**Database, (SOC-3020), Fall 2017**

**Homework#1, Group 004**

**Name:** **ID:**

Maftuna Sharabbaeva U1510067

Akhror Gaibnazarov U1510005

Azizbek Tursunov U1510009

**Date:** 10/17/2017

**Answer for question #1:**

I used phpMyAdmin to build a database which has name as “University DB”. I created it by uploading ‘DDL.sql’ and ‘smallRelationsInsertFile.sql’ filles. However when I imported, there were some syntax errors in DDL.sql file. Such as in 21, 31, 47, 58, 70, 82, 94, 112, 114 lines actually when there is foreigh key in referencing relations assigning foreign keys are lost. For example in line 112 “foreign key (course\_id) references course” is written which is syntax error. Here should be foreign key (course\_id) references course (course\_id). However in line114 should be written (course\_id) instead of (prereq\_id). In ‘smallRelationsInsertFile.sql’ ther is not any syntax error. To be visible I provide that code with changes which will be highlighted by bold characters:

In DDL.sql:

create table classroom

(building varchar(15),

room\_number varchar(7),

capacity numeric(4,0),

primary key (building, room\_number)

);

create table department

(dept\_name varchar(20),

building varchar(15),

budget numeric(12,2) check (budget > 0),

primary key (dept\_name)

);

create table course

(course\_id varchar(8),

title varchar(50),

dept\_name varchar(20),

credits numeric(2,0) check (credits > 0),

primary key (course\_id),

foreign key (dept\_name) references department **(dept\_name)**

on delete set null

);

create table instructor

(ID varchar(5),

name varchar(20) not null,

dept\_name varchar(20),

salary numeric(8,2) check (salary > 29000),

primary key (ID),

foreign key (dept\_name) references department **(dept\_name)**

on delete set null

);

create table section

(course\_id varchar(8),

sec\_id varchar(8),

semester varchar(6)

check (semester in ('Fall', 'Winter', 'Spring', 'Summer')),

year numeric(4,0) check (year > 1701 and year < 2100),

building varchar(15),

room\_number varchar(7),

time\_slot\_id varchar(4),

primary key (course\_id, sec\_id, semester, year),

foreign key (course\_id) references course **(course\_id)**

on delete cascade,

foreign key (building, room\_number) references classroom **(building, room\_number)**

on delete set null

);

create table teaches

(ID varchar(5),

course\_id varchar(8),

sec\_id varchar(8),

semester varchar(6),

year numeric(4,0),

primary key (ID, course\_id, sec\_id, semester, year),

foreign key (course\_id,sec\_id, semester, year) references section **(course\_id,sec\_id, semester, year)**

on delete cascade,

foreign key (ID) references instructor **(ID)**

on delete cascade

);

create table student

(ID varchar(5),

name varchar(20) not null,

dept\_name varchar(20),

tot\_cred numeric(3,0) check (tot\_cred >= 0),

primary key (ID),

foreign key (dept\_name) references department **(dept\_name)**

on delete set null

);

create table takes

(ID varchar(5),

course\_id varchar(8),

sec\_id varchar(8),

semester varchar(6),

year numeric(4,0),

grade varchar(2),

primary key (ID, course\_id, sec\_id, semester, year),

foreign key (course\_id,sec\_id, semester, year) references section **(course\_id,sec\_id, semester, year)**

on delete cascade,

foreign key (ID) references student **(ID)**

on delete cascade

);

create table advisor

(s\_ID varchar(5),

i\_ID varchar(5),

primary key (s\_ID),

foreign key (i\_ID) references instructor **(ID)**

on delete set null,

foreign key (s\_ID) references student **(ID)**

on delete cascade

);

create table time\_slot

(time\_slot\_id varchar(4),

day varchar(1),

start\_hr numeric(2) check (start\_hr >= 0 and start\_hr < 24),

start\_min numeric(2) check (start\_min >= 0 and start\_min < 60),

end\_hr numeric(2) check (end\_hr >= 0 and end\_hr < 24),

end\_min numeric(2) check (end\_min >= 0 and end\_min < 60),

primary key (time\_slot\_id, day, start\_hr, start\_min)

);

create table prereq

(course\_id varchar(8),

prereq\_id varchar(8),

primary key (course\_id, prereq\_id),

foreign key (course\_id) references course **(course\_id)**

on delete cascade,

foreign key (prereq\_id) references course **(course\_id)**

);

And in ‘smallRelationsInsertFile.sql’ :

delete from prereq;

delete from time\_slot;

delete from advisor;

delete from takes;

delete from student;

delete from teaches;

delete from section;

delete from instructor;

delete from course;

delete from department;

delete from classroom;

insert into classroom values ('Packard', '101', '500');

insert into classroom values ('Painter', '514', '10');

insert into classroom values ('Taylor', '3128', '70');

insert into classroom values ('Watson', '100', '30');

insert into classroom values ('Watson', '120', '50');

insert into department values ('Biology', 'Watson', '90000');

insert into department values ('Comp. Sci.', 'Taylor', '100000');

insert into department values ('Elec. Eng.', 'Taylor', '85000');

insert into department values ('Finance', 'Painter', '120000');

insert into department values ('History', 'Painter', '50000');

insert into department values ('Music', 'Packard', '80000');

insert into department values ('Physics', 'Watson', '70000');

insert into course values ('BIO-101', 'Intro. to Biology', 'Biology', '4');

insert into course values ('BIO-301', 'Genetics', 'Biology', '4');

insert into course values ('BIO-399', 'Computational Biology', 'Biology', '3');

insert into course values ('CS-101', 'Intro. to Computer Science', 'Comp. Sci.', '4');

insert into course values ('CS-190', 'Game Design', 'Comp. Sci.', '4');

insert into course values ('CS-315', 'Robotics', 'Comp. Sci.', '3');

insert into course values ('CS-319', 'Image Processing', 'Comp. Sci.', '3');

insert into course values ('CS-347', 'Database System Concepts', 'Comp. Sci.', '3');

insert into course values ('EE-181', 'Intro. to Digital Systems', 'Elec. Eng.', '3');

insert into course values ('FIN-201', 'Investment Banking', 'Finance', '3');

insert into course values ('HIS-351', 'World History', 'History', '3');

insert into course values ('MU-199', 'Music Video Production', 'Music', '3');

insert into course values ('PHY-101', 'Physical Principles', 'Physics', '4');

insert into instructor values ('10101', 'Srinivasan', 'Comp. Sci.', '65000');

insert into instructor values ('12121', 'Wu', 'Finance', '90000');

insert into instructor values ('15151', 'Mozart', 'Music', '40000');

insert into instructor values ('22222', 'Einstein', 'Physics', '95000');

insert into instructor values ('32343', 'El Said', 'History', '60000');

insert into instructor values ('33456', 'Gold', 'Physics', '87000');

insert into instructor values ('45565', 'Katz', 'Comp. Sci.', '75000');

insert into instructor values ('58583', 'Califieri', 'History', '62000');

insert into instructor values ('76543', 'Singh', 'Finance', '80000');

insert into instructor values ('76766', 'Crick', 'Biology', '72000');

insert into instructor values ('83821', 'Brandt', 'Comp. Sci.', '92000');

insert into instructor values ('98345', 'Kim', 'Elec. Eng.', '80000');

insert into section values ('BIO-101', '1', 'Summer', '2009', 'Painter', '514', 'B');

insert into section values ('BIO-301', '1', 'Summer', '2010', 'Painter', '514', 'A');

insert into section values ('CS-101', '1', 'Fall', '2009', 'Packard', '101', 'H');

insert into section values ('CS-101', '1', 'Spring', '2010', 'Packard', '101', 'F');

insert into section values ('CS-190', '1', 'Spring', '2009', 'Taylor', '3128', 'E');

insert into section values ('CS-190', '2', 'Spring', '2009', 'Taylor', '3128', 'A');

insert into section values ('CS-315', '1', 'Spring', '2010', 'Watson', '120', 'D');

insert into section values ('CS-319', '1', 'Spring', '2010', 'Watson', '100', 'B');

insert into section values ('CS-319', '2', 'Spring', '2010', 'Taylor', '3128', 'C');

insert into section values ('CS-347', '1', 'Fall', '2009', 'Taylor', '3128', 'A');

insert into section values ('EE-181', '1', 'Spring', '2009', 'Taylor', '3128', 'C');

insert into section values ('FIN-201', '1', 'Spring', '2010', 'Packard', '101', 'B');

insert into section values ('HIS-351', '1', 'Spring', '2010', 'Painter', '514', 'C');

insert into section values ('MU-199', '1', 'Spring', '2010', 'Packard', '101', 'D');

insert into section values ('PHY-101', '1', 'Fall', '2009', 'Watson', '100', 'A');

insert into teaches values ('10101', 'CS-101', '1', 'Fall', '2009');

insert into teaches values ('10101', 'CS-315', '1', 'Spring', '2010');

insert into teaches values ('10101', 'CS-347', '1', 'Fall', '2009');

insert into teaches values ('12121', 'FIN-201', '1', 'Spring', '2010');

insert into teaches values ('15151', 'MU-199', '1', 'Spring', '2010');

insert into teaches values ('22222', 'PHY-101', '1', 'Fall', '2009');

insert into teaches values ('32343', 'HIS-351', '1', 'Spring', '2010');

insert into teaches values ('45565', 'CS-101', '1', 'Spring', '2010');

insert into teaches values ('45565', 'CS-319', '1', 'Spring', '2010');

insert into teaches values ('76766', 'BIO-101', '1', 'Summer', '2009');

insert into teaches values ('76766', 'BIO-301', '1', 'Summer', '2010');

insert into teaches values ('83821', 'CS-190', '1', 'Spring', '2009');

insert into teaches values ('83821', 'CS-190', '2', 'Spring', '2009');

insert into teaches values ('83821', 'CS-319', '2', 'Spring', '2010');

insert into teaches values ('98345', 'EE-181', '1', 'Spring', '2009');

insert into student values ('00128', 'Zhang', 'Comp. Sci.', '102');

insert into student values ('12345', 'Shankar', 'Comp. Sci.', '32');

insert into student values ('19991', 'Brandt', 'History', '80');

insert into student values ('23121', 'Chavez', 'Finance', '110');

insert into student values ('44553', 'Peltier', 'Physics', '56');

insert into student values ('45678', 'Levy', 'Physics', '46');

insert into student values ('54321', 'Williams', 'Comp. Sci.', '54');

insert into student values ('55739', 'Sanchez', 'Music', '38');

insert into student values ('70557', 'Snow', 'Physics', '0');

insert into student values ('76543', 'Brown', 'Comp. Sci.', '58');

insert into student values ('76653', 'Aoi', 'Elec. Eng.', '60');

insert into student values ('98765', 'Bourikas', 'Elec. Eng.', '98');

insert into student values ('98988', 'Tanaka', 'Biology', '120');

insert into takes values ('00128', 'CS-101', '1', 'Fall', '2009', 'A');

insert into takes values ('00128', 'CS-347', '1', 'Fall', '2009', 'A-');

insert into takes values ('12345', 'CS-101', '1', 'Fall', '2009', 'C');

insert into takes values ('12345', 'CS-190', '2', 'Spring', '2009', 'A');

insert into takes values ('12345', 'CS-315', '1', 'Spring', '2010', 'A');

insert into takes values ('12345', 'CS-347', '1', 'Fall', '2009', 'A');

insert into takes values ('19991', 'HIS-351', '1', 'Spring', '2010', 'B');

insert into takes values ('23121', 'FIN-201', '1', 'Spring', '2010', 'C+');

insert into takes values ('44553', 'PHY-101', '1', 'Fall', '2009', 'B-');

insert into takes values ('45678', 'CS-101', '1', 'Fall', '2009', 'F');

insert into takes values ('45678', 'CS-101', '1', 'Spring', '2010', 'B+');

insert into takes values ('45678', 'CS-319', '1', 'Spring', '2010', 'B');

insert into takes values ('54321', 'CS-101', '1', 'Fall', '2009', 'A-');

insert into takes values ('54321', 'CS-190', '2', 'Spring', '2009', 'B+');

insert into takes values ('55739', 'MU-199', '1', 'Spring', '2010', 'A-');

insert into takes values ('76543', 'CS-101', '1', 'Fall', '2009', 'A');

insert into takes values ('76543', 'CS-319', '2', 'Spring', '2010', 'A');

insert into takes values ('76653', 'EE-181', '1', 'Spring', '2009', 'C');

insert into takes values ('98765', 'CS-101', '1', 'Fall', '2009', 'C-');

insert into takes values ('98765', 'CS-315', '1', 'Spring', '2010', 'B');

insert into takes values ('98988', 'BIO-101', '1', 'Summer', '2009', 'A');

insert into takes values ('98988', 'BIO-301', '1', 'Summer', '2010', null);

insert into advisor values ('00128', '45565');

insert into advisor values ('12345', '10101');

insert into advisor values ('23121', '76543');

insert into advisor values ('44553', '22222');

insert into advisor values ('45678', '22222');

insert into advisor values ('76543', '45565');

insert into advisor values ('76653', '98345');

insert into advisor values ('98765', '98345');

insert into advisor values ('98988', '76766');

insert into time\_slot values ('A', 'M', '8', '0', '8', '50');

insert into time\_slot values ('A', 'W', '8', '0', '8', '50');

insert into time\_slot values ('A', 'F', '8', '0', '8', '50');

insert into time\_slot values ('B', 'M', '9', '0', '9', '50');

insert into time\_slot values ('B', 'W', '9', '0', '9', '50');

insert into time\_slot values ('B', 'F', '9', '0', '9', '50');

insert into time\_slot values ('C', 'M', '11', '0', '11', '50');

insert into time\_slot values ('C', 'W', '11', '0', '11', '50');

insert into time\_slot values ('C', 'F', '11', '0', '11', '50');

insert into time\_slot values ('D', 'M', '13', '0', '13', '50');

insert into time\_slot values ('D', 'W', '13', '0', '13', '50');

insert into time\_slot values ('D', 'F', '13', '0', '13', '50');

insert into time\_slot values ('E', 'T', '10', '30', '11', '45 ');

insert into time\_slot values ('E', 'R', '10', '30', '11', '45 ');

insert into time\_slot values ('F', 'T', '14', '30', '15', '45 ');

insert into time\_slot values ('F', 'R', '14', '30', '15', '45 ');

insert into time\_slot values ('G', 'M', '16', '0', '16', '50');

insert into time\_slot values ('G', 'W', '16', '0', '16', '50');

insert into time\_slot values ('G', 'F', '16', '0', '16', '50');

insert into time\_slot values ('H', 'W', '10', '0', '12', '30');

insert into prereq values ('BIO-301', 'BIO-101');

insert into prereq values ('BIO-399', 'BIO-101');

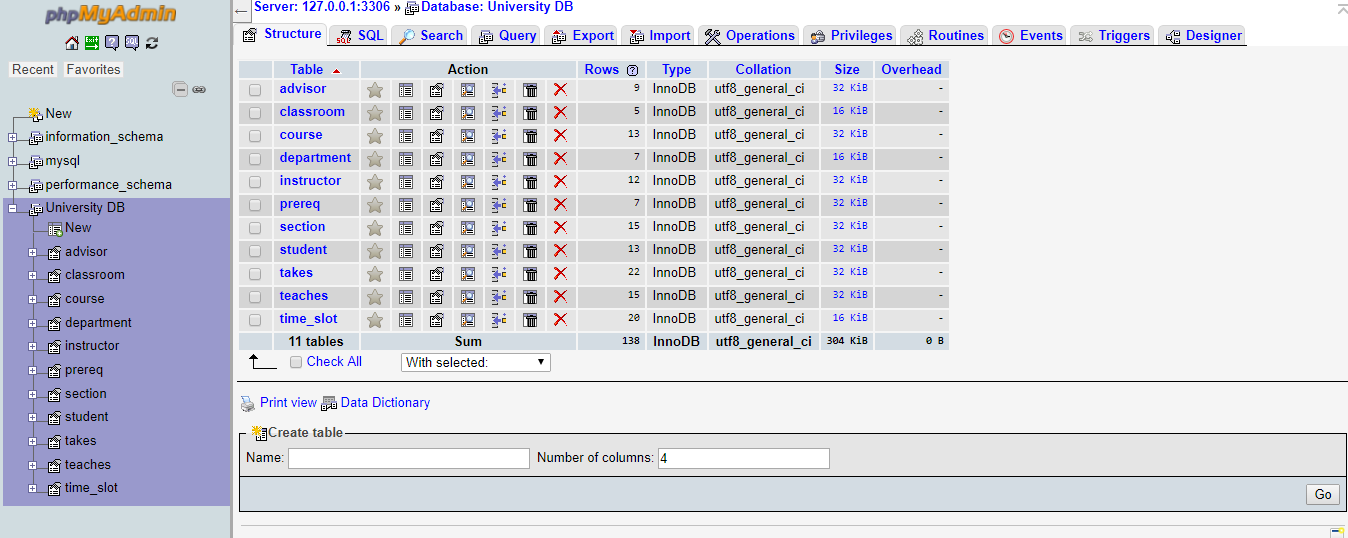
insert into prereq values ('CS-190', 'CS-101');

insert into prereq values ('CS-315', 'CS-101');

insert into prereq values ('CS-319', 'CS-101');

insert into prereq values ('CS-347', 'CS-101');

insert into prereq values ('EE-181', 'PHY-101');

After correcting them result is:

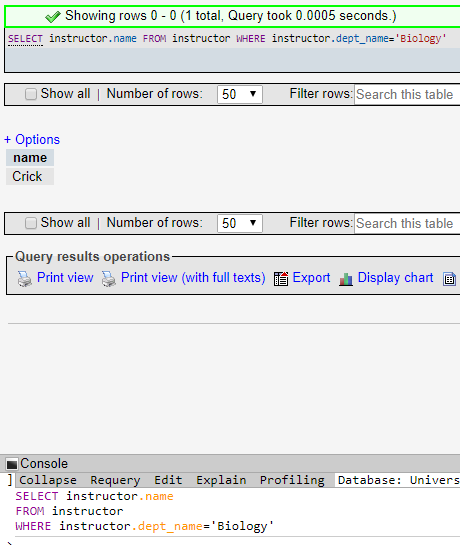
**Answer for question #2:**

**Answer of (i):**

SELECT instructor.name

FROM instructor

WHERE instructor.dept\_name='Biology';

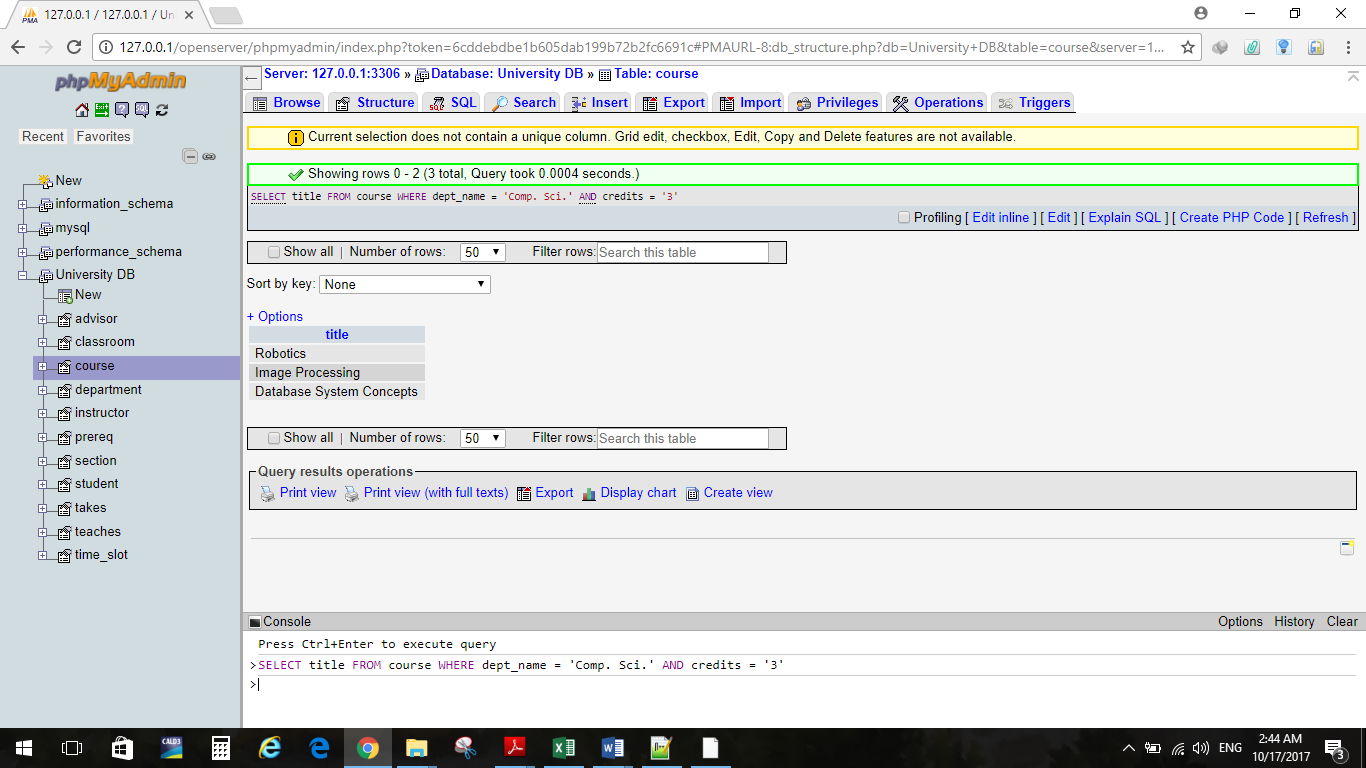
****

**Answer of (ii) :**

SELECT title

FROM course

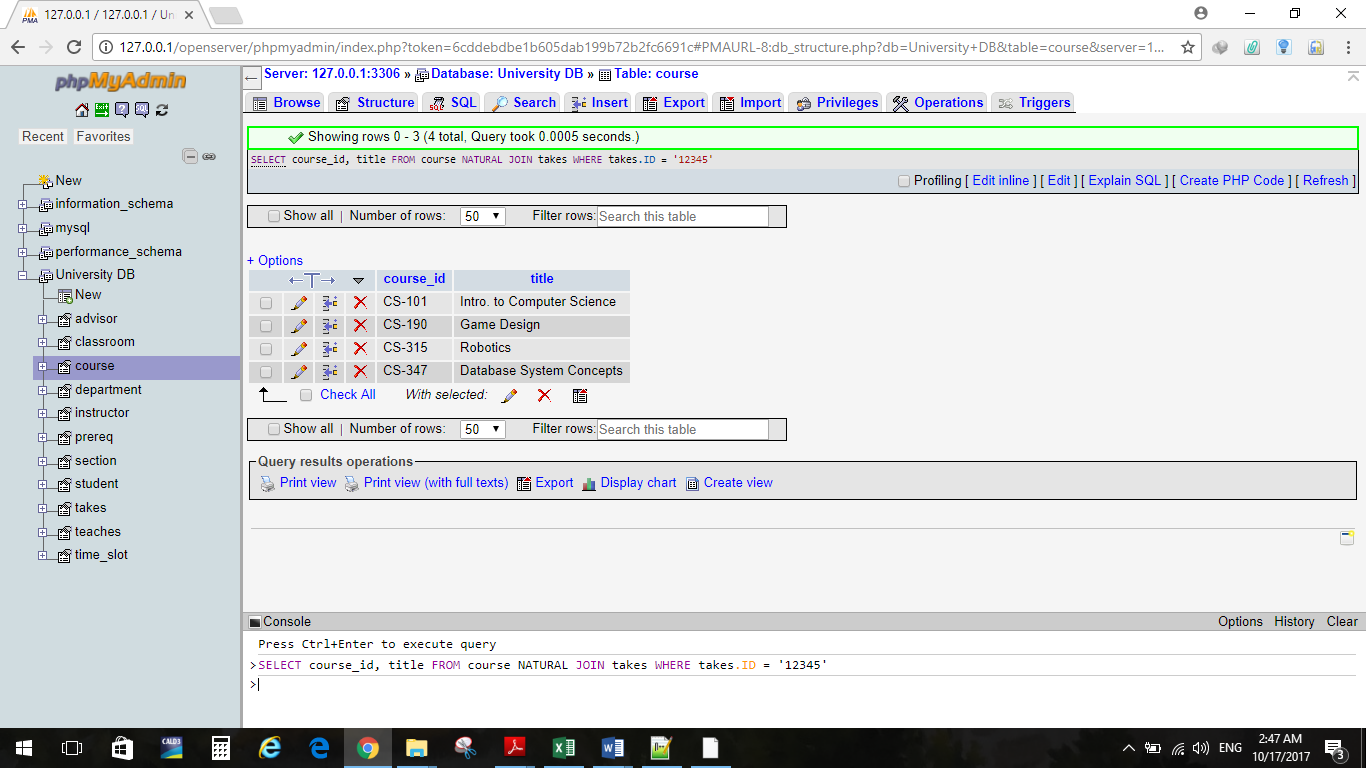
WHERE dept\_name = 'Comp.Sci.'

**** AND credits = '3';

**Answer of (iii) :**

SELECT course\_id, title

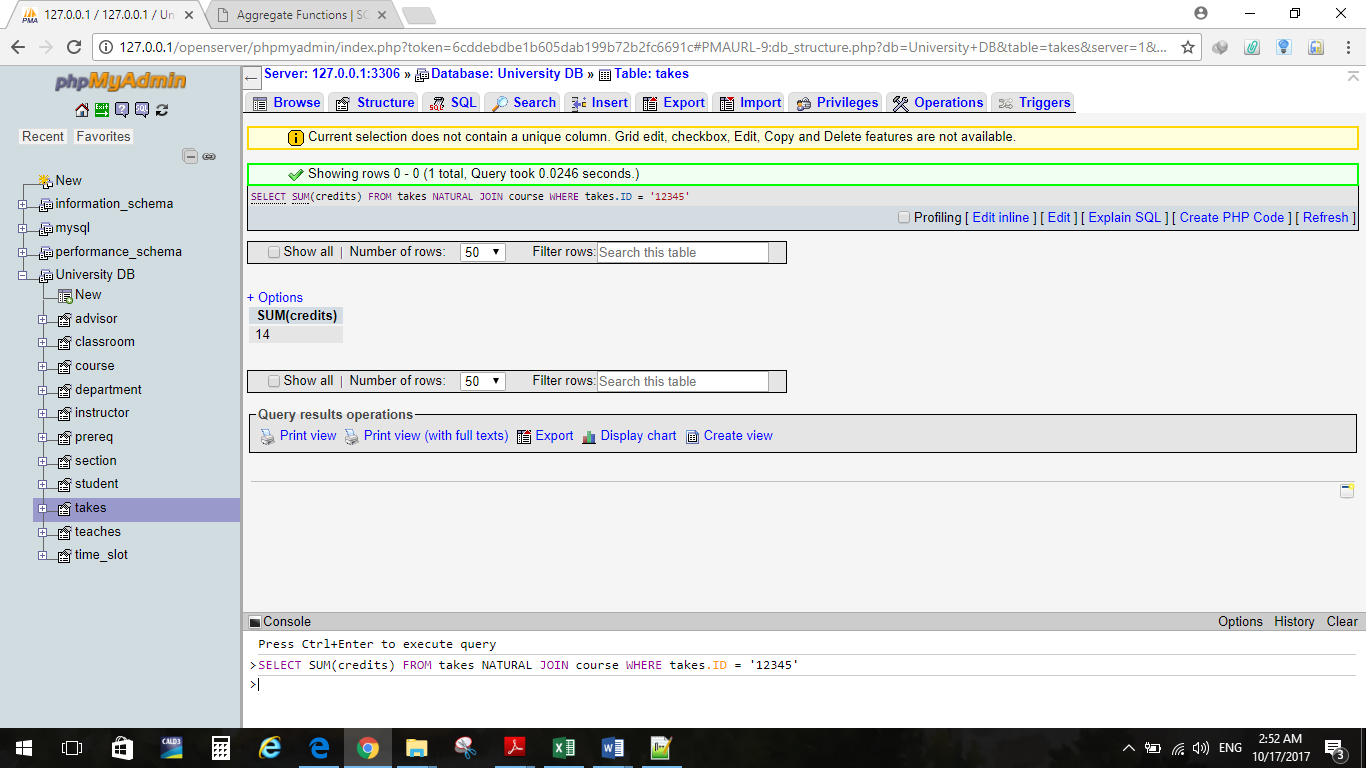
FROM course NATURAL JOIN takes

 WHERE takes.ID = '12345';

**Answer of (iv):**

SELECT SUM(credits)

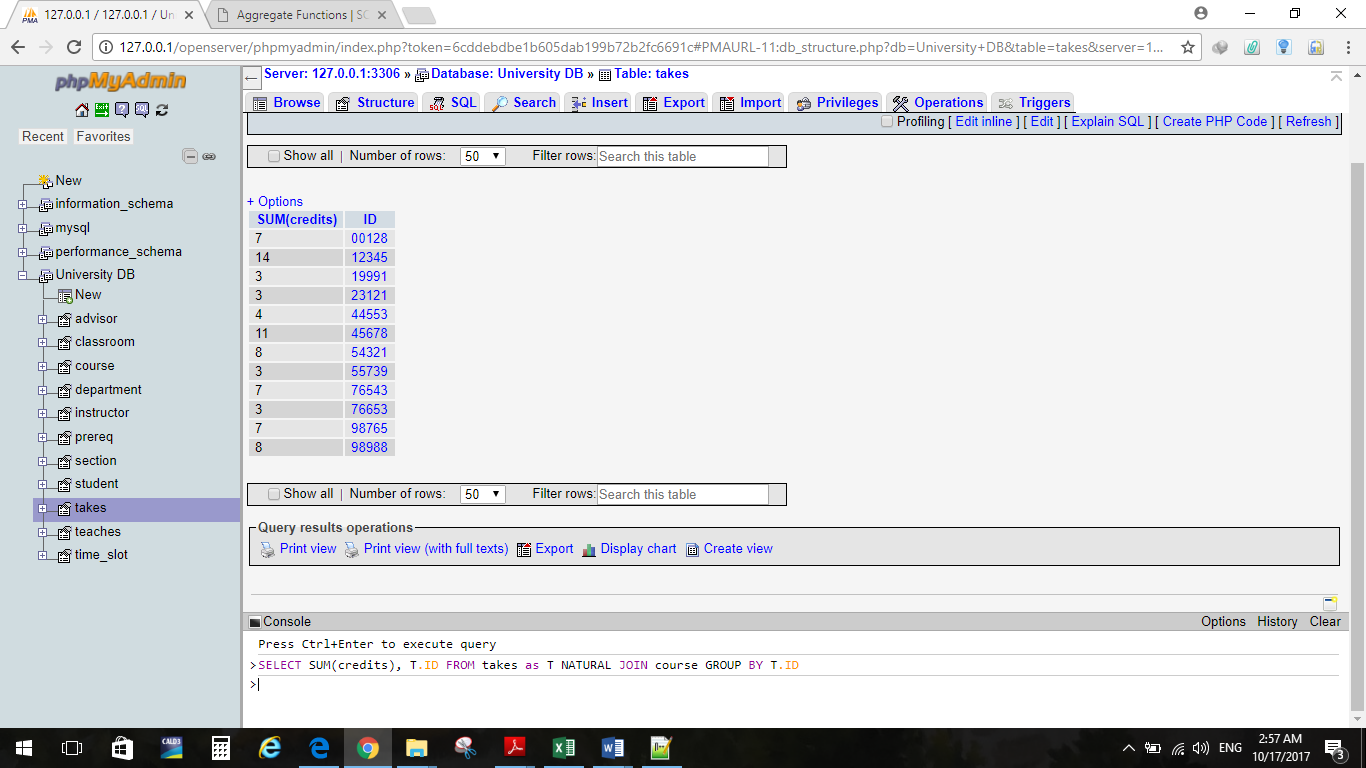
FROM takes NATURAL JOIN course

**** WHERE takes.ID = '12345'

**Answer of (v):**

SELECT SUM(credits), T.ID

FROM takes as T NATURAL JOIN course

**** GROUP BY T.ID

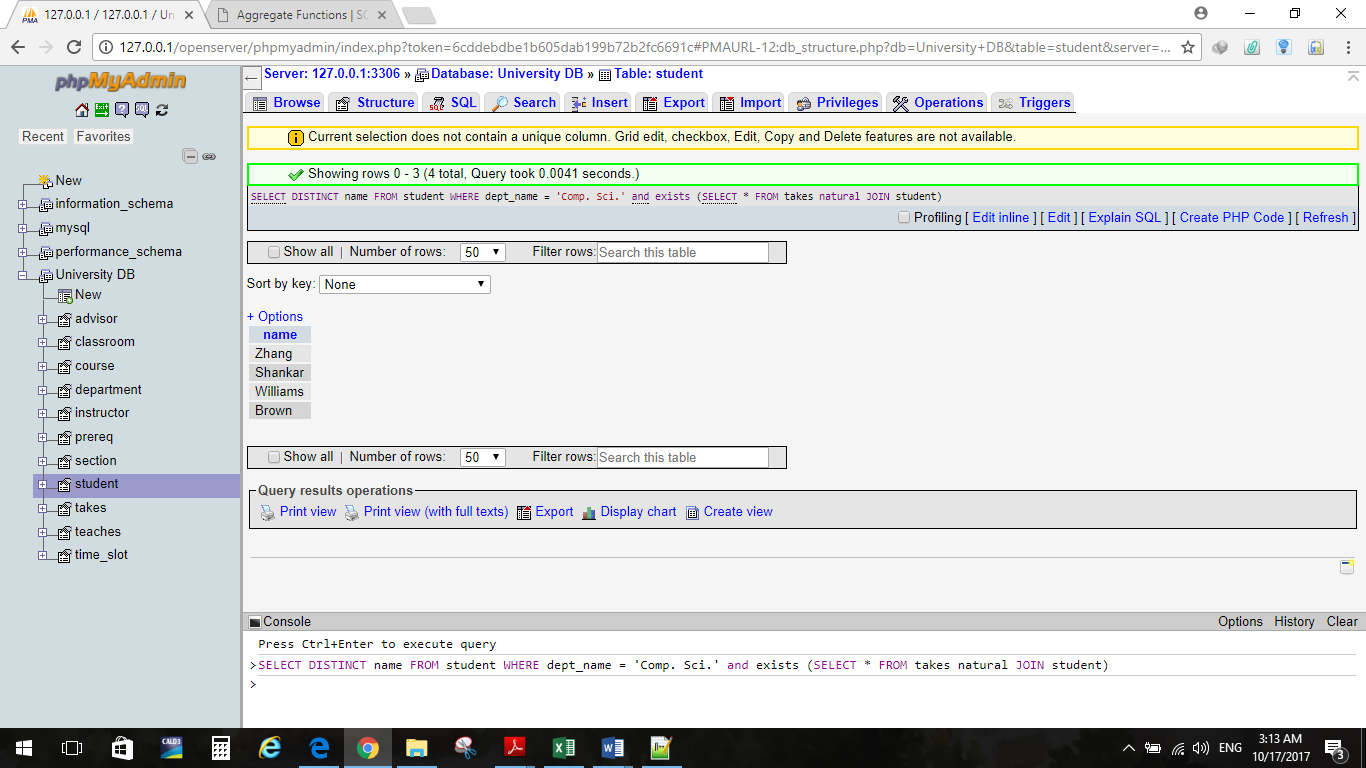
**Answer of (vi) :**

SELECT DISTINCT name

FROM student

WHERE dept\_name = 'Comp. Sci.' and

exists(SELECT \*

**** FROM takes natural JOIN student)

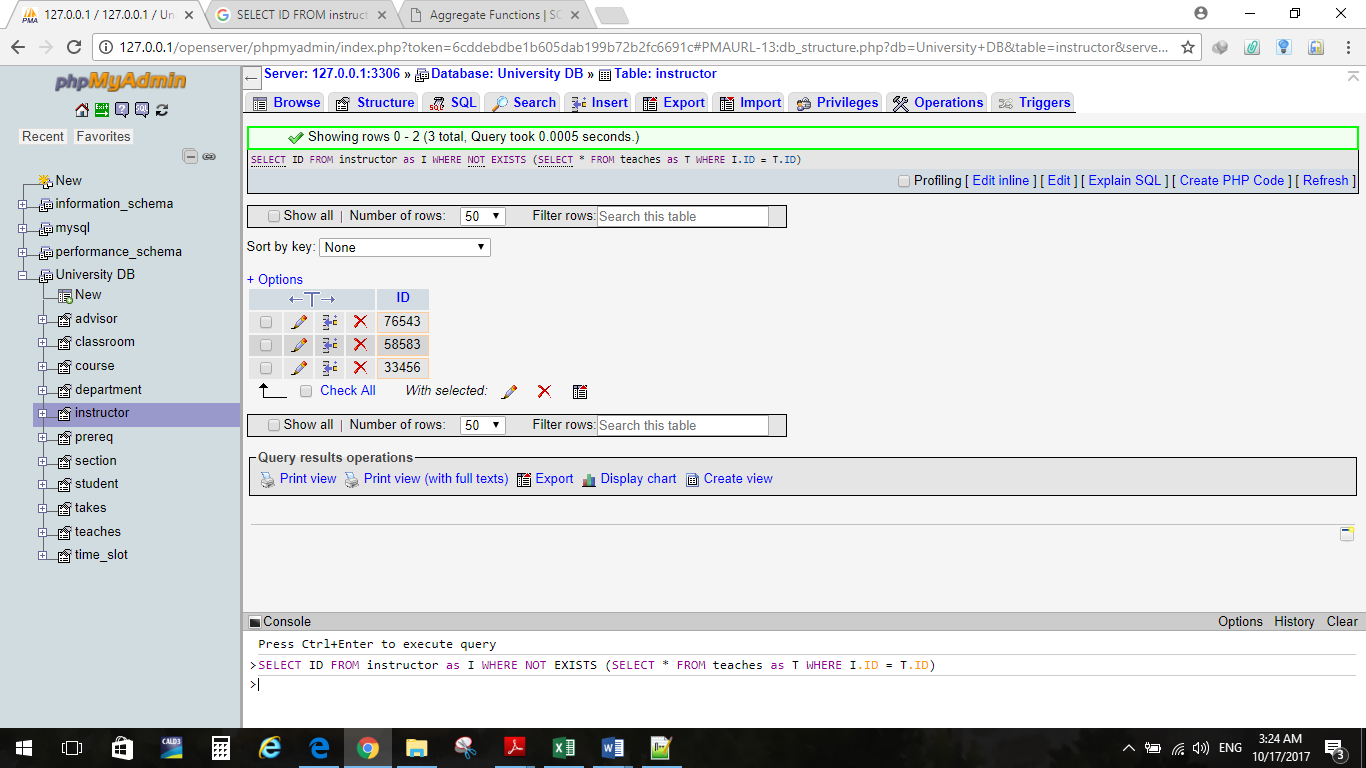
**Answer of (vii):**

SELECT ID

FROM instructor as I

WHERE NOT EXISTS (SELECT \*

FROM teaches as T

**** WHERE I.ID = T.ID);

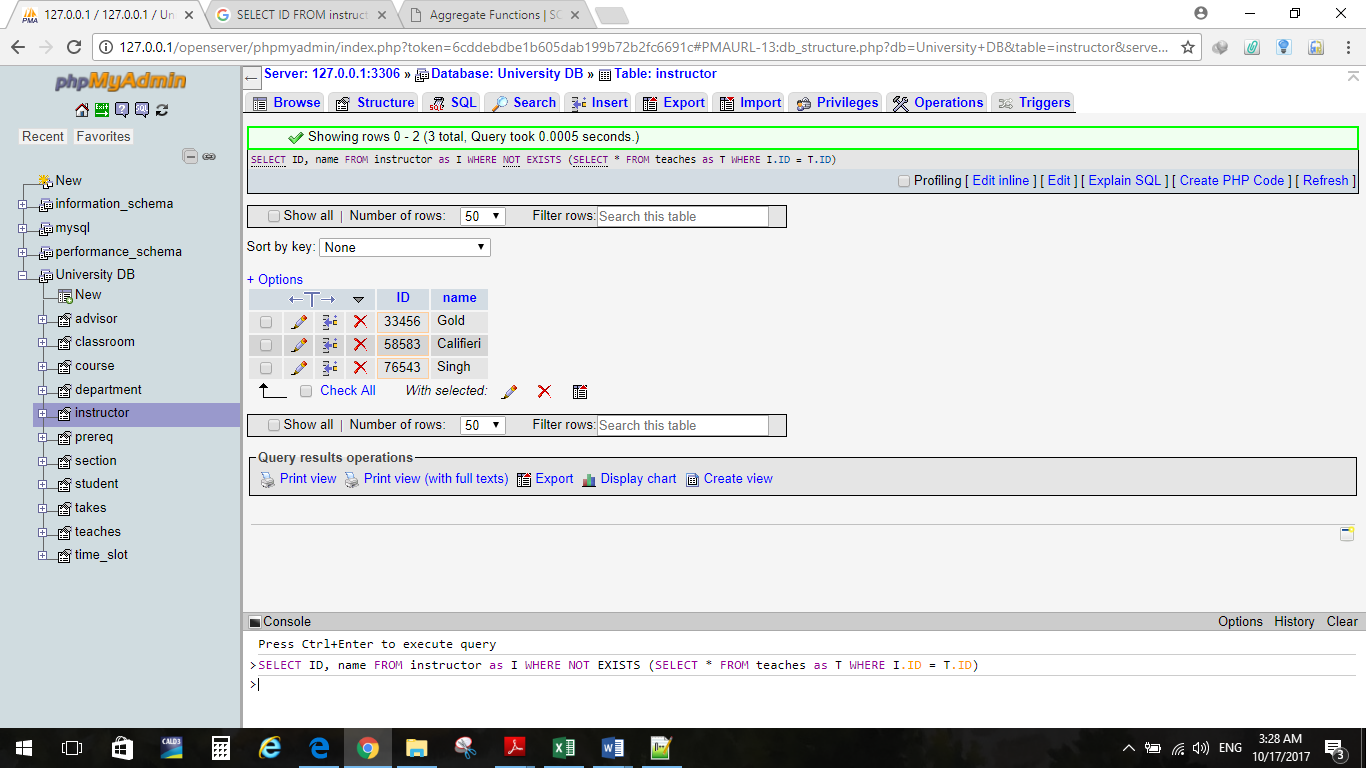
**Answer of (viii):**

SELECT ID, name

FROM instructor as I

WHERE NOT EXISTS (SELECT \*

FROM teaches as T

**** WHERE I.ID = T.ID)

**Answer for question #3:**

**(a)** List the entire student relation who took some course in or before 2010.

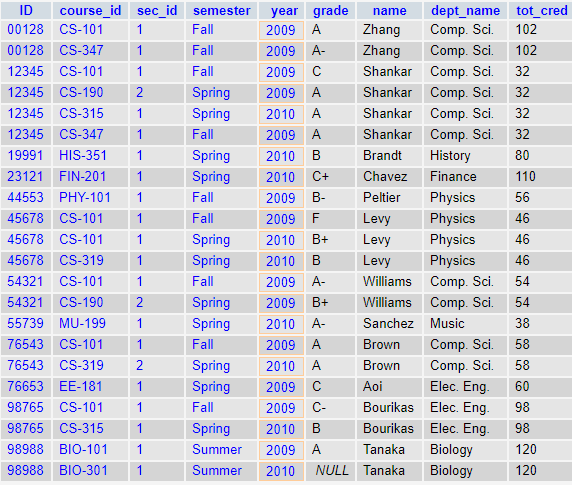
**(b)** σyear<=2010(takes + student)

**(c)** there is syntax error when running query. We should use ` instead of ‘. And the results is:

SELECT \*

from takes natural join student

where `year`<=2010



**Answer for question #4:**

List the names of students along with courses that they take since 2018.

Πname, course\_id (σyear>=2008(takes) + student)

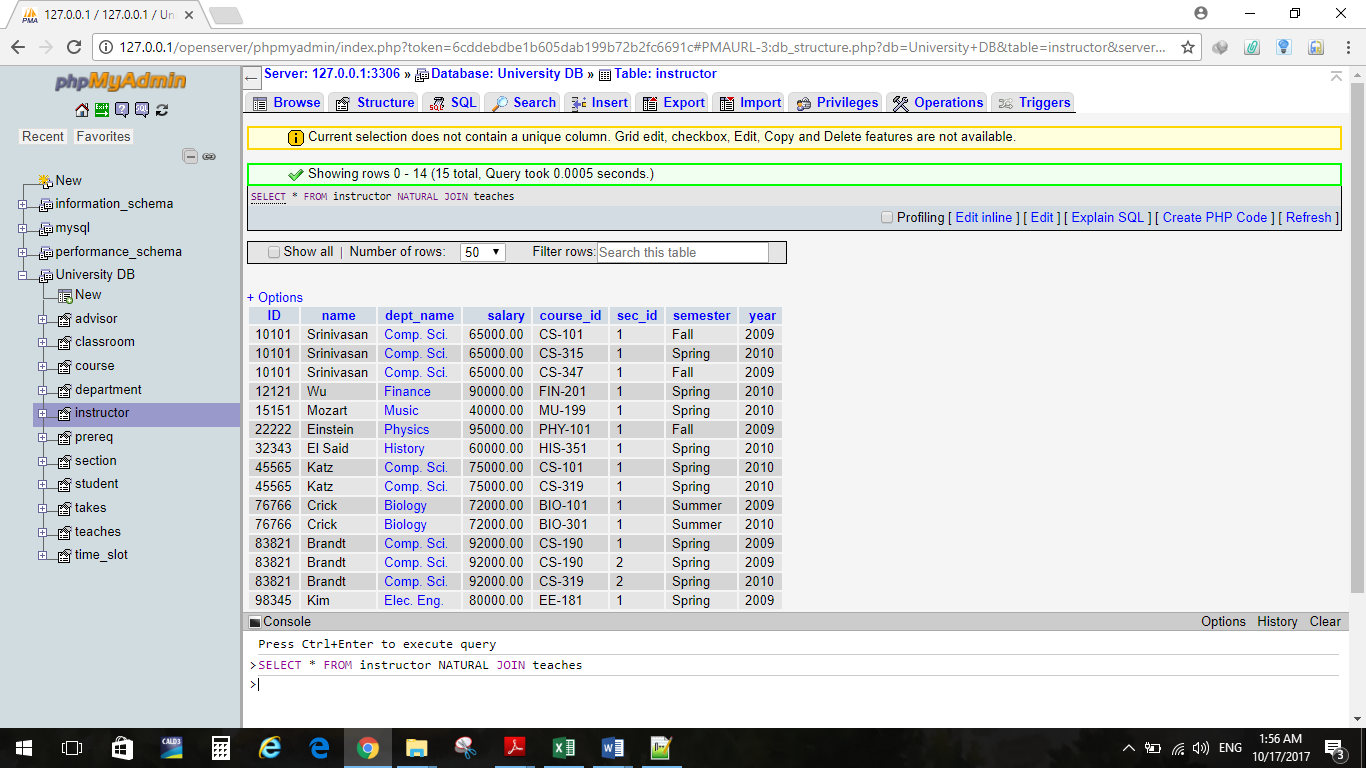
**Answer for question #5:**

1. Find names of students whose age is higher than 21.
2. Πperson\_name(σage>21(student))

**Answer for question #6:**

1. SELECT \*

FROM instructor NATURAL JOIN teaches

**(b):**

**(c)** SELECT \*

FROM instructor NATURAL JOIN teaches

WHERE (ID + year) < 85831

**Answer for question #7:**

1. Find the student’s distinct ID who have taken a course

section taught by the instructor with name ‘Einstein’.

1. SELECT DISTINCT student.ID

FROM (student JOIN takes USING(ID)) JOIN (instructor

JOIN teaches USING(ID)) USING (course\_id, sec\_id,

semester, year)

WHERE instructor.name = 'Einstein'



**Answer for question #8:**

1. Find the student’s ID who have taken a course section taught by the instructor with name ‘Einstein’ remove duplicates.
2. SELECT DISTINCT T.ID

FROM takes AS T

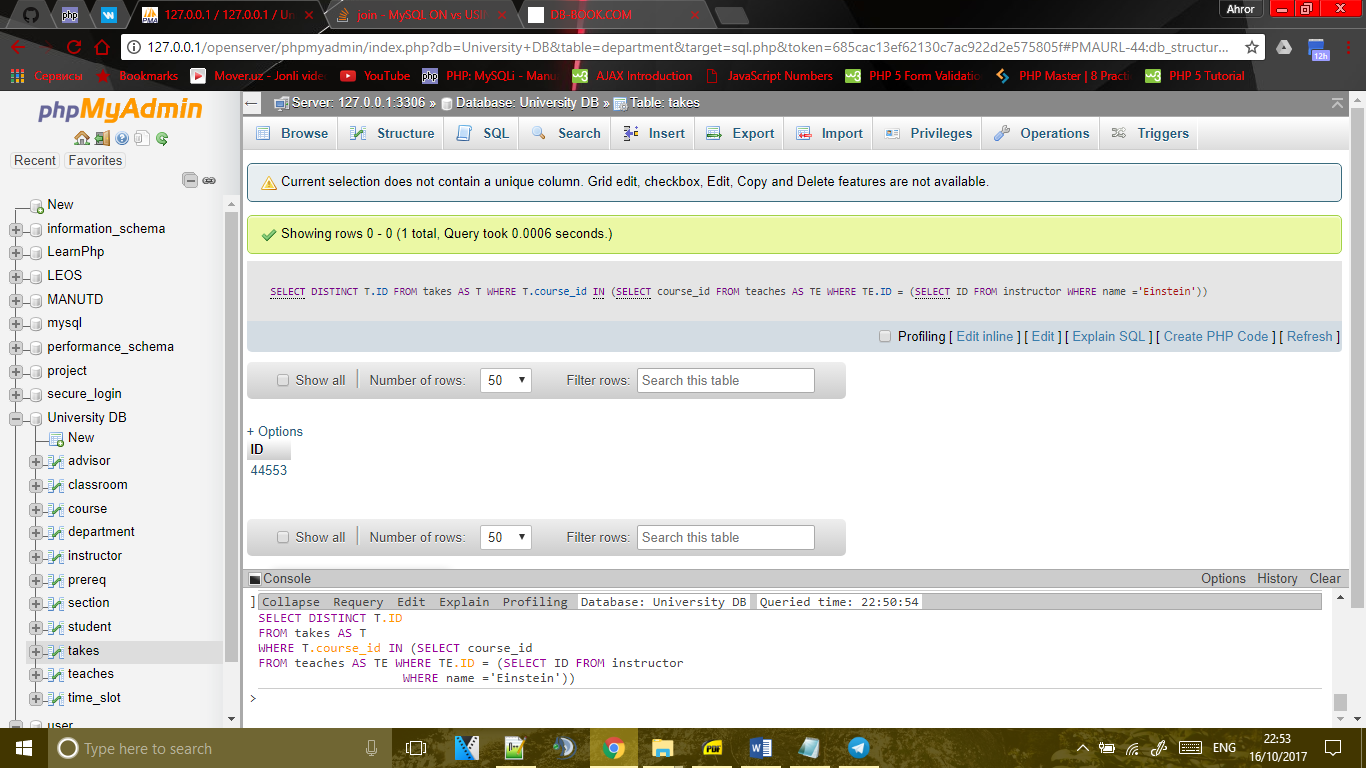
WHERE T.course\_id IN (SELECT course\_id

FROM teaches AS TE

WHERE TE.ID = (SELECT ID

FROM instructor

WHERE name ='Einstein'));



**Answer for question #9:**

They are not equal. Because in exercise 7 SQL query searches student ID with more columns. It means course\_id, sec\_id, semester, year is used to find student ID. However, in the 8th exercise query start to select ID which name equal to ‘Einstein’, then it finds course\_id from teaches where instructor ID is equal to TE.ID. Then it selects distinct ID from takes, where T.course\_id in course\_id which came from teaches. If information were many and semester and year had doubles, second way would be not useful.

**Answer for question #10:**

**(a)** Find the salaries of all instructors in computer science department whose salary is less than at least one instructor in the other departments and remove duplicates.

**(b)**

SELECT DISTINCT salary

FROM instructor

WHERE salary<some(SELECT salary

FROM instructor

where dept\_name NOT in

(SELECT salary

FROM instructor

WHERE dept\_name='Comp. Sci.'))