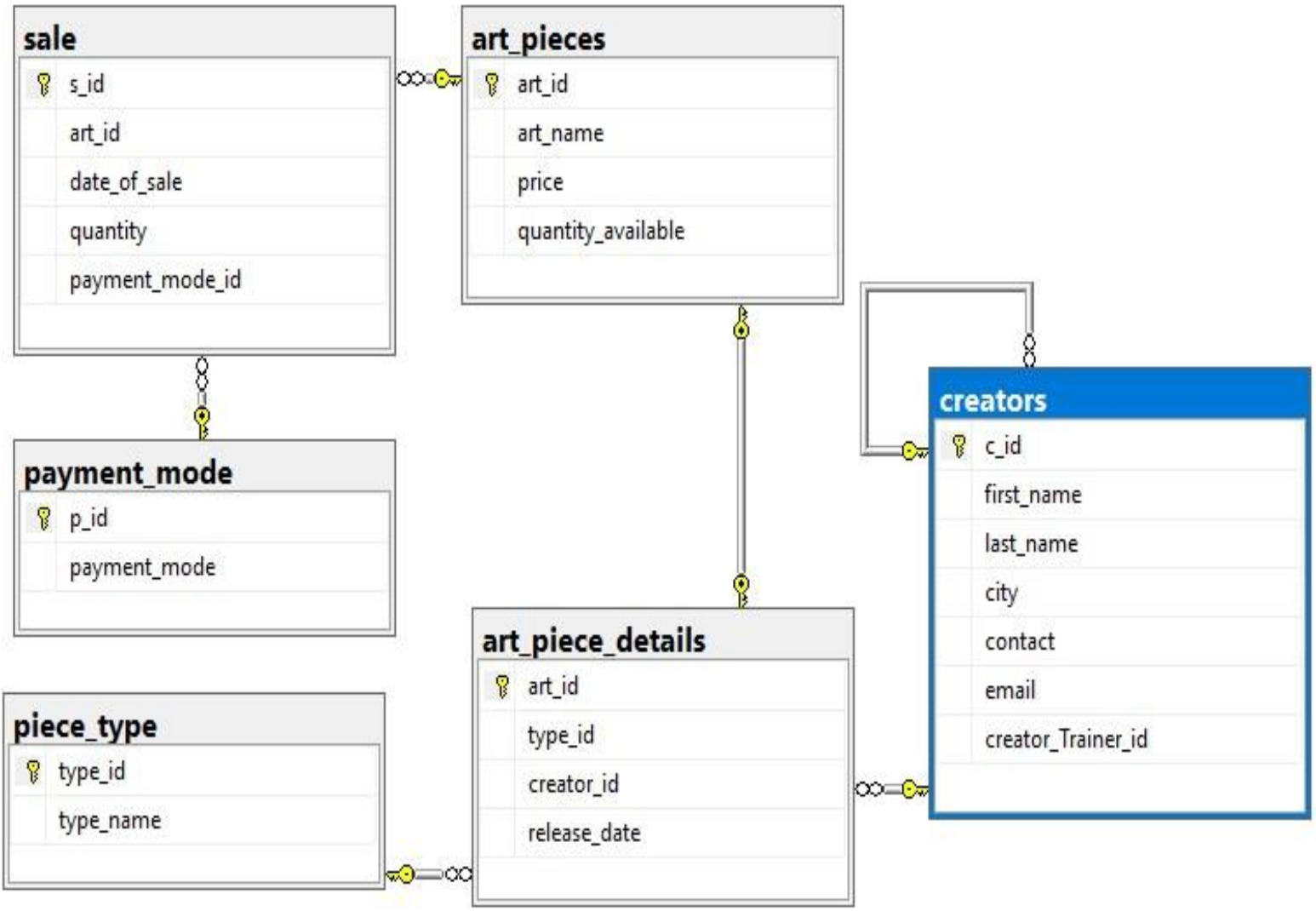
b) Create a view that returns **Name and ID** of the creator whose every art piece of type 'scenery' has been sold more times than other creators who are trainers as well.

Question#4 [6+2]

We want to add a procedure in the database that allows the customers to return the art pieces that they have bought. The amount that the customer had paid for the art pieces will be refunded. For this purpose, create a procedure ReturnSoldArtPieces which actually deletes the given s_id from sale table. But before deleting, it will perform the following tasks.

- ➔ The procedure checks whether the s_id is valid or not. If the s_id is valid, then deletion will be successful, and a status "Deleted Sucessfully" will be returned to the output parameter. If the id is invalid, then deletion will be unsuccessful, and a status "Sale Id Invalid" will be returned to the output parameter.
- ➔ System will also return the amount refunded to the customer as output parameter. Amount refunded will be calculated using price per unit and total quantity returned.
- ➔ The quantity of art piece which has been returned will also be restored in art_pieces table.

Also write the code to execute the procedure with s_id=1. Print the result of output parameters as well.


sale
s_id
art_id
date_of_sale
quantity
payment_mode_id
art_pieces
art_id
art_name
price
quantity_available
creators
c_id
first_name
last_name
city
contact
email
creator_Trainer_id
payment_mode
p_id
payment_mode
art_piece_details
art_id
type_id
creator_id
release_date
piece_type
type_id
type_name

0009

National University of Computer and Emerging Sciences, Lahore Campus

Course:	Data Base (Lab)	Course Code:	CS-2005
Program:	BS (Computer Science)	Semester:	Spring 2023
Duration:	2.5 Hrs	Total Marks:	55
Quiz Date:	12-June-23	Weight	_40_
Section:	BCS-4E, BCS-4F, BCS-4G, BCS-4H, BCS- 4J, BSE- 6A, BSE- 6B [SLOT 1]	Page(s):	2
Name		Roll No.	21L-5279

Instruction/Notes:

1. Use the schema and final exam from the folder: \\cactus1\Xeon\Spring 2023\Samman Ashraf\Database Lab Final.
2. Use of internet is not allowed.
3. **Submission Format:**
   a. Make a folder with your Registration No and paste TWO files (Word & SQL) in it.
   b. **Word File:** Queries should be pasted in this given file along with their screenshots of running queries (these are must otherwise marks will be deducted). Submit this word file named as (L13-4632.docx).
   c. **SQL File:** It includes all your .sql queries of the given questions. (e.g; yourName.sql)
4. Submit your final paper in the given folder. \\cactus1\Xeon\Spring 2023\Samman Ashraf\Database Final Submissions\Your Section (e.g; ABC-4K).
5. **Attempt all questions carefully. Your marking is Binary. Please Note SS is Screenshots.**

tashe

Use CompanyDB for the paper

Question no 1: **[5 (1.5 SS) + 5 (1.5 SS)= 10 Marks]**

a) Write a SQL query to retrieve the total hours worked by employees in each department, along with the department name and the employee with the highest hours in each department.

b) Write a SQL query which retrieves the names of employees and their corresponding department names for employees who have worked on all tasks in their department. (Nested Query)

Question no 2: **[7.5 (1.5 SS) + 7.5 (1.5 SS)= 15 Marks]**

a) Create a view named "EmployeeSummary" that displays the top 2 employee name, department name, and the total number of hours worked by each employee. Include only those employees who have worked at least 20 hours in total. Order the results in descending order of the total number of hours worked. Create a view named "EmployeeSummary" that displays the top 2 employee name, department name, and the total number of hours worked by each employee. Include only those employees who have worked at least 20 hours in total. Order the results in descending order of the total number of hours worked.

b) Create a view "TimeEntrySummary" to get a summary of time entries by employee and task.

0009

Question no 3: **[7.5 (1.5 SS) + 7.5 (1.5 SS)= 15 Marks]**

a) Create a stored procedure named GetDepartmentTasks that accepts a department ID as a parameter and returns all the tasks associated with that department, along with the total number of employees assigned to each task.

b) Create a stored procedure that calculates the average hours worked per task for a specific department:

Question no 4: **[7.5 (1.5 SS) + 7.5 (1.5 SS)= 15 Marks]**

a) Create a trigger that automatically updates the "updated_at" column of the "employees" table whenever a row is updated.

b) Create a trigger that prevents the deletion of a department if there are employees associated with it.

****

Question #3: Attempt all parts of the questions. Also, Write commands to execute your answer correctly if needed. ***Marks [10]***

g) Write a query to find the DoctorName , Designation that have less salary in comparison to highest Nurse 's salary of Cancer Department. [5]

h) Write a query to find all patients who have appointments with Doctor ID = 3. [2]

i) Write a query to retrieve the NurseID , NurseName of nurses who are on Night shift in the "Cardiology" department. [Night shifts starts from 20:00 to 8:00] [3]

Question #4: Attempt all parts of the questions. Also, Write commands to execute your answer correctly if needed. ***Marks [35]***

j) Write a trigger to automatically update the NumberOfRooms in the Department table whenever a new NurseShift is added. [10]

k) The Hospital administrator wants to keep track of nurse shifts and ensure that no department is understaffed. Whenever a nurse's shift is added, updated, or deleted, the administrator needs an automatic update of the current number of nurses per department to manage staffing levels efficiently. Write a trigger that updates a summary table, DepartmentNurseCount, which stores the DepartmentID, deptName, and the current count of nurses (NurseCount) for each department. [15]

l) Create a trigger named AuditNurseSalaryChange that will be fired after any update to the salary in the Nurse table. [10]
   a) The trigger should insert a record into a new table called NurseSalaryHistorywith the NurseID,the old salary, and the new salary.
   b) Assume the NurseSalaryHistory table has already been created with the appropriate columns.

Question # 5: Attempt all parts of the questions. Also, Write commands to execute your answer correctly if needed. ***Marks [5x1 =5]***

m) Write a command to create a new table 'Medications' with appropriate fields and constraints.

n) Write a DML command to insert multiple new nurses in a single command.

o) Write a DML command to update the address of a patient to default value '123 St 1,A town, USA' if null .

p) Write a command such that the start_time of Nurse shift cannot exceed end_time. (24 hr format)

q) Write a constraint statement that will delete the records of nurse shifts and department assigned automatically if the nurse has been removed.

National University of Computer and Emerging Sciences

Lahore Campus

Database Systems Lab

Date: 23rd May, 2024

Sections : BCS , BSE

Final Exam

Total Time (Hrs): 2hr 30 mins

Total Marks: 80

Total Questions: 5

Do not write below this line.

22L-7947 Roll No

BS(SE)-4C2 Section

Uhami Student Signature

Instructions for Students and Invigilator:

1. You have to submit the question paper with **your name, roll no written** on it. Invigilator kindly make sure **to collect all question paper back** from Students.
2. Make a folder named as your **Roll_no_Section. Place Sql files , Text Files and Screenshots** of your answer in it .
3. Submit your folder on Submission path on Xeon. I.e. **Xeon\Spring 2024\DB Lab Final\Your Section\your Lab sub_Section** for example : Xeon\Spring 2024\DB Lab Final\4Z\4Z1 submissions.
4. In case of any issue, make assumption and mention it in comments.
5. **Plagiarism and Cheating are strictly prohibited.**

Question #1: Attempt all parts of the questions. Also, Write commands to execute your answer correctly if needed. ***Marks [10]***

a) Create a view to show all departments with their number of rooms. [3]

b) Create a view to list nurses and their shift timings. [2]

c) Create a view named DoctorPatientAppointment that displays the following information: [5]

   a) Additionally, sort the results by the appointment time in ascending order.

   b) The view should only include appointments scheduled for the March 2021.

   c) Doctors name, patient;s name, department name, and appointment time.

Question #2: Attempt all parts of the questions. Also, Write commands to execute your answer correctly if needed. ***Marks [20]***

d) Create a stored procedure named **UpdateNurseSalary** that takes two input parameters: NurseID and NewSalary. [10]

   a) If no nurse with the given NurseID exists, return an error message.

   b) If the update is successful, return a message indicating the salary has been updated.

   c) The procedure should update the salary of the nurse with the given NurseID to the NewSalary.

e) Write a stored procedure that takes DepartmentID and returns the total number of doctors and Nurses in that department. [4]

f) Write a stored procedure to schedule an appointment, which inserts a new record into the Appointment table. Ensure the doctor and patient IDs exist before scheduling, if they don't exist raise error. [6]