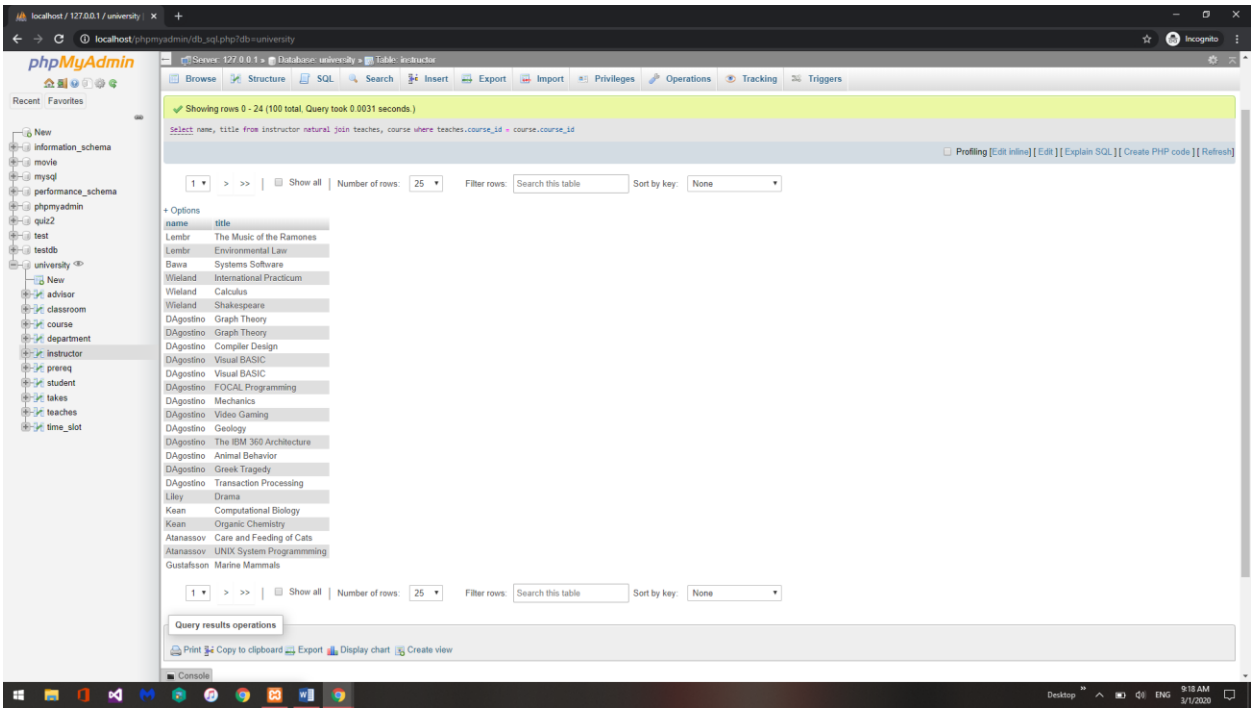
Chris Grimes
CS 33007
Homework 3
Part1: Practice Queries [35 points]

Execute following queries and observe the results

1. List names of the instructors along with the title of the courses they have taught.

(observe results of these queries).

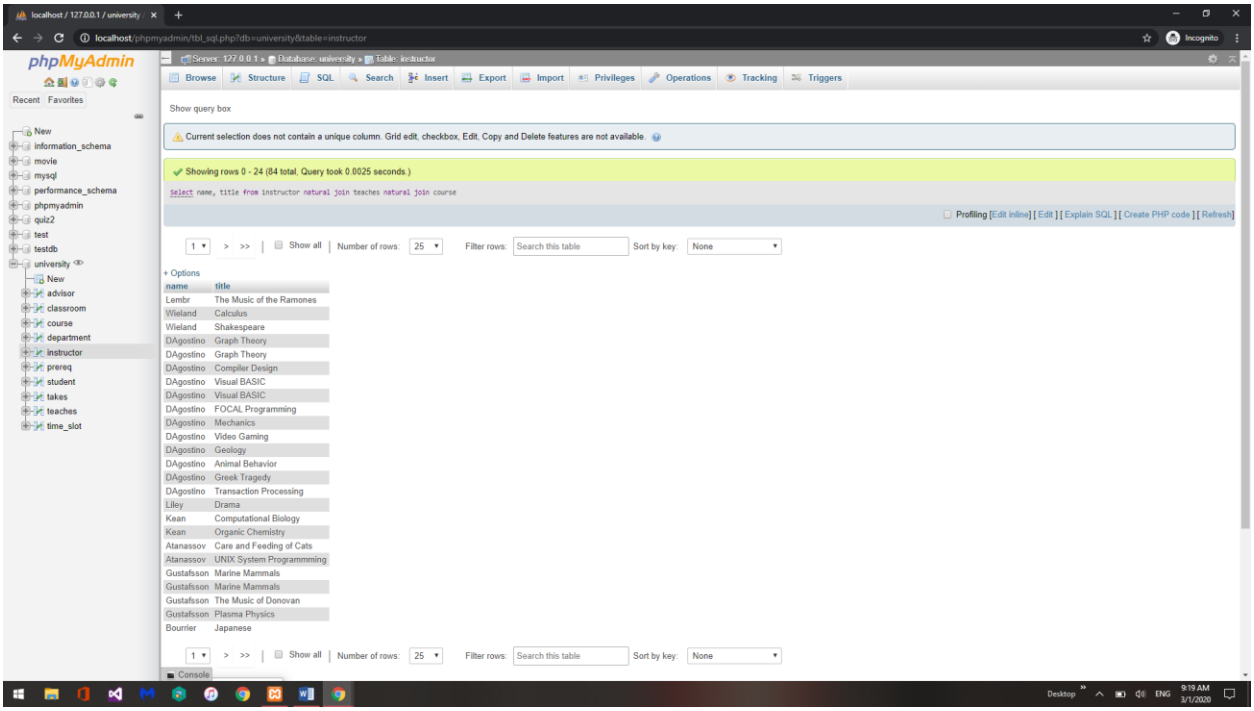
a. Select name, title from instructor natural join teaches, course where teaches.course_id = course.course_id



This query shows the name of the instructor and the title of the class said instructor is teaching.

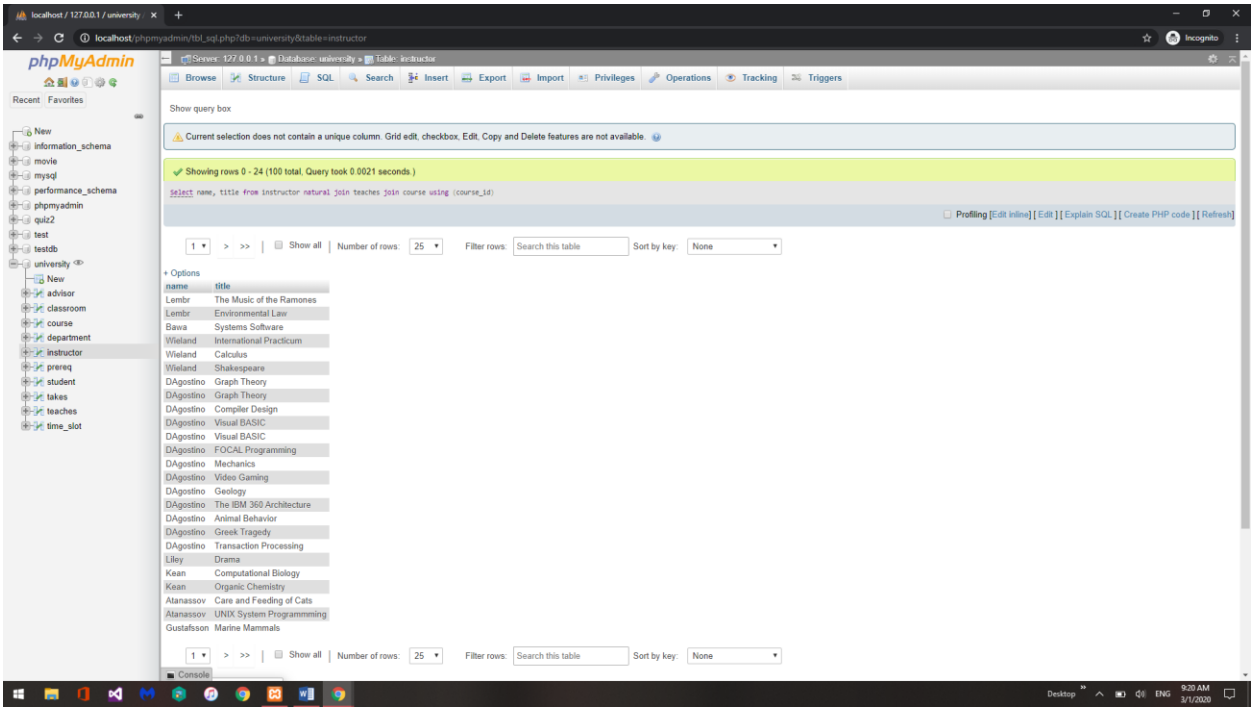
Chris Grimes
CS 33007
Homework 3

b. Select name, title from instructor natural join teaches natural join course



This query shows the same results as A since the natural joins is joining the tables based on common attributes.

c. Select name, title from instructor natural join teaches join course using (course_id)

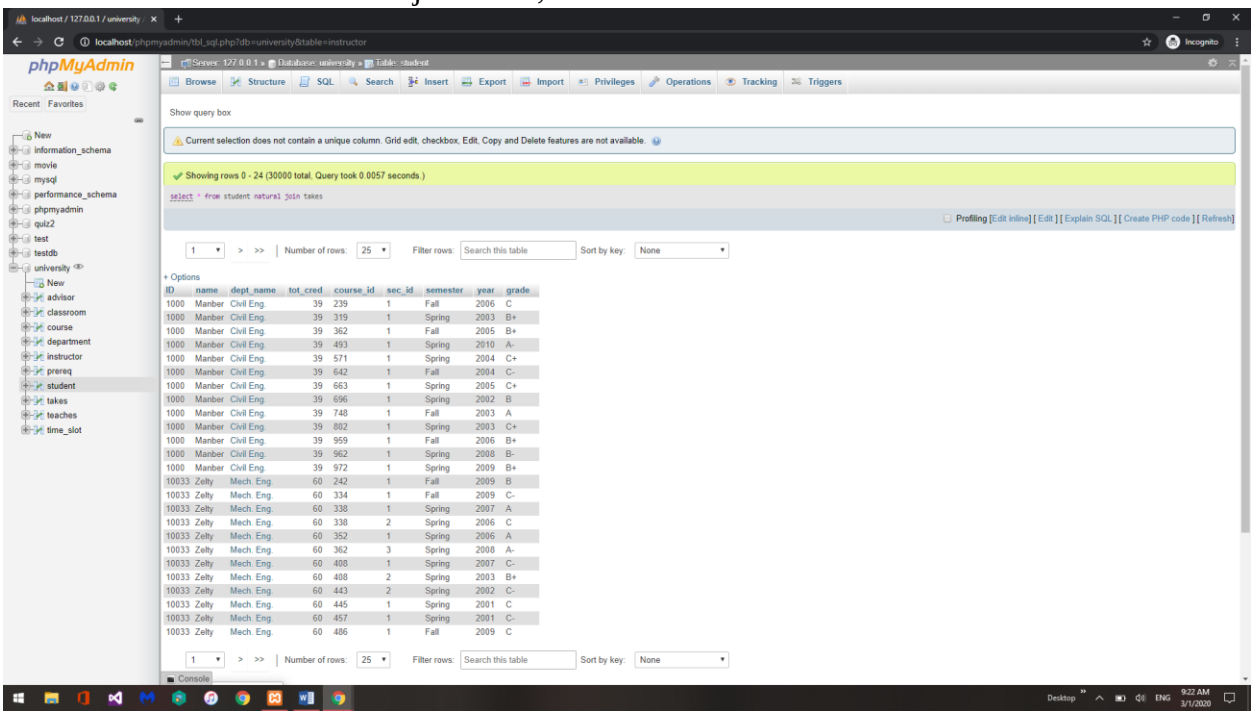


This query show the same results but was specifically joined on the common attribute course_id instead of letting the DBMS choose the common attribute

Chris Grimes
CS 33007
Homework 3

2. List of all students, displaying their ID, and name, dept name, and tot cred, along with the courses that they have taken (observe the differences in the result of the queries)

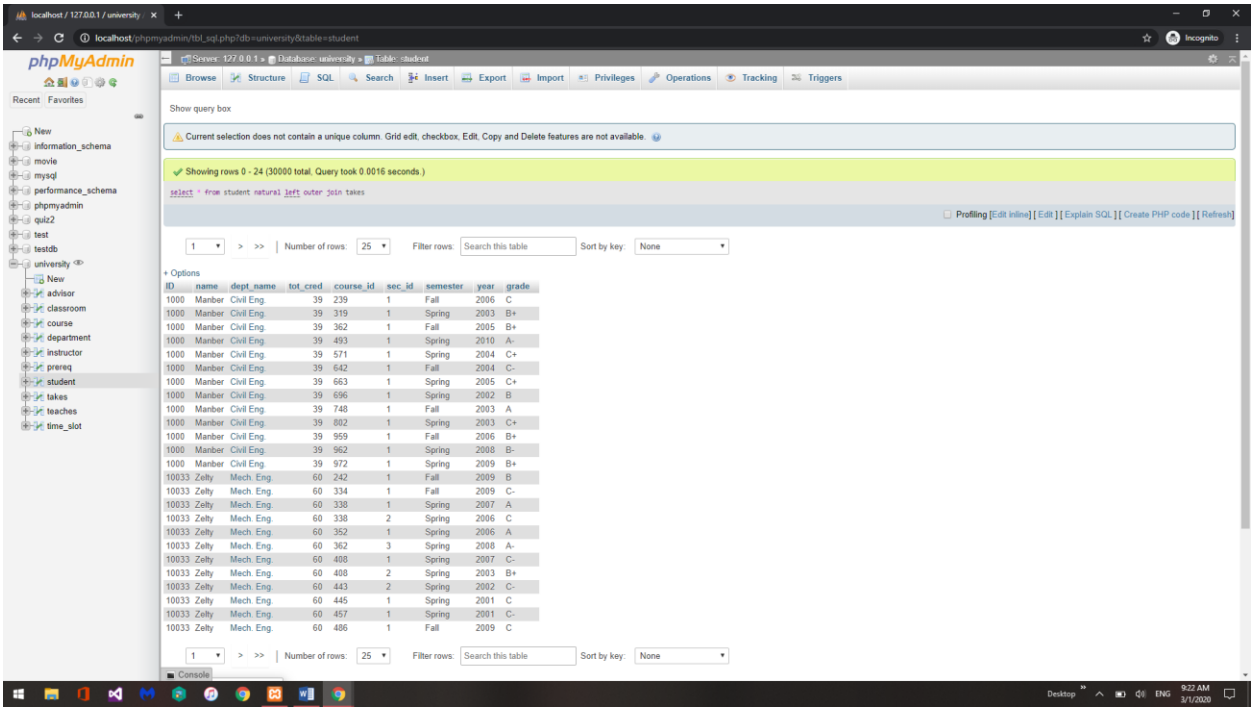
a. select * from student natural join takes;



This query shows results from student being naturally joined by takes

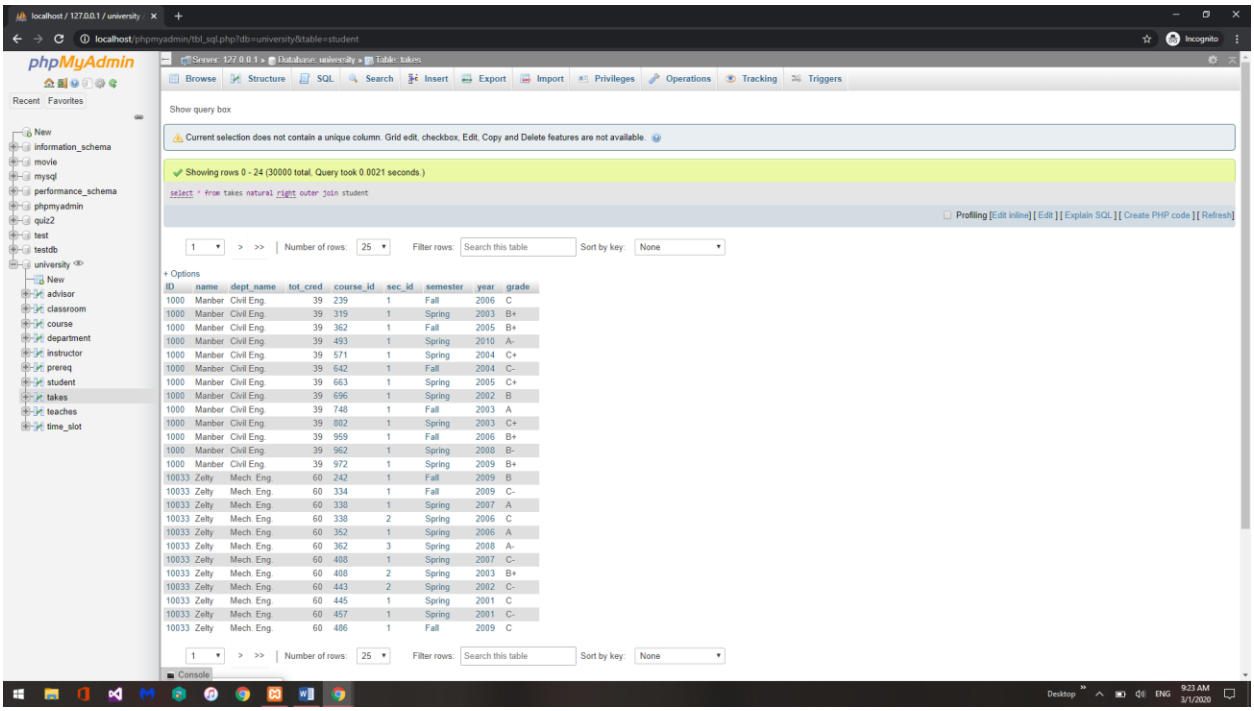
Chris Grimes
CS 33007
Homework 3

b. select * from student natural left outer join takes;



This query shows the same results as A since there are no empty columns in the previous query.

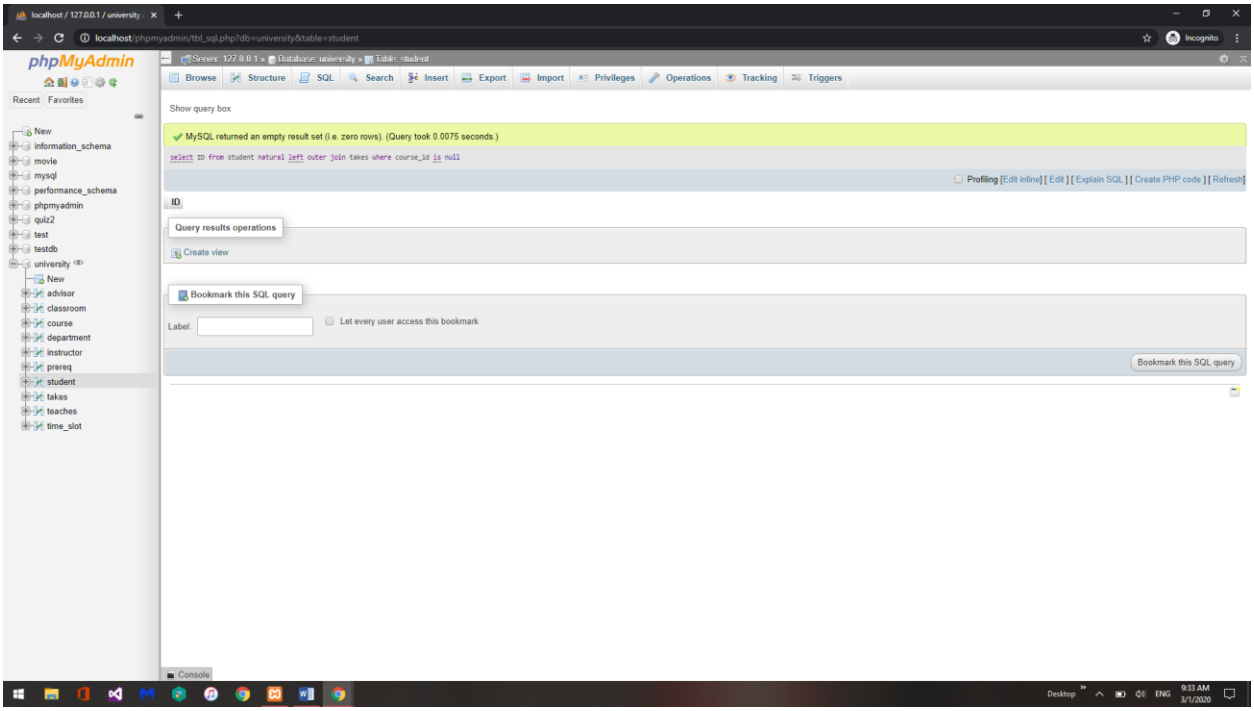
c. select * from takes natural right outer join student;



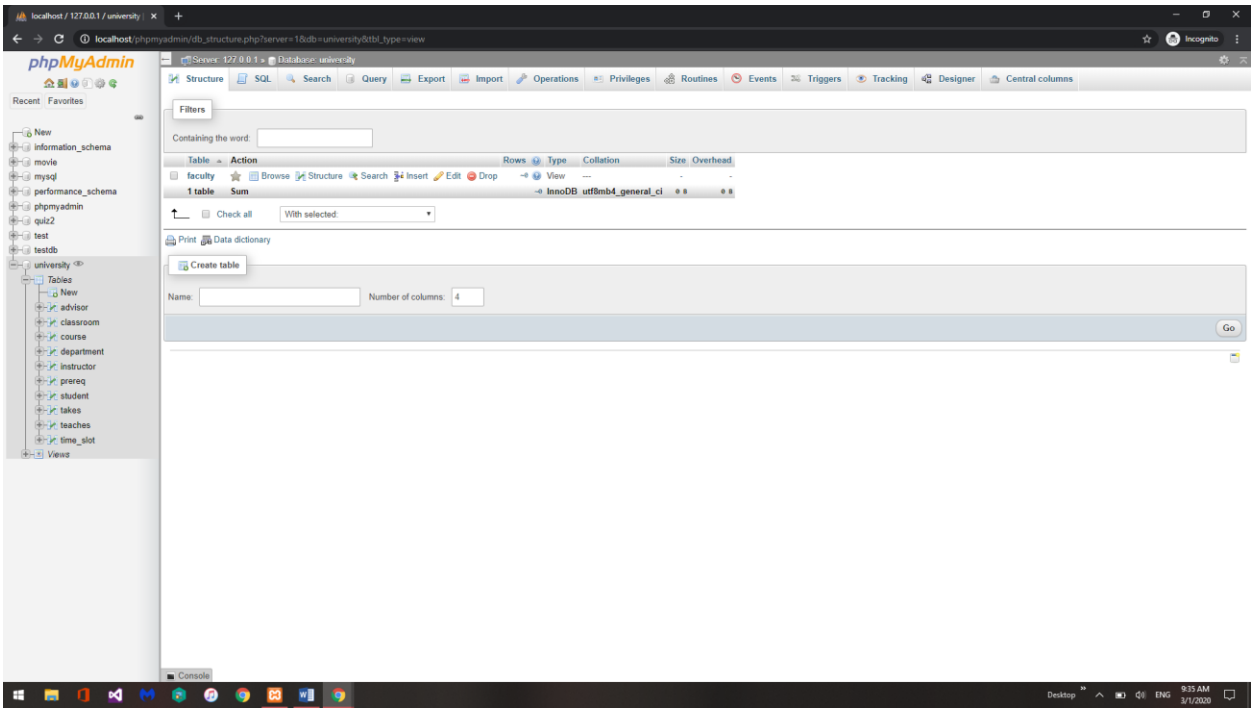
This query shows the same results as A since there are no empty columns in the previous query.

Chris Grimes
CS 33007
Homework 3

3. Find all students who have not taken a course" select ID from student natural left outer join takes where course id is null;

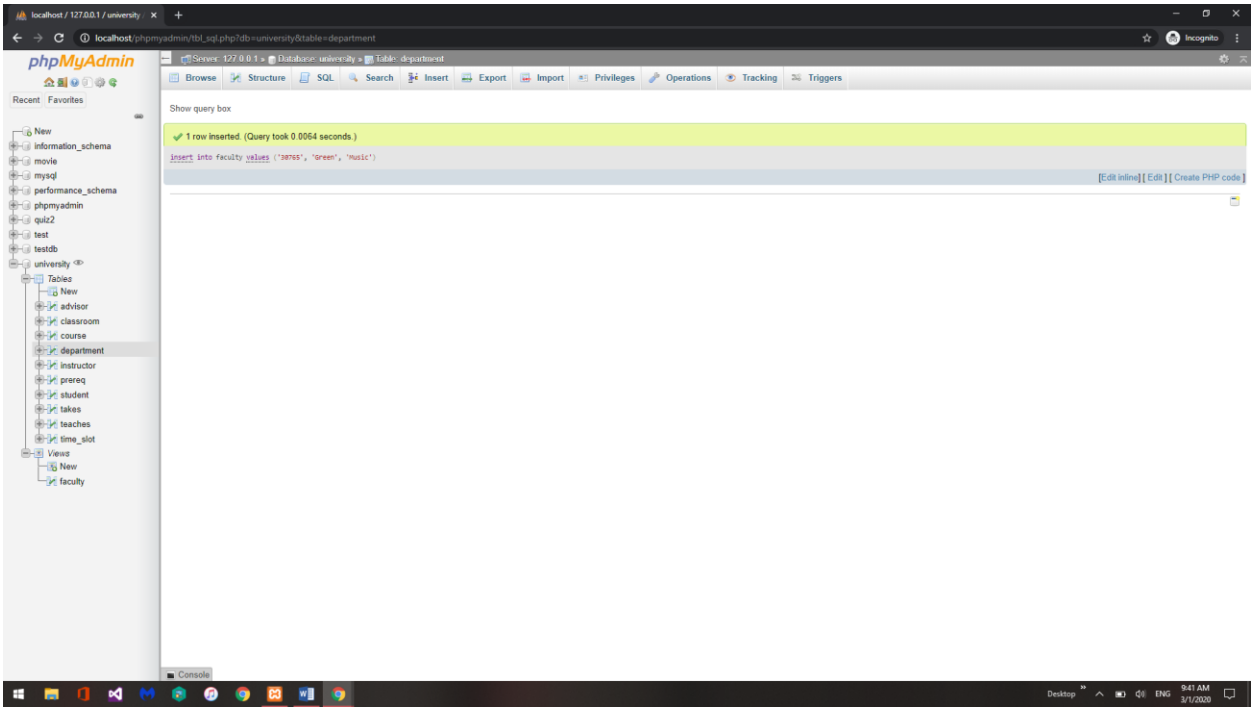


4. Create a view of instructors without their salary create view faculty as select ID, name, dept_name from instructor

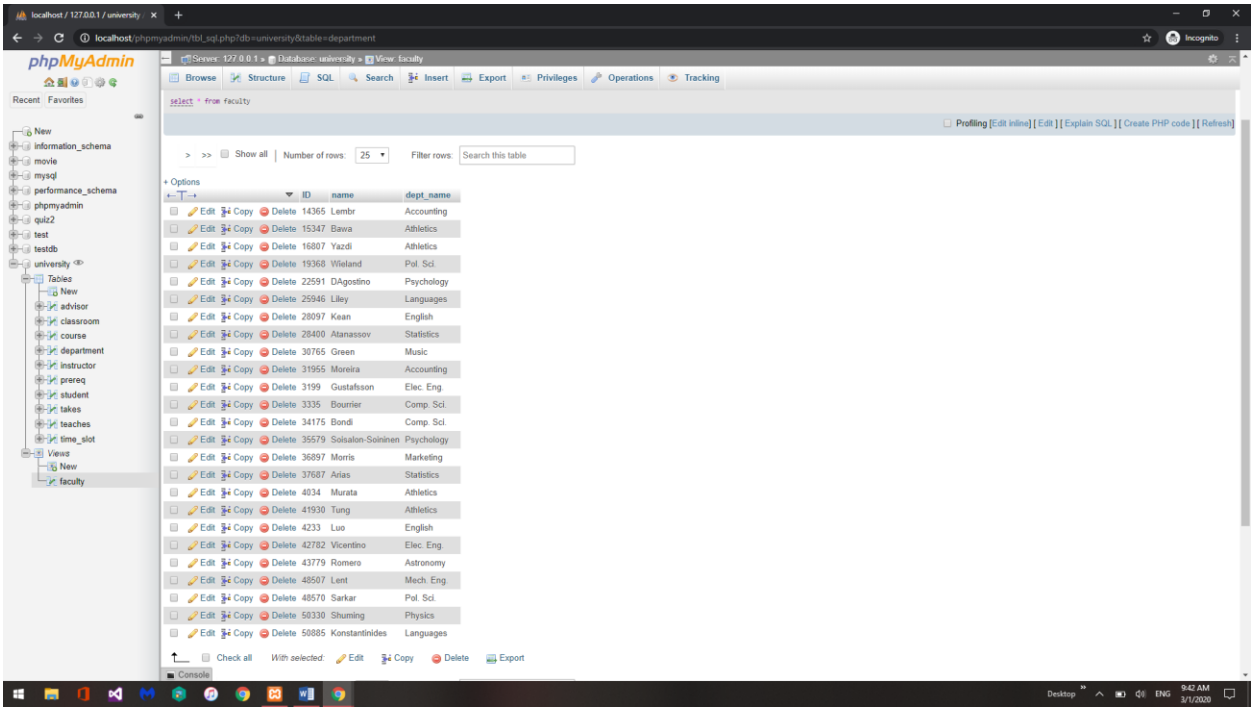


Chris Grimes
CS 33007
Homework 3

5. Insert a record into the view created in 4, then display the view to show that insertion was successful. insert into faculty values ('30765', 'Green', 'Music');

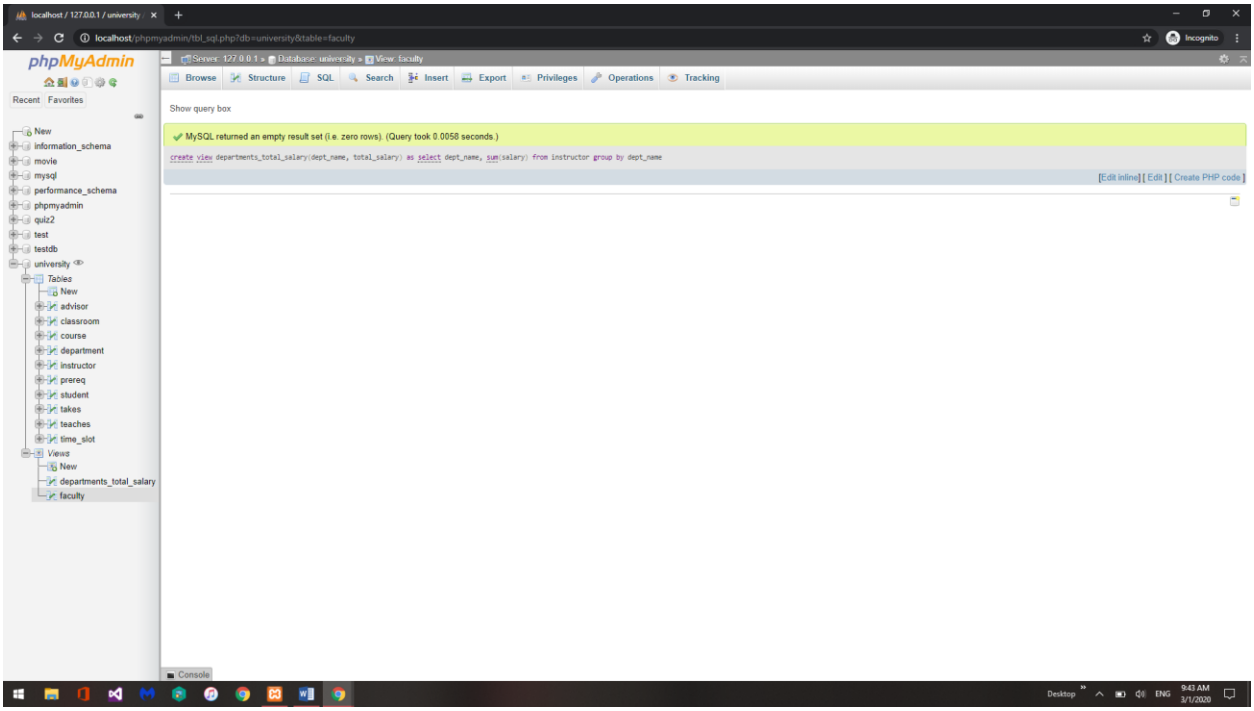


select * from faculty

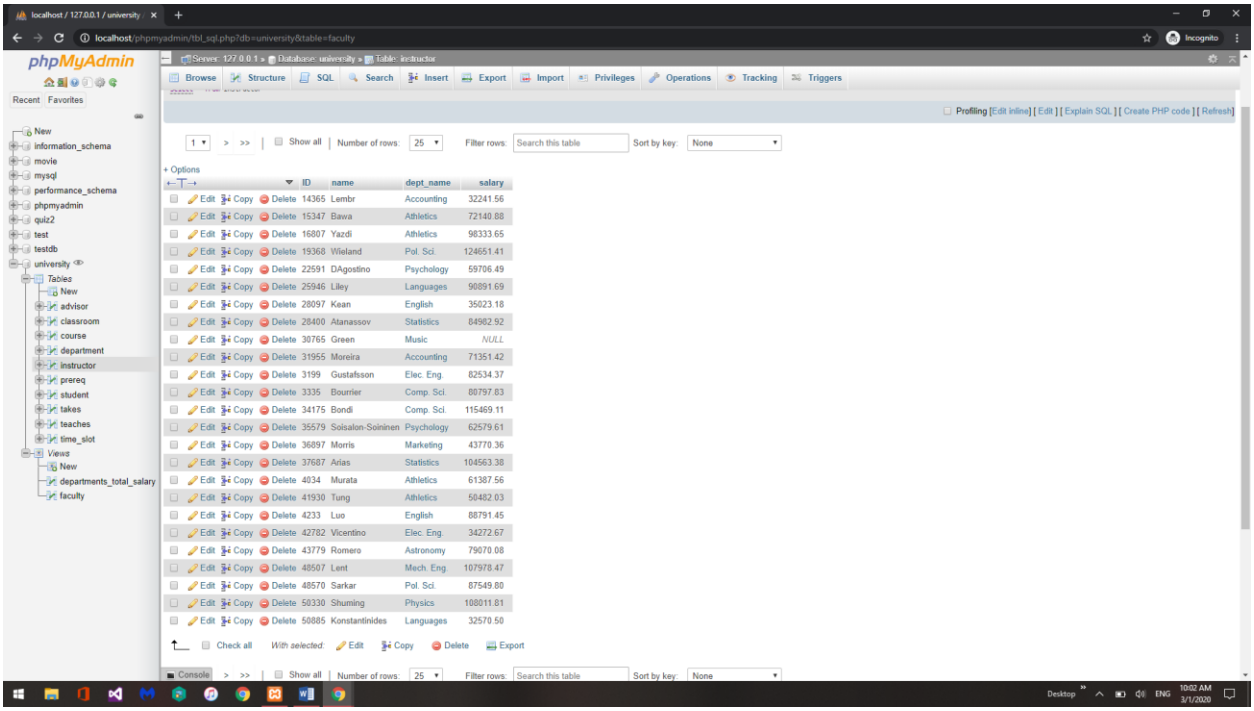


Chris Grimes
CS 33007
Homework 3

6. Create a view of department salary totals create view departments_total_salary(dept_name, total_salary) as select dept_name, sum(salary) from instructor group by dept_name;

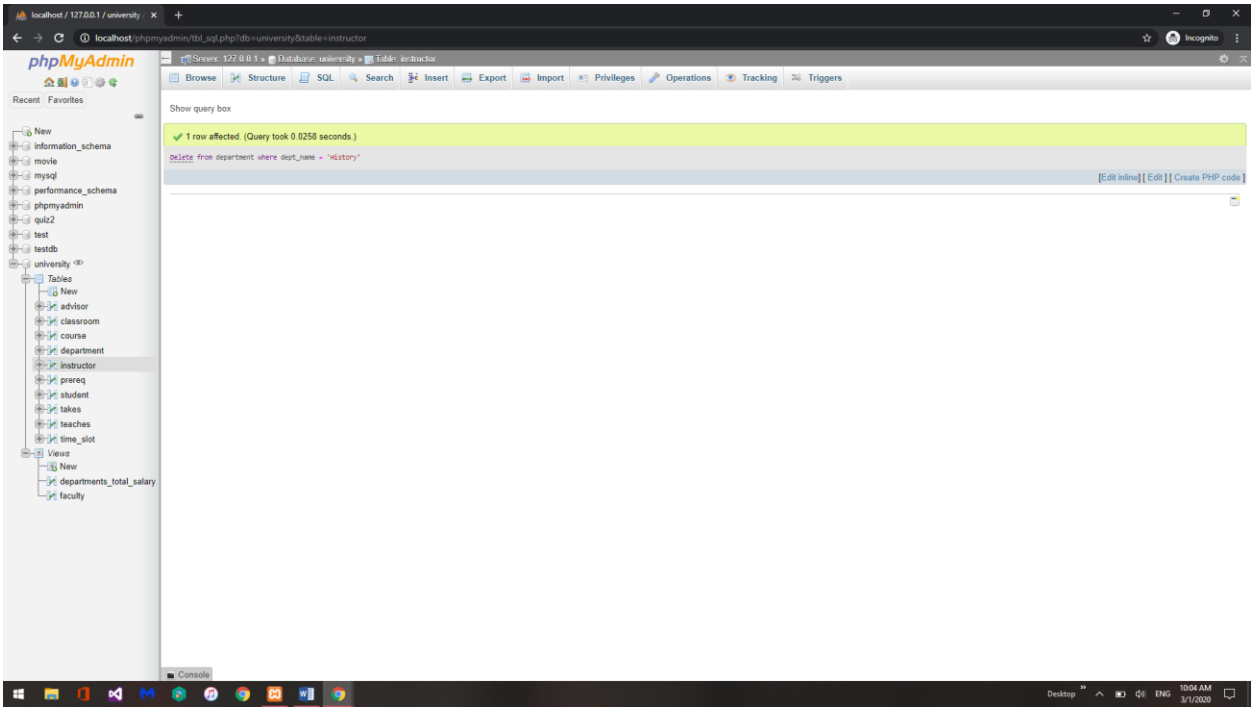


7. Delete a record from department table and show the effect on instructor table. (Effect of integrity constraints)- report the changes. Select * from instructor; // observe records of instructor from history department

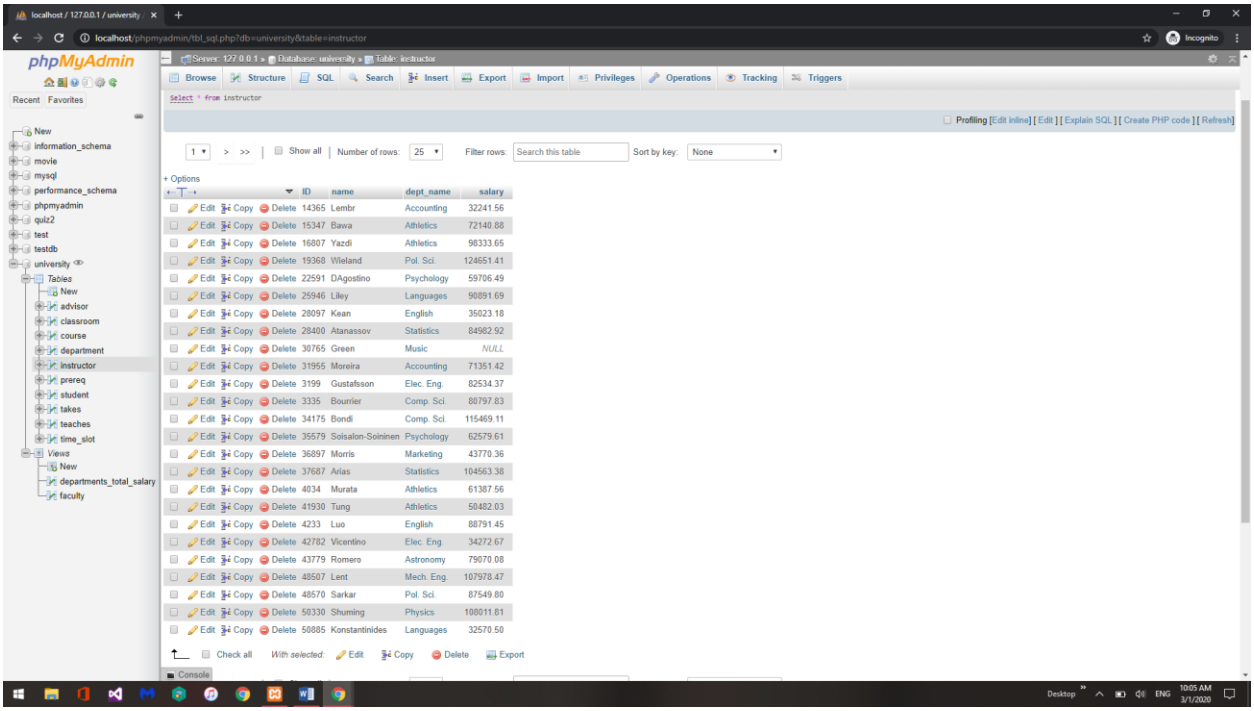


Chris Grimes
CS 33007
Homework 3

Delete from department where dept_name = “History”;



Select * from instructor; // again observe records of instructor from history department

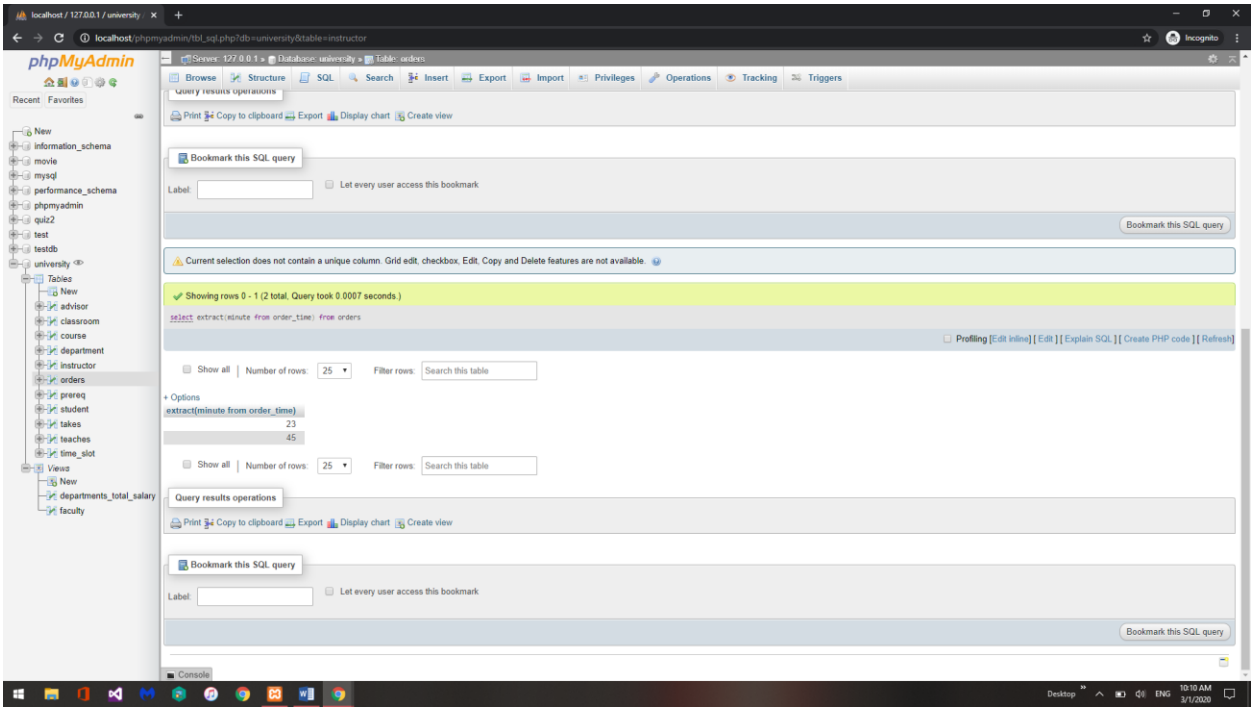


The changes made by this query are that there are no longer any records in the instructor table with an instructor form the history department.

Chris Grimes
CS 33007
Homework 3

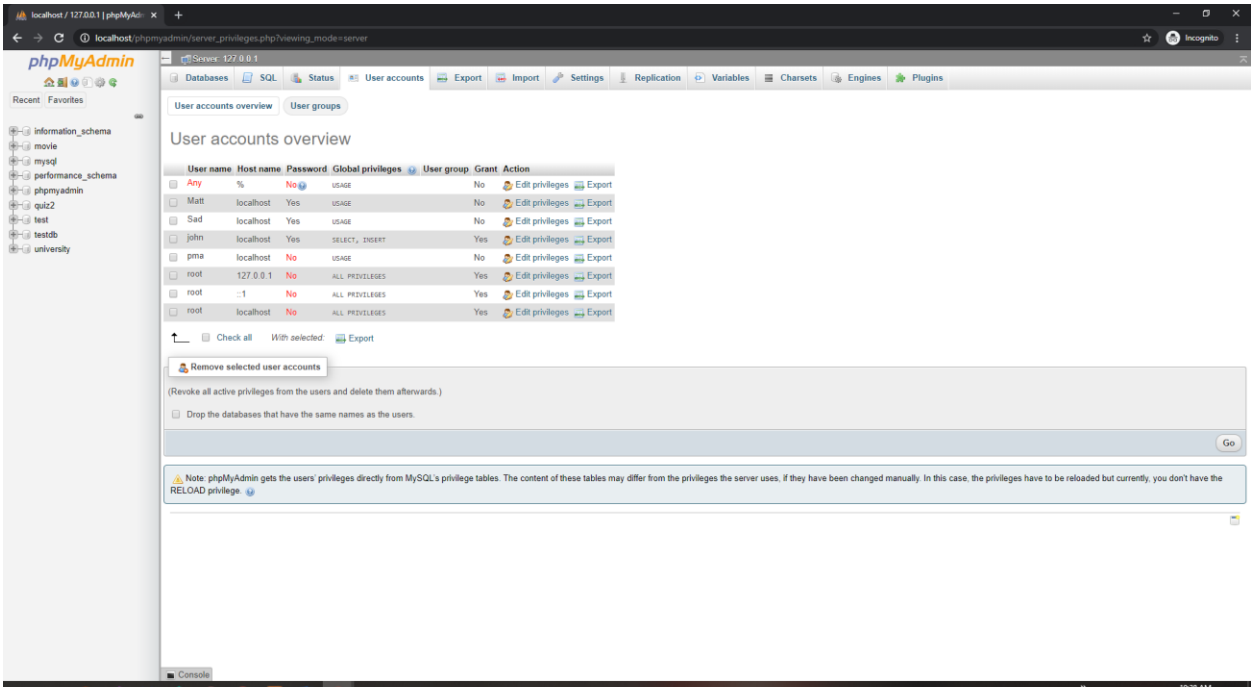
8. Create a transaction table, insert few records, extract minutes from transaction time.

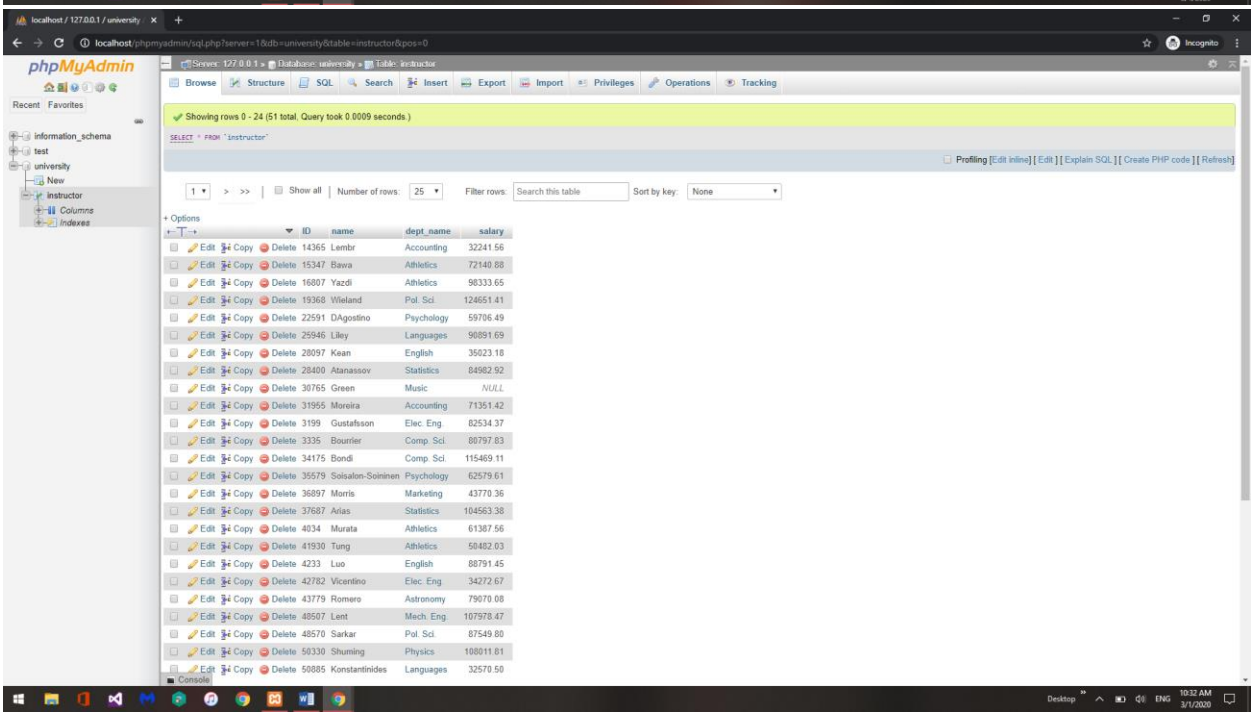
create table orders(order_id char(30), order_date date, order _time time, order _timestm timestamp);
insert into orders values("345","2018-09-06","5:23:56", "2018-09-06 5:23:56.34")
insert into orders values("6778","2018-09-06","5:45:56", "2018-09-06 5:45:56.34")
select * from orders
select extract(minute from order_time) from orders



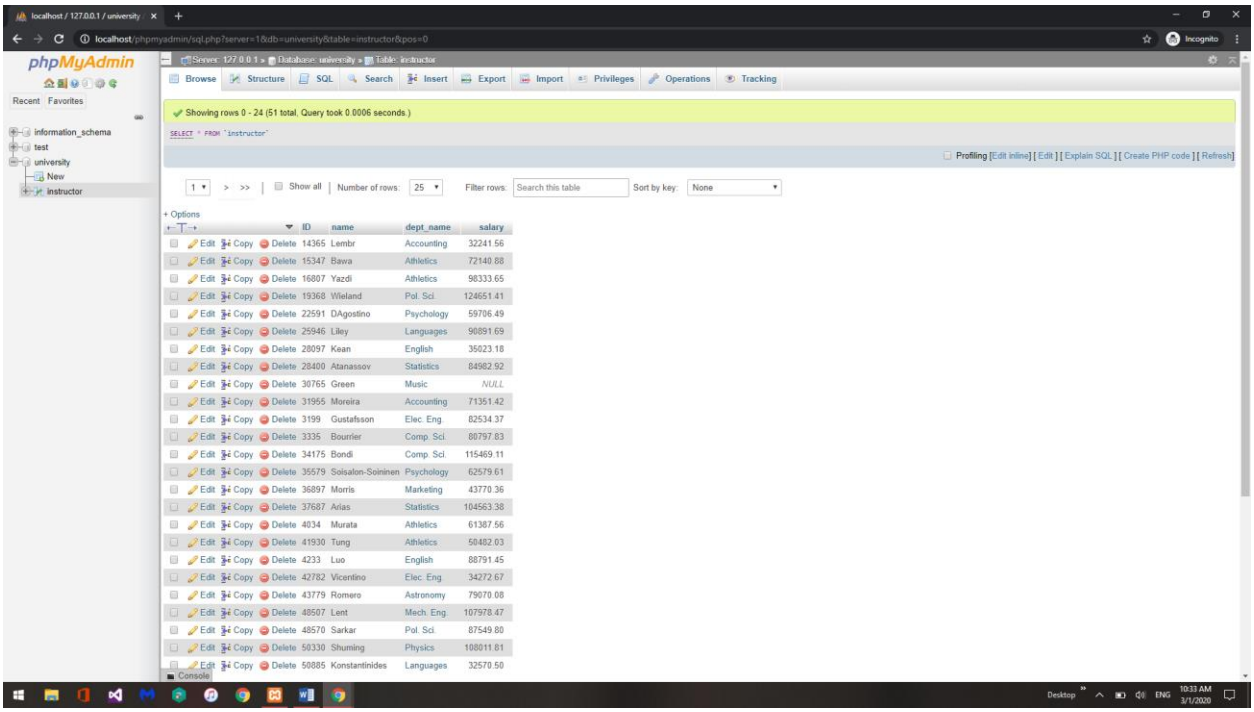
Chris Grimes
CS 33007
Homework 3

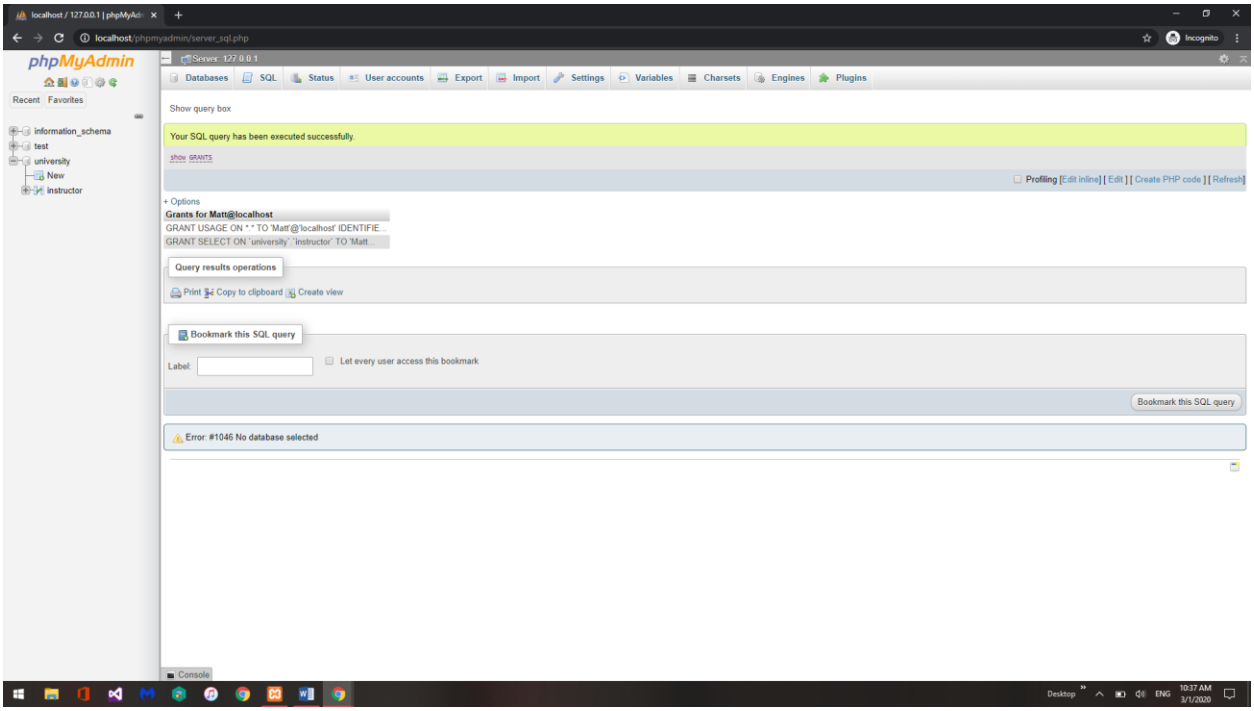
Part 2:



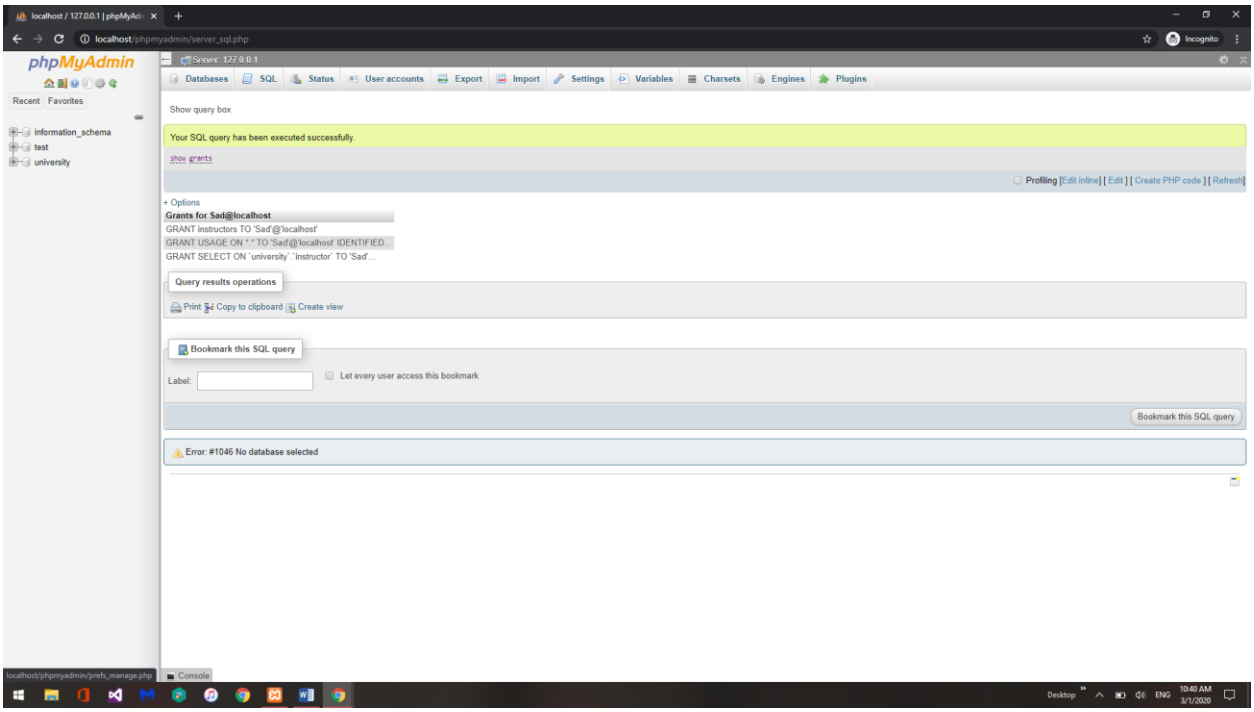


Chris Grimes
CS 33007
Homework 3





Chris Grimes
CS 33007
Homework 3



Part 3:

1. Assume you are given two relations, student(name, rollno) and marks(rollno, exam, mark) Show names of all students who have got marks in at least two exams.

   select name from student natural join marks where (select (rollno) from marks where mark is not null);

2. Create a view CSinstructors, showing all information about instructors from the Comp. Sci. department.

   create view CSinstructors as select * from instructor where dept_name = 'Comp. Sci.';

3. Create a transaction query that the increase the salary of Biology instructors by 3% and Physics instructors by 5%.

   START TRANSACTION;
   UPDATE instructor SET salary = salary* 1.03 where dept_name = 'Biology';
   UPDATE instructor SET salary = salary* 1.05 where dept_name = 'Physics';
   COMMIT;

Chris Grimes
CS 33007
Homework 3

4. Grant permission to one of your friends to view all data in your student relation. Also make sure that you are granting your friend to pass the permission to others.

   GRANT SELECT on university.student TO cgrimes91@gmail.com with GRANT option

5. (i) create a role CS_staff

create role CS_staff;

(ii) Allow CS_staff to only see/read data from the view CSinstructors created in exercise 2.

GRANT SELECT on university.CSinstructors to CS_staff;

(iii) Add user Matt (you created before in part 2) as CS_staff.

GRANT CS_staff to Matt@localhost;

(iv) Login as Matt and show he can only execute select queries on CSinstructors