# DATABASE SYSTEMS

# Hassaan Akbar Cheema

# Reg no. 174351

# Bscs6c

# Lab 10

**LAB TASKS**

**Given the following database schema:**

**Student (snum: integer, sname: char(30), major: char(25), level: char(2), age: integer)**

**Faculty (fid: integer, fname: char(30), deptid: integer)**

**Class (cname: char(40), meets\_at: char(20), room: char(10), fid: integer | fid REFS**

**Faculty.fid)**

**Enrolled (snum: integer, cname: char(40) | snum REFS student.snum, cname REFS**

**class.name)**

**---------------------------------------------------------------------------------------------------------------------**

**TASK:**

**Write SQL expressions for each of the following create view queries and then use the view in other queries for executing them:**

**1. Create a view named “CSstudents” that retrieve all students whose major is “Computer Science”.**

**CREATE OR REPLACE VIEW CStudents AS**

**SELECT \* FROM student**

**WHERE major = 'Computer Science';**

**a. Retrieve age of the oldest student whose major is “Compter Science”.**

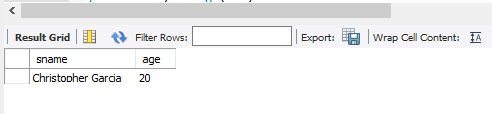
**SELECT MAX(age) FROM CStudents;**



**b. Find the name and age of the oldest student whose major = “Computer Science”**

**SELECT sname, age FROM CStudents**

**WHERE age = (SELECT MAX(age) FROM CStudents);**



**c. Find the names, majors and ages of all juniors (Level = JR) who are enrolled in a class**

**taught by Ivana Teach in “Computer Science” major.**

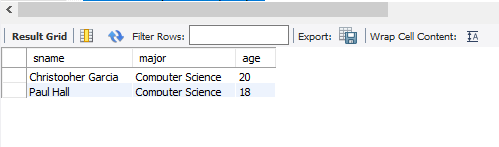
**SELECT sname,major,age FROM CStudents**

**join enrolled using (snum)**

**join class using (cname)**

**join faculty using (fid)**

**where fname like 'Ivana Teach' and level like'JR'**



**d. Find the names of faculty members and their departments, classes and room number**

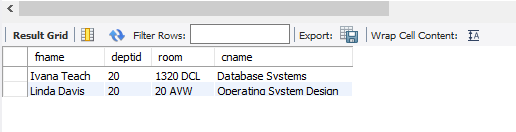
**which they teaching in “Computer Science” major..**

**SELECT distinct fname, deptid, room, cname FROM CStudents**

**join enrolled using (snum)**

**join class using (cname)**

**join faculty using (fid)**



**2. Define a view “ElderStudents” that retrieve oldest students in each major.**

**CREATE OR REPLACE VIEW ElderStudents AS**

**SELECT snum,sname,major,level, MAX(age) as max FROM student**

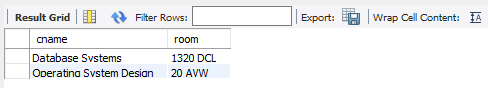
**GROUP BY major;**

**a. Find the names of classes and their rooms where elder students are studying.**

**SELECT distinct cname,room FROM ElderStudents**

**join enrolled using (snum)**

**join class using (cname) ;**

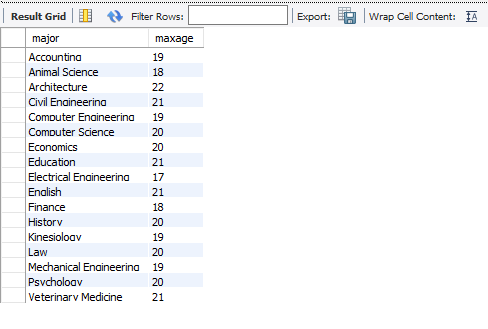


**b. Find name of major and age of student who includes the eldest student.**

**SELECT distinct cname,room FROM ElderStudents**

**join enrolled using (snum)**

**join class using (cname) ;**



**3. Define a view “EnrolledStudents” that retrieve students who are enrolled for a class.**

**Create view EnrolledStudents as**

**SELECT \* from student s**

**Where snum in (select snum from enrolled);**

**4. Find the names of all students who are enrolled in two classes that meet at the same time.**

**Select distinct s.sname**

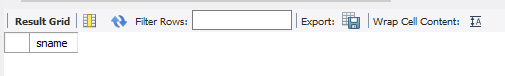
**From student s**

**Where s.snum in**

**(Select e1.snum**

**From enrolled e1, enrolled e2, class c1, class c2**

**Where e1.snum = e2.snum and e1.cname <> e2.cname and e1.cname = c1.cname and e2.cname = c2.cname and c1.meets\_at = c2.meets\_at)**



**5. Redefine the above query with “EnrolledStudents”.**

**create view EnrolledStudentstu as select \* from student s join enrolled e using (snum) ;**

**SELECT DISTINCT St.sname**

**FROM EnrolledStudentstu St**

**WHERE St.snum IN (SELECT E1.snum**

**FROM Enrolled E1, Enrolled E2, Class C1, Class C2**

**WHERE E1.snum = E2.snum AND E1.cname <> E2.cname**

**AND E1.cname = C1.cname**

**AND E2.cname = C2.cname AND C1.meets\_at = C2.meets\_at);**

