CS3DB3/SE4DB3/ SE6DB3 TUTORIAL
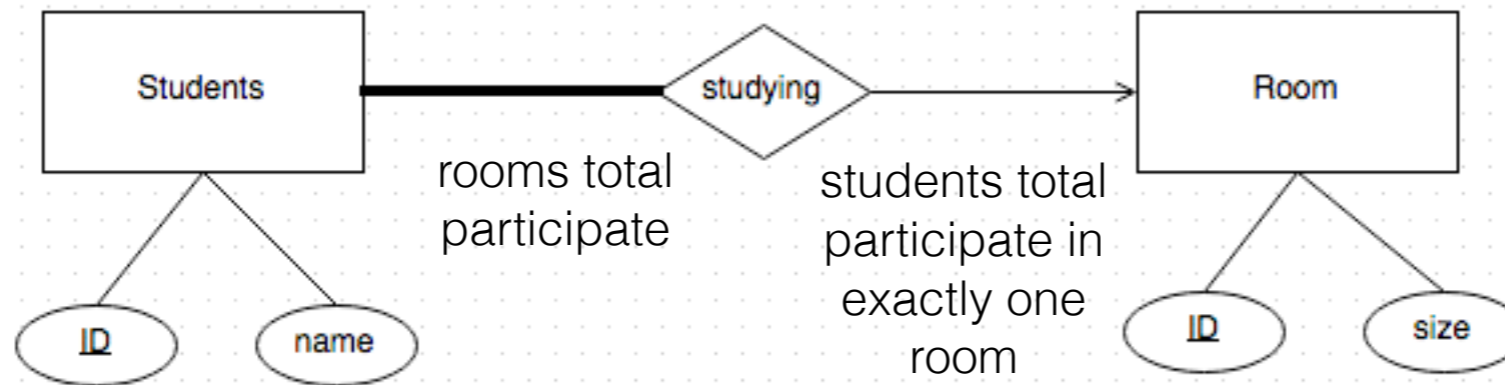
YU HUANG

2015-01-28

Outline

- (many-to-one)at least one

- primary key, candidate key, super key

- foreign key

- is-a relationship

- review DB2 command;

at least one(many-to-one)

"all students are studying in some room, but one student is studying exactly one room at one time, and every room has at least one student".

It's a many-to-one relationship. All students are studying in some room, and one student is studying exactly one room at one time, means it's total participation and exactly one relationship; "every room has at least one student" means no room is empty, that is, all room are participated, therefore, it's total participation.

student room

- left side(students) total participate in exactly one room(many-to-exactly one)

- right side(room) at least has one student, means all room are total participate
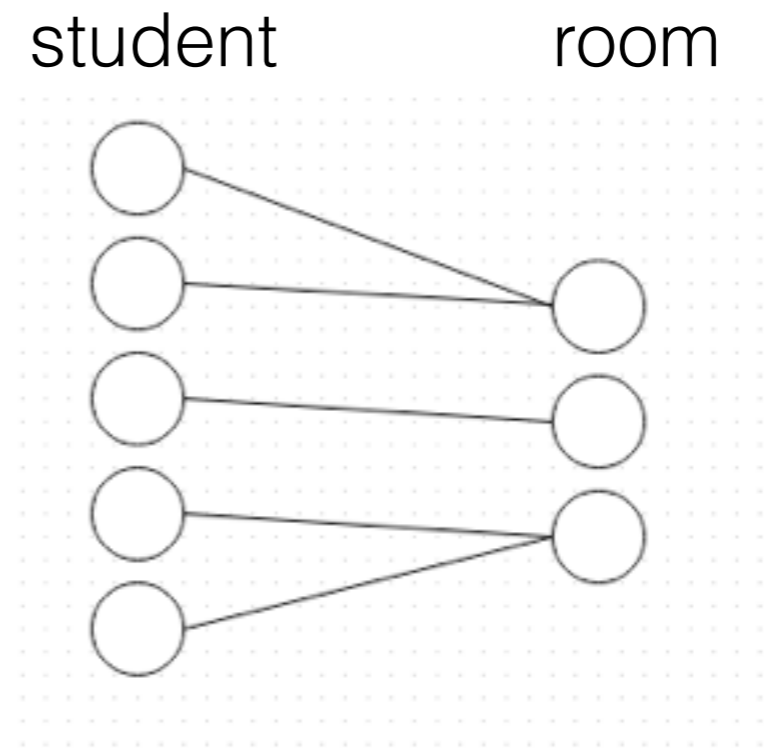
```
connect to cs3db3;

create table student(
    ID integer not null,
    name varchar(30),
    primary key (ID)
);


create table room(
    ID integer not null,
    size varchar(10),
    primary key (ID)
);


create table studying(
    studentID integer not null,
    roomID integer not null,
    primary key (studentID, roomID),
    foreign key (studentID) references student (ID) on delete cascade,
    foreign key (roomID) references room (ID) on delete cascade
);

list tables;
```

Super key, Candidate key, Primary key

- Super Key—is defined as a set of attributes within a table that uniquely identifies each record within a table.

- Candidate Key – A Candidate Key can be any column or a combination of columns that can qualify as unique key in database. There can be multiple Candidate Keys in one table. Each Candidate Key can qualify as Primary Key.

- Primary Key – A Primary Key is a column or a combination of columns that uniquely identify a record. Only one Candidate Key can be Primary Key.
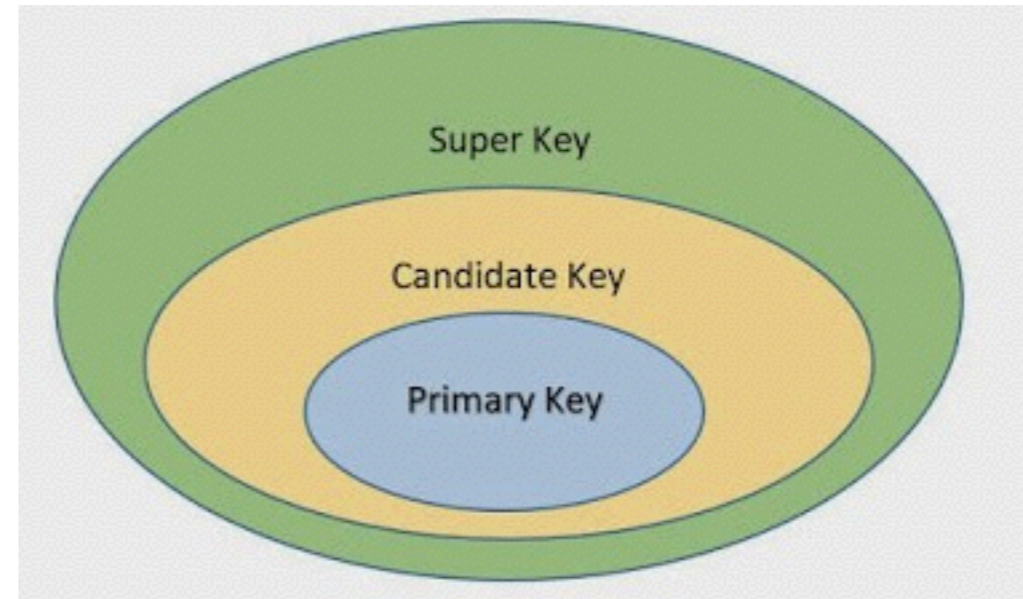
| StudentID	Name	Major	Email
1001	Jone	CS	aaa@mcmaster.ca
1002	Jack	CS	bbb@mcmaster.ca
1003	White	EE	ccc@mcmaster.ca
1004	Jone	SE	ddd@mcmaster.ca

Candidate key: StudenetID, Email

Primary key: StudentID

ISA relationship

- Student is a Person, Teacher is a Person

- Student and Teacher both inherit the Person's attributes(ID, Name, Email)

- Student has one additional attribute "Major"

- Teacher has one additional attribute "Office"
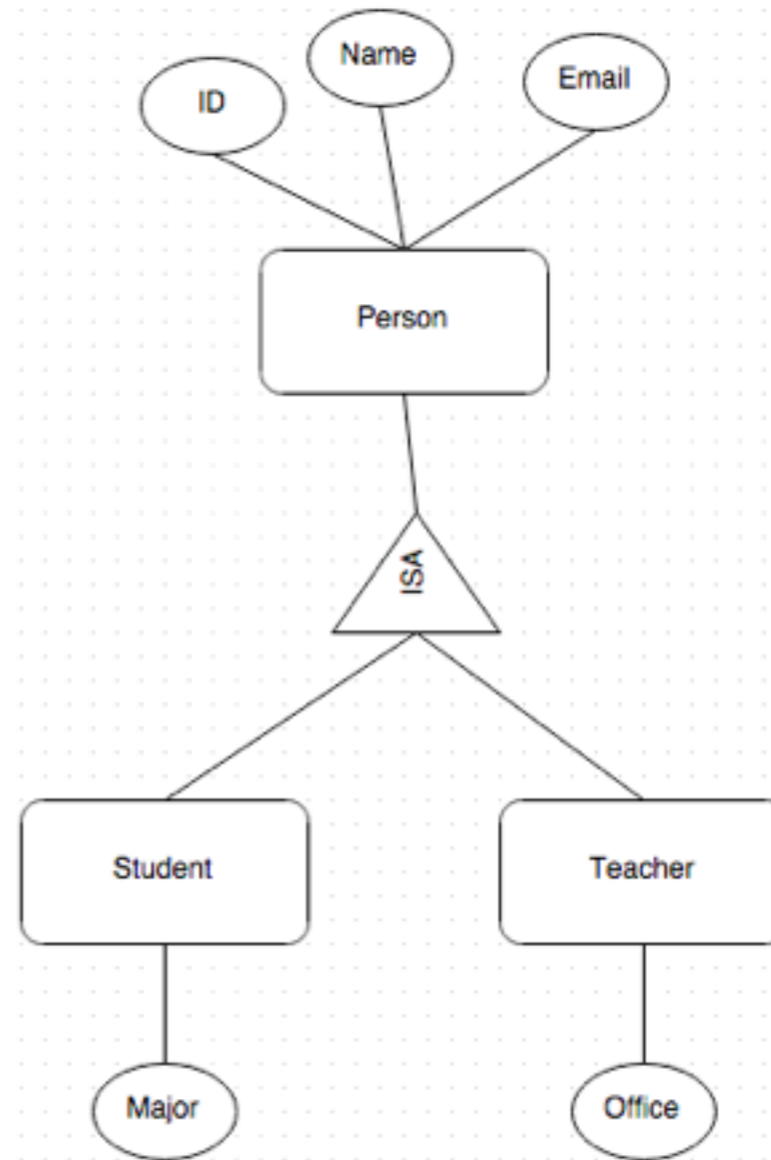
ISA relationship

- Don't need to create table for ISA relationship

- Create the Person table first and then create student and Teacher Table

- The primary key ID in student table is the same as the ID in Person table

```sql
connect to cs3db3;

create table Person (
    ID integer not null,
    Name varchar(20) not null,
    Email varchar(40) not null,
    primary key (ID)
);

create table Student(
    ID integer not null references Person (ID) on delete cascade,
    Major varchar(30) not null,
    primary key (ID)

);

create table Teacher(
    ID integer not null references Person(ID) on delete cascade,
    Office varchar (30) not null,
    primary key (ID)
);


list tables;
```

ER-diagram design Tips

- find out all entities and circle them out

- figure out the relationship between these entities according to the description

- read description again, make sure you don't miss anything out

- Don't over read the description, sentences may be redundancy, but design should be simple

Access to DB2

- Windons

  download the SSH software (putty or secureCRT)

  http://www.cas.mcmaster.ca/~fchiang/courses/db3/W15/help/ssh.html

- Mac, linux etc

open Terminal, input ssh

Access to DB2

- two servers you can use
  . CS3DB3 ssh macid@db2srv2.mcmaster.ca

  . SE4DB3 ssh macid@db2srv3.mcmaster.ca

Tips:

1. please connect to the MacSecure WiFi, not MacConnect

2. Off-campus, please ssh to Mills first, then ssh to server from Mills (ssh macid@mills.cas.mcmaster.ca)

3. You don't need VPN !!

upload script

- Windows (use the software, very easy)

- Mac, Linux(use scp command)

scp file's source path MACID@mills.mcmaster.ca: destination path

example:

 scp /Users/yu/script.txt MacID@mills.mcmaster.ca: / u20/somepath/

Basic command

- - - comment

- first line in your script

(login db2srv2.mcmaster.ca)

CONNECT TO CS3DB3;

- -your own code;

list tables;

Please don't include terminate;

Run Script

- run your script in server

  db2 -tnf script.ddl

create table

CREATE TABLE Course(

    cid CHAR(2) NOT NULL,

    name CHAR(10),

    PRIMARY KEY (cid)

);

IMPORTANT:

1. Make sure you create all entity tables before

you create any relationship tables !!!

2. Create tables that without foreign key before tables with foreign key!!!

view table

- view all tables

  list tables

- view one specified table in detail

  describe table tablename

Important tips

1. Make sure you create all entity tables **before**

you create any relationship tables !!!

2. Create tables that without foreign key **before** tables with foreign key!!!

3. when you write your ddl,

- - first two lines;

  connect to cs3db3;

      - -last line

    list tables;

    - - please do **NOT** include Terminate;

example code

```sql
1   -- connect to database;
2   connect to cs3db3;
3   --+++++++++++++++++++++++++++++++++++++++++++++++++++
4   -- CREATE TABLES student, taking, course
5   --+++++++++++++++++++++++++++++++++++++++++++++++++++
6
7   create table Student(
8       ID integer not null,
9       name varchar(10),
10      primary key (ID)
11  );
12
13  create table Course(
14      ID integer not null,
15      name varchar(10),
16      primary key (ID)
17  );
18
19  create table Taking(
20      studentID integer not null,
21      courseID integer not null,
22      primary key (studentID, courseID),
23      foreign key (studentID) references  Student (ID) on delete cascade,
24      foreign key (courseID) references  Course (ID) on delete cascade
25  );
26
27  list tables;
28
```

example code Drop table

```
-- connect to database;
connect to cs3db3;
--++++++++++++++++++++++++++++++++++++++++++++++++++++++
-- DROP TABLES student, taking, course
--++++++++++++++++++++++++++++++++++++++++++++++++++++++
DROP table Student;
DROP table Course;
DROP table Taking;
```