Keys
- SuperKey
  - a set of attributes whose values together uniquely identify a tuple in a relation
- Candidate Key
  - a superkey for which no proper subset is a superkey...a key that is minimal.
  - Can be more than one for a relation
- Primary Key
  - a candidate key chosen to be the main key for the relation.
  - One for each relation
- Keys can be composite

Examples:
- e.g.: Staff(lecturer, roomno, appraiser)
  - SK = {lecturer, roomno, appraiser},
    {lecturer, roomno}, {lecturer, appraiser},
    {roomno, appraiser}, {lecturer} and {roomno}
- CK = {lecturer} and {roomno}
- PK = {lecturer}

Foreign Key
- a (set of) attribute(s) in a relation that exactly matches a (primary) key in another relation
- the names of the attributes don’t have to be the same but must be of the same domain
- a foreign key in a relation A matching a primary key in a relation B represents a many:one relationship between A and B

Examples:
- Student(studno, name, tutor, year)
- Staff(lecturer, roomno, appraiser)

Relationship -v- Relation
- "an association between several entities represented by a Relationship Type of which there will be many Relationship Instances"

Joins
- e.g., get studno, name and tutor’s roomno for each student
SELECT a.studno, a.name, b.roomno FROM STUDENT a, STAFF b
WHERE a.tutor=b.lecturer

Join as Path
More SQL

Cinema (cid, name, city)
Film (fid, title, director)
Showing (fid, cid)

Name, city and the number of different films showing at each cinema, in ascending alphabetical order of city and name

```
select name, city, count(*) as films
```
select name, city, count(*) as films from Film f, Cinema c, Showing s

select name, city, count(*) as films from Film f, Cinema c, Showing s
where f.fid=s.fid and c.cid=s.cid

group by c.cid

order by city, name;

Cinema (cid, name, city)
Film (fid, title, director)
Showing (fid, cid)

Titles of films showing at every cinema listed in database.

(select * from showing s where s.cid=c.cid and f.fid=s.fid)
(select *
from cinema c
where not exists
  (select *
   from showing s
   where s.cid=c.cid and f.fid=s.fid))

select title
from film f
where not exists
  (select *
   from cinema c
   where not exists
     (select *
      from showing s
      where s.cid=c.cid and f.fid=s.fid));