-- Student Faculty database

-- Creating Table

use student\_faculty;  
create table student(  
 snum int,  
 sname varchar(30),  
 major varchar(10),  
 lvl varchar(2),  
 age int,  
 constraint s\_sum Primary key (snum)  
);  
  
create table class(  
 cname varchar(30),  
 meetsat time,  
 room varchar(5),  
 fid int,  
 constraint c\_cname primary key(cname),  
 constraint c\_fid foreign key (fid) references faculty(fid)  
   
);  
  
create table enrolled(  
 snum int,  
 cname varchar(30),  
 constraint e\_snum foreign key (snum) references student(snum) on delete cascade on update cascade,   
 constraint e\_cname foreign key (cname) references class(cname) on delete cascade on update cascade  
);  
  
create table faculty(  
 fid int,  
 fname varchar(30),  
 deptid int,  
 constraint f\_fid primary key(fid)  
);

– Inserting into Table

Insert into student values(1,'s1','m1','JR',18);  
Insert into student values(2,'s2','m2','FR',16);  
Insert into student values(3,'s3','m1','SR',17);  
Insert into student values(4,'s4','m4','JR',19);  
Insert into student values(5,'s5','m3','FR',20);  
Insert into student values(6,'s6','m1','JR',18);  
Insert into student values(7,'s7','m1','SR',18);  
  
insert into class values('c1','9:00','r110',1);  
insert into class values('c2','10:00','r120',2);  
insert into class values('c3','11:00','r128',3);  
insert into class values('c4','9:00','r128',1);  
insert into class values('c5','10:00','r110',2);  
insert into class values('c6','10:00','r128',2);  
delete from class where cname='c6';  
insert into faculty values(1,'f1',1);  
insert into faculty values(2,'f2',2);  
insert into faculty values(3,'f3',3);  
 insert into enrolled values(1,'c1');  
insert into enrolled values(1,'c4');  
insert into enrolled values(1,'c2');  
insert into enrolled values(1,'c3');  
insert into enrolled values(2,'c2');  
insert into enrolled values(3,'c2');  
insert into enrolled values(4,'c2');  
insert into enrolled values(5,'c4');  
insert into enrolled values(5,'c3');  
delete from enrolled where snum=5;  
select \* from enrolled;

-- i Find the names of all Juniors (level = JR) who are enrolled in a class taught by  
select s.sname from student s where s.snum in (select distinct e.snum from enrolled e where e.cname in (select c.cname from class c where c.fid=1)) and s.lvl='JR';

-- ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.   
  
select c.cname from class c where c.room='R128' union select e.cname from enrolled e group by e.cname having count(e.snum)>=5;  
insert into enrolled values(5,'c2');  
select c.cname from class c where c.room='R128' union select e.cname from enrolled e group by e.cname having count(e.snum)>=5;  
delete from enrolled e where e.snum=5 and e.cname='c2';

-- iii. Find the names of all students who are enrolled in two classes that meet at the same time.  
select distinct(e1.snum) from enrolled e1,enrolled e2 where e1.snum=e2.snum and e1.cname!=e2.cname and (select c.meetsat from class c where c.cname=e1.cname)=(select c.meetsat from class c where c.cname=e2.cname);   
select \* from enrolled;  
  
  
-- iv. Find the names of faculty members who teach in every room in which some class is taught.  
select distinct(c.fid) from class c where not exists (select c1.room from class c1 where c1.room not in (select c2.room from class c2 where c2.fid=c.fid));  
  
-- v.Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.   
select f.fname from faculty f where (select count(e.snum) from enrolled e where e.cname in (select c.cname from class c where c.fid=f.fid))<=5;  
insert into enrolled values(5,'c2');  
select f.fname from faculty f where (select count(e.snum) from enrolled e where e.cname in (select c.cname from class c where c.fid=f.fid))<=5;  
  
  
-- vi.Find the names of students who are not enrolled in any class.  
select s.sname from student s where s.snum not in (select distinct(e.snum) from enrolled e);  
  
-- vii. For each age value that appears in Students, find the level value that appears most often. For example, if there are more FR level students aged 18 than SR, JR, or SO students aged 18, you should print the pair (18, FR).  
select distinct(st.age),st.lvl from student st where st.lvl = (select s.lvl from student s where s.age=st.age group by s.lvl order by count(s.lvl) desc limit 1);