## **COSC 30603**

**Lab Assignment 6: More on SQL**

Name: Alfredo Perez

Due: Oct 12

1. In this assignment, we will first complete data loading.
   1. Download lab6\_data.zip and unzip it on your computer. E.g. I copied the folder to D:\ (Assume I installed MySQL on my local PC, if you are using VMware, you need to make some adjustments)
   2. Create the relation schemas.

Run lab6\_schema.sql in MySQL Workbench. The schema is based on the EER diagram on the next page. In addition to the tables used in Lab 5, three new tables are added, namely CourseDescription, CourseOffering and taken. Note that the only difference between this EER diagram and the one used in Lab 5 is that 1:N instead of N:M cardinality constraints have been assumed for the offer relationship type between CourseDescription and Department.

* 1. Enable MySQL LOAD

To enable MySQL’s load data capability, you need to start your MySQL from terminal using:

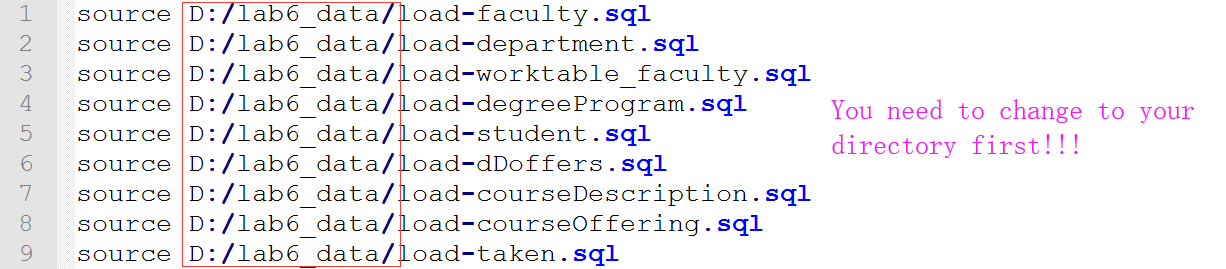


See this StackOverflow post for more info.

* <https://stackoverflow.com/questions/59993844/error-loading-local-data-is-disabled-this-must-be-enabled-on-both-the-client>

In this lab, if you need to execute load, you have to do that through terminal, NOT MySQL Workbench.

* 1. Modify source.sql



MySQL source command:

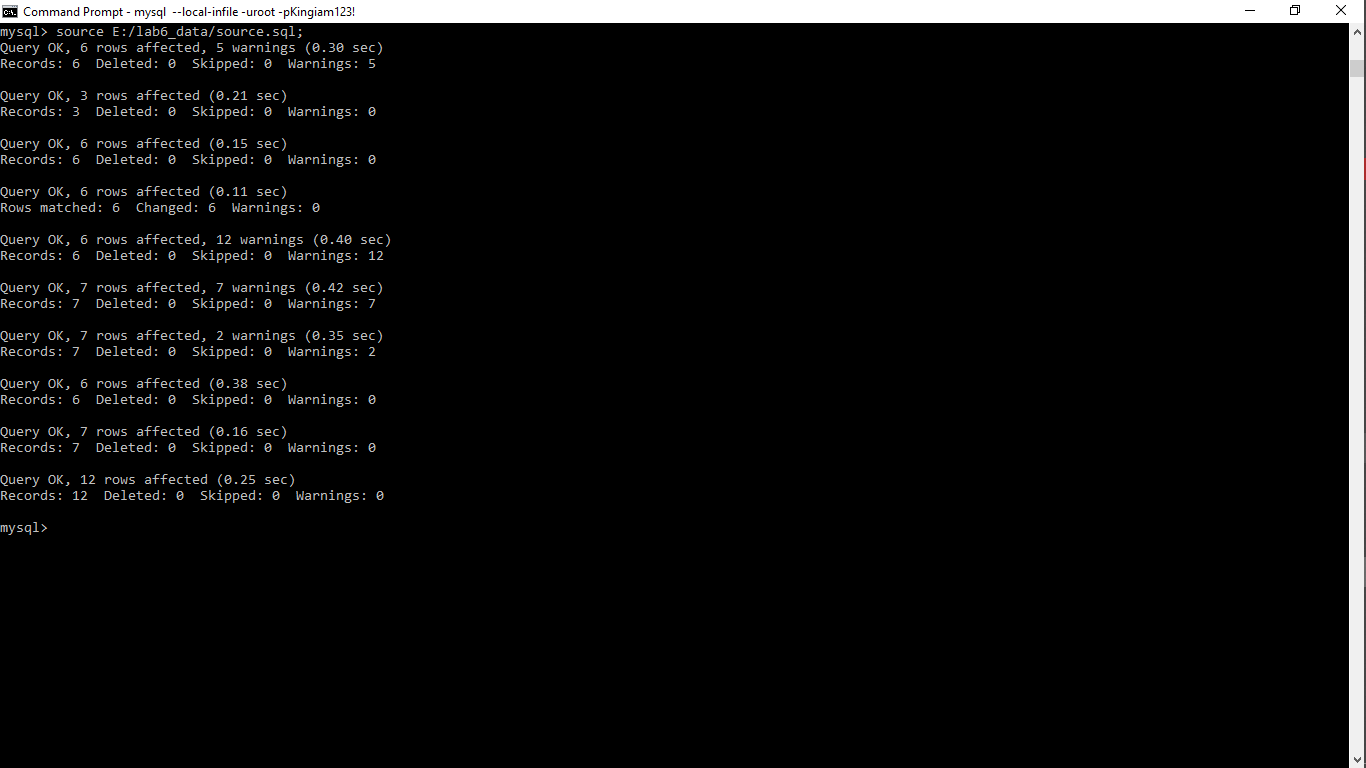
<https://dev.mysql.com/doc/refman/8.0/en/mysql-batch-commands.html>

* 1. Execute the following command in terminal.

source D:/lab6\_data/source.sql;

You need to change the file directory first.

Copy your results (prompts from terminal) here:





1. Create a Foreign Key Reference Graph.

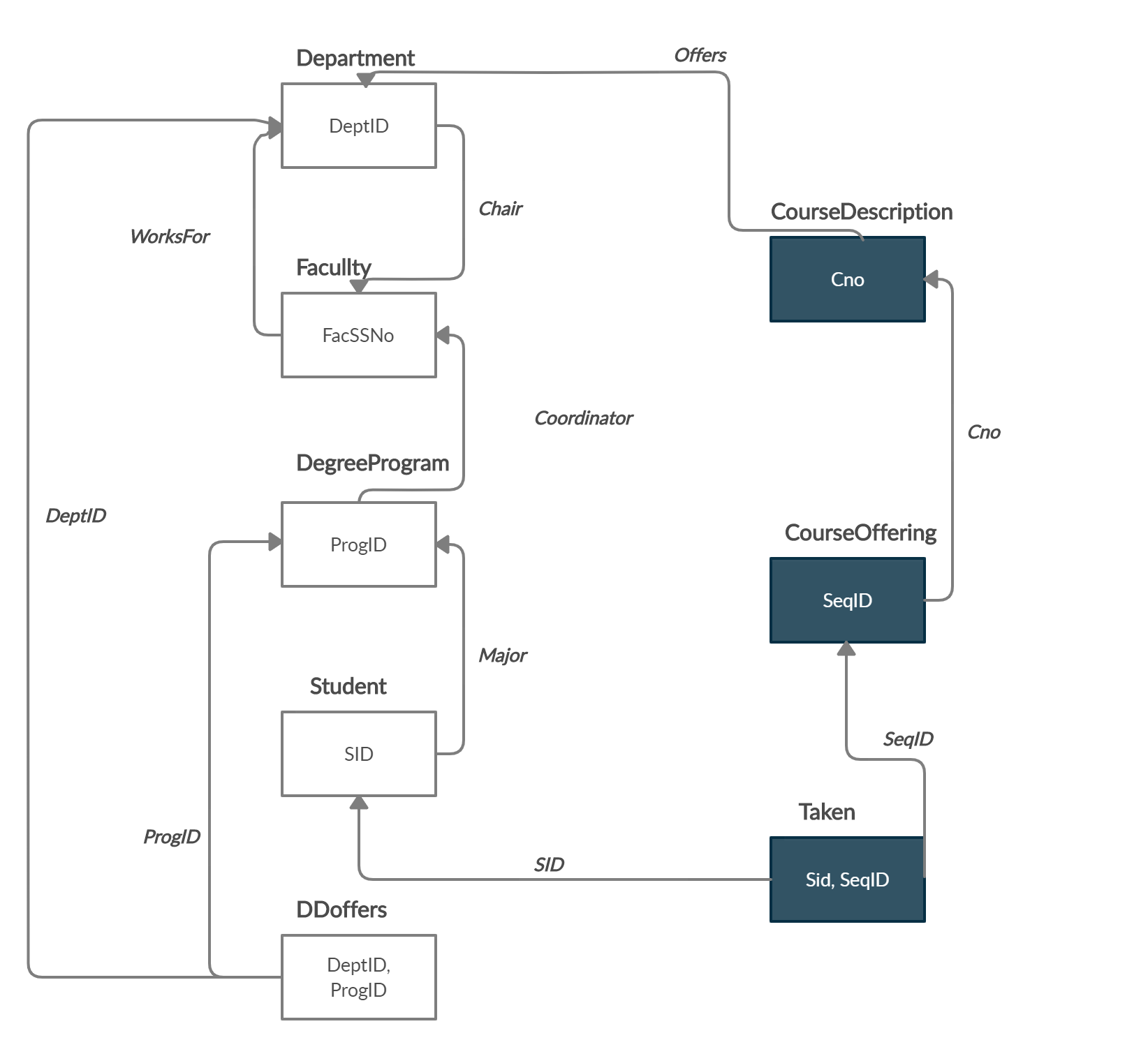
(a) A foreign key reference graph is a directed graph where nodes represent the referencing and referenced tables and a directed edge represents the foreign key reference from the referencing table to the referenced table. An edge is labeled with the foreign key.

A foreign key reference graph provides a good reference for designing queries that requires join operations of several tables. It is more helpful when the database consists of a large number of tables.

(b) A foreign key reference graph based on the lab schema is presented as follows.



(c) In the above graph, add the foreign references for the three additional tables of this lab, namely CourseDescription, CourseOffering and taken.



1. Nested queries. Note that you can use the foreign key reference graph on the previous page to design the queries.

(a) Consider the following nested query:

select deptname

from department

where deptid in (select deptid

from ddoffers x, degreeprogram y

where x.progid=y.progid and progtype='BS');

Give the meaning of the SQL statement in English.

Answer:

Fetching results of department names who offer degree program and of type BS

Run the SQL statement and check your answer.

(b) Give an SQL statement for the following plain English query. You are required to use a nested query.

Give the name of those departments that offers BS or PhD programs.

Answer:

select deptname

from department

where deptid in (select deptid

from ddoffers x, degreeprogram y

where x.progid=y.progid and progtype='BS' or progtype='Phd');

Run the SQL statement and check your answer.

(c) Give an SQL statement for the following plain English query. You are required to use a nested query.

Give the name of those departments that offers both BS and PhD programs.

Answer:

select deptname

from department

where DeptId in (

select j.DeptId from faculty x, degreeprogram y, ddoffers j

where x.FacSSNo = y.Coordinator and y.ProgType = 'PhD' and j.ProgId= y.progid

);

Run the SQL statement and check your answer.

1. Correlated nested queries

(a) Consider the following correlated nested query:

select sid as StudentId, sname StudentName

from student x

where 3.5<(select max(Grade)

from taken a, courseoffering b

where x.sid=a.sid and a.seqid=b.seqid and b.cno='CS480');

Give the meaning of the SQL statement in English.

Answer:

Fetching Student Id and Student Name of students who had maximum grades in course number CS480 greater than 3.5

Run the SQL statement and check your answer.

(b) Give an SQL statement for the following plain English query. You are required to use a correlated nested query.

Give id and name of those students who take CS480 and have an average grade greater than 3.5.

Answer:

select sid as StudentId, sname StudentName

from student x

where 3.5<(select avg(Grade)

from taken a, courseoffering b

where x.sid=a.sid and a.seqid=b.seqid and b.cno='CS480');

Run the SQL statement and check your answer.

1. Like operator

(a) Consider the following correlated nested query:

select deptid as DepartmentId, deptname as DepartmentName

from Department x

where deptid in (select DeptId

from ddoffers y, degreeprogram z

where x.deptId=y.deptid and y.progid=z.progid and

z.programname like '%Computer%' and

z.progtype='BS');

Give the meaning of the SQL statement in English.

Answer:

Fetching Department Id and Name of departments who offer programs with any name that contains “Computer” and of type “BS”

Run the SQL statement and check your answer.

(b) Give an SQL statement for the following plain English query. You are required to use the like operator.

Get department name and name of department chair for those departments that offer BS degree program and program name containing “Computer”.

Answer:

select deptname as DepartmentName, facName as 'Name of Chair'

from Department x, faculty a

where x.chair = a.FacSSNo and deptid in (select DeptId

from ddoffers y, degreeprogram z

where x.deptId=y.deptid and y.progid=z.progid and

z.programname like '%Computer%' and

z.progtype='BS');

Run the SQL statement and check your answer.

1. Exists and not exists

(a) Consider the following correlated nested query:

select sname

from student x

where not exists (select sid

from taken y, courseoffering z

where x.sid = y.sid and y.seqid = z.seqid

and z.cno = 'CS480');

Give the meaning of the SQL statement in English.

Answer:

Fetching Students names of Students who don’t take CS480

Run the SQL statement and check your answer.

(b) Give an SQL statement for the following plain English query. You are required to use exists.

Get id and name of those students who major in CS Department and take CS480.

Answer:

select sid as 'Student ID', sname as 'Student Name'

from student x

where exists (select sid

from taken y, courseoffering z

where x.sid = y.sid and y.seqid = z.seqid

and z.cno = 'CS480');

Run the SQL statement and check your answer.

1. Group by and having

(a) Consider the following query:

select x.seqid, y.cno, count(\*) as StudentCount,

avg(grade) as AverageGrade

from taken x, courseoffering y

where x.seqid=y.seqid

group by x.seqid, y.cno

having count(\*) >= 2

order by cno asc;

Give the meaning of the SQL statement in English.

Answer:

Fetching course sequence id, course number, total number of students taking the course being two or more and the average grade of the students in the course

Run the SQL statement and check your answer.

(b) Give an SQL statement for the following plain English query. You are required to use group by and having

Get instructor id, course sequence id and course number for all course offerings having an average grade of 4.0.

Answer:

select y.Instructor, x.seqid, y.cno,

avg(grade) as AverageGrade

from taken x, courseoffering y

where x.seqid=y.seqid

group by y.Instructor, x.seqid, y.cno

having avg(grade) = 4

order by cno asc;

Run the SQL statement and check your answer.

(c) Give an SQL statement for the following plain English query. You are required to use group by and having

Get instructor id and course number for all courses having an average grade of 4.0.

Answer:

select y.Instructor, y.cno,

avg(grade) as AverageGrade

from taken x, courseoffering y

where x.seqid=y.seqid

group by y.Instructor, y.cno

having avg(grade) = 4

order by cno asc;

Run the SQL statement and check your answer.

1. Outer join

*Even though we didn’t formally introduce outer join and inner join. We are using inner join all the time. We have never used outer join.*

*Here is a tutorial:*

[*https://stackoverflow.com/questions/38549/what-is-the-difference-between-inner-join-and-outer-join*](https://stackoverflow.com/questions/38549/what-is-the-difference-between-inner-join-and-outer-join)

*Read the Stackoverflow thread before you proceed, the example given in the best answer really helps understand difference between inner and outer join.*

(a) Insert two tuples first

insert into faculty (facname, facssno)

values ('John Doe', '000559999');

insert into department (deptid, deptname)

values ('MATH', 'Mathematics');

select \* from faculty;

select \* from department;

(b) Try the following two outer join queries and observe the results.

Left join:

SELECT facname, facssno, deptname

FROM faculty X LEFT JOIN department Y ON x.worksfor=y.deptid;

Right join:

SELECT facname, facssno, deptname

FROM faculty X RIGHT JOIN department Y ON x.worksfor=y.deptid;

1. Queries involving long join paths (in the foreign key reference graph)

(a) Consider the following query:

select deptname, facname

from department d, faculty f, courseoffering co,

coursedescription cd

where d.deptid=f.worksfor and f.facssno=co.instructor and

co.cno=cd.cno and cd.offers=d.deptid;

Give the meaning of the SQL statement in English.

Answer:

Fetching all department names and their faculty names

Run the SQL statement and check your answer.

(b) Give SQL statements for the following three English language queries. These queries involve only equi-joins just like the above SQL statement. For each of these queries, you first identify the correct join path in the foreign key reference graph. After you have identified the join path for a query, you design the SQL statements by using a sequence of equijoins.

* Get the name of each student and the student’s department name.

Answer:

select s.sname,d.deptname

from student s, department d, degreeprogram dp, ddoffers dd

where s.Major = dd.ProgId and dp.ProgId = dd.progid and dd.deptid = d.deptid;

* Get the name of each student and the names of those department that the student takes at least one course from.

select s.sname,d.deptname

from student s, department d, degreeprogram dp, ddoffers dd, taken t, courseoffering c

where s.Major = dd.ProgId and dp.ProgId = dd.progid and dd.deptid = d.deptid and s.sid = t.sid and t.seqid = c.seqid;

* Get student name and the student’s department name if the student takes at least one course offered by the student’s department.

select distinct s.sname,d.deptname

from student s, department d, degreeprogram dp, ddoffers dd, taken t, courseoffering c

where s.Major = dd.ProgId and dp.ProgId = dd.progid and dd.deptid = d.deptid and s.sid = t.sid and t.seqid = c.seqid

;