Lab Manual

5

EE436L: Database Engineering

Department of Electrical Engineering
University of Engineering and Technology Lahore 2020
Instructor: Mr. Umer Shahid

Name _____
Registration Number _____

Lab Title: WRITING BASIC SQL QUERIES

Note: Please write simple and non-redundant SQL queries for all parts. Do not use the workbench interface options.

Exercise 1

Following the steps below, create a simple database of University Campus which has multiple buildings. Each building entity is distinguished on the basis of a unique id. Each building also has a foreign key relationship with its parent table -Campus.

1. Create a Table 'Campus' with following attributes

Attribute Name	**Type**		
CampusID	INTEGER(5)	Primary Key	NOT NULL
CampusName	VARCHAR(30)		NOT NULL
CampusAddress	VARCHAR(50)		NULL
Country	VARCHAR(20)		NULL
City	VARCHAR(20)		NULL
isOpen	BIT		NOT NULL

2. Create Table Building with following attributes

Attribute Name Type

BuildingID	INTEGER(5)	Primary Key	NOT NULL
BuildingName	VARCHAR(30)		NULL
Address	VARCHAR(30)		NULL

3. Implement the foreign key relationship between the two entities such that CampusId should appear as the Foreign Key in the Buildings table. Implement referential integrity constraint of Set Null on Delete Rule on above relationship. Implement referential integrity constraint of Set Cascade on Update Rule on above relationship.

4. Insert 5 different Buildings and Campuses of your own choice. Make sure that the primary keys are unique.

5. Display all the campuses and buildings.

6. At least add two Buildings in previous step such that they belong to the same campus and then select them.

7. Update a CampusName to "myCampus".

8. Delete the campus added in the step (VII) on the basis of CampusName.

Exercise 2
Consider a relational schema with the following relations:

Gym (**gymName**, owner, street)
Frequents (**customerName, gymName**)
Customer (**customerName**, street, age)

The relations Gym and Customer store information about gyms and people respectively, while the 'Frequents' relation stores information about which people visit which gym frequently.

1. Write a query to return the names of gyms that "Usman" frequents. Here "Usman" is the name of one of the customers.
2. Write a query that returns the set of customers who frequent only gyms on the same street that (s)he lives on.