

## 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*

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}

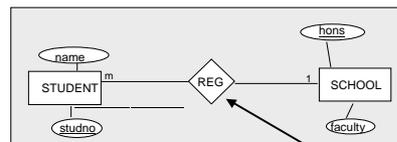
## Foreign Key

- 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

Student(studno,name,tutor,year)

Staff(lecturer,roomno,appraiser)

## Relationship -v- Relation



Relationship

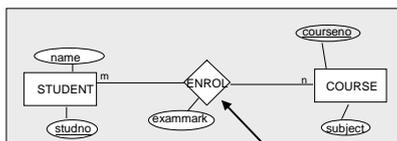
"an association between several entities represented by a *Relationship Type* of which there will be many *Relationship Instances*"

STUDENT (studno, name, honors)

SCHOOL(honors, faculty)

Relation (table)

## Relationship -v- Relation



ENROL Relationship

STUDENT (studno, name)

ENROL(studno, courseno, exammark)

COURSE(courseno, subject)

ENROL Relation

## Joins

STUDENT		
studno	name	tutor
s1	jones	bush
s2	brown	kahn
s3	smith	goble

STAFF	
lecturer	roomno
kahn	IT206
bush	2.26

e.g., get studno, name and tutor's roomno for each student

STUDENT			STAFF	
studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	goble		

STUDENT			STAFF	
studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	goble		

$\pi_{studno, name, roomno} (STUDENT \times_{tutor=lecturer} STAFF)$

studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s1	jones	bush	bush	2.26
s2	brown	kahn	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	bush	kahn	IT206
s3	smith	bush	bush	2.26

SELECT  
 FROM STUDENT a, STAFF b  
 WHERE

STUDENT			STAFF	
studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	goble		

$\pi_{studno, name, roomno} (STUDENT \bowtie_{tutor=lecturer} STAFF)$

studno	name	tutor	lecturer	roomno
s1	jones	bush	bush	2.26
s2	brown	kahn	kahn	IT206
s3	smith	bush	bush	2.26

SELECT  
 FROM STUDENT a, STAFF b  
 WHERE a.tutor=b.lecturer

STUDENT			STAFF	
studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	goble		

$\pi_{studno, name, roomno} (STUDENT \bowtie_{tutor=lecturer} STAFF)$

studno	name	tutor	lecturer	roomno
s1	jones			2.26
s2	brown			IT206
s3	smith			2.26

SELECT  
 a.studno, a.name, b.roomno  
 FROM STUDENT a, STAFF b  
 WHERE a.tutor=b.lecturer

STUDENT			STAFF	
studno	name	tutor	lecturer	roomno
s1	jones	bush	kahn	IT206
s2	brown	kahn	bush	2.26
s3	smith	goble		

$\sigma_{studno='s1'} (\pi_{studno, name, roomno} (STUDENT \bowtie_{tutor=lecturer} STAFF))$

studno	name	tutor	lecturer	roomno
s1	jones			2.26

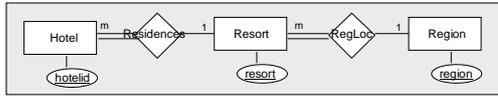
SELECT  
 a.studno, a.name, b.roomno  
 FROM STUDENT a, STAFF b  
 WHERE a.tutor=b.lecturer and studno='s1'

### Join as Path

Get hotels and their regions

SELECT  
 FROM  
 WHERE

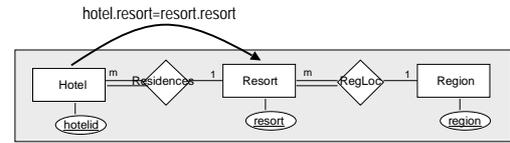
HOTEL (hotelid, name, resort)  
 RESORT (resort, region)  
 REGION (region, country)



Get hotels and their regions

HOTEL (hotelid, name, resort)  
 RESORT (resort, region)  
 REGION (region, country)

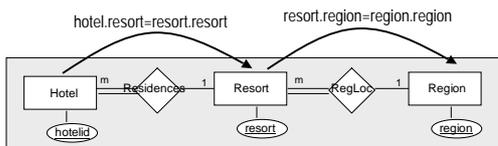
SELECT  
 FROM  
 HOTEL, RESORT, REGION  
 WHERE



Get hotels and their regions

HOTEL (hotelid, name, resort)  
 RESORT (resort, region)  
 REGION (region, country)

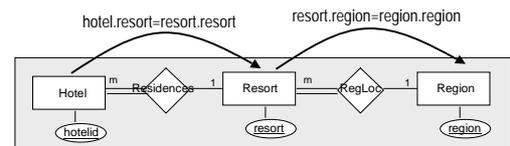
SELECT  
 FROM  
 HOTEL, RESORT, REGION  
 WHERE  
 hotel.resort=resort.resort



Get hotels and their regions

HOTEL (hotelid, name, resort)  
 RESORT (resort, region)  
 REGION (region, country)

SELECT  
 FROM  
 HOTEL, RESORT, REGION  
 WHERE  
 hotel.resort=resort.resort and  
 resort.region=region.region

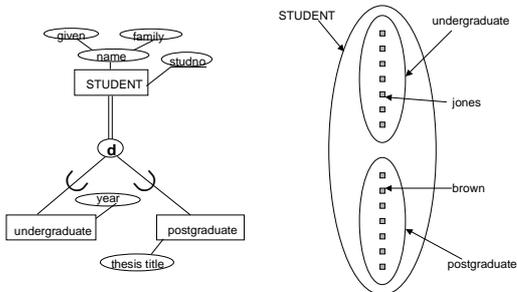


Get hotels and their regions

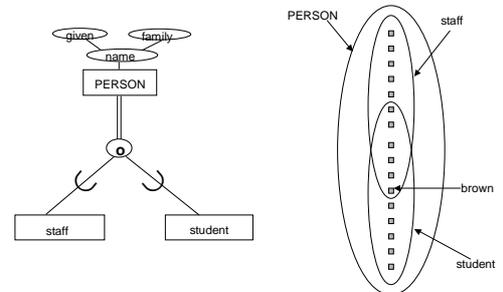
HOTEL (hotelid, name, resort)  
 RESORT (resort, region)  
 REGION (region, country)

SELECT  
 hotel.name, region.region  
 FROM  
 HOTEL, RESORT, REGION  
 WHERE  
 hotel.resort=resort.resort and  
 resort.region=region.region

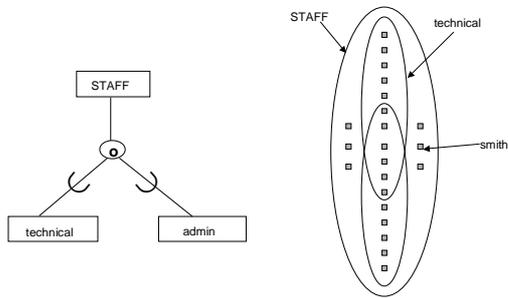
### Sub and Super-Types (Classes)



### Sub and Super-Types (Classes)



## Sub and Super-Types (Classes)



## Referential Integrity

- Student(studno,name,tutor,year)
- Staff(lecturer,roomno,appraiser)
- CASCADE
  - delete all matching foreign key tuples  
e.g. STUDENT
- RESTRICT
  - can't delete primary key tuple STAFF whilst a foreign key tuple STUDENT matches
- NULLIFY
  - foreign key STUDENT.tutor set to *null* if the foreign key ids allowed to take on null

## Lab Extensions

- Completed work must be handed in by 9:30 on the day of the subsequent lab!