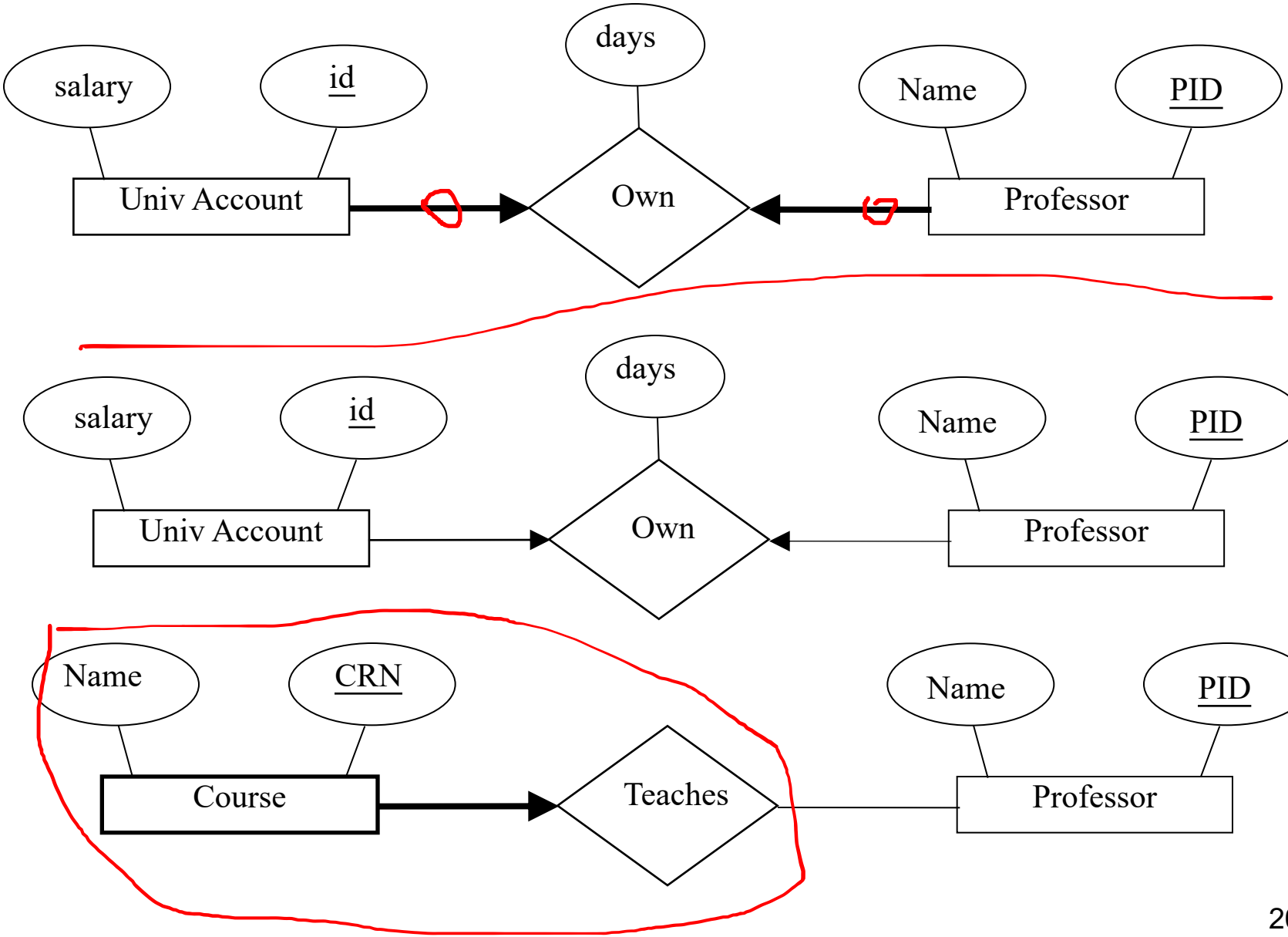
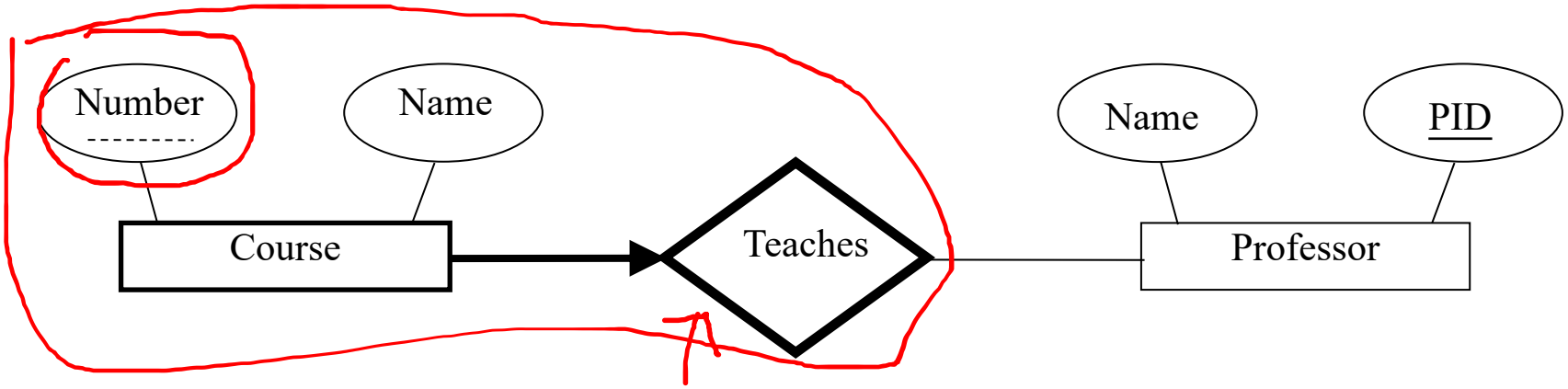
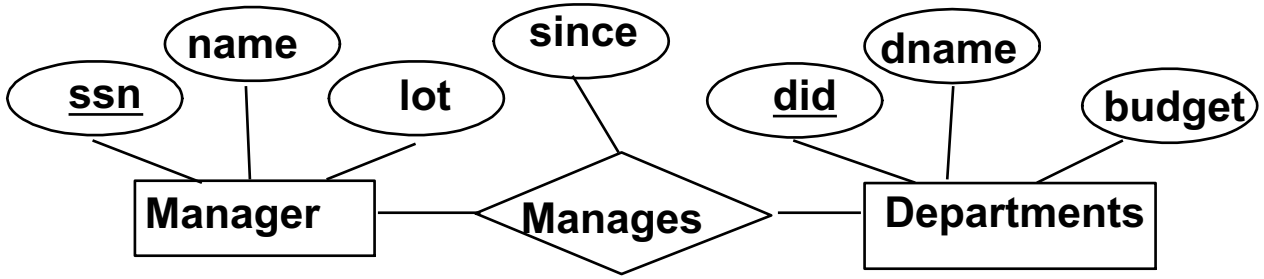
CSCI4333 Database Design & Implement

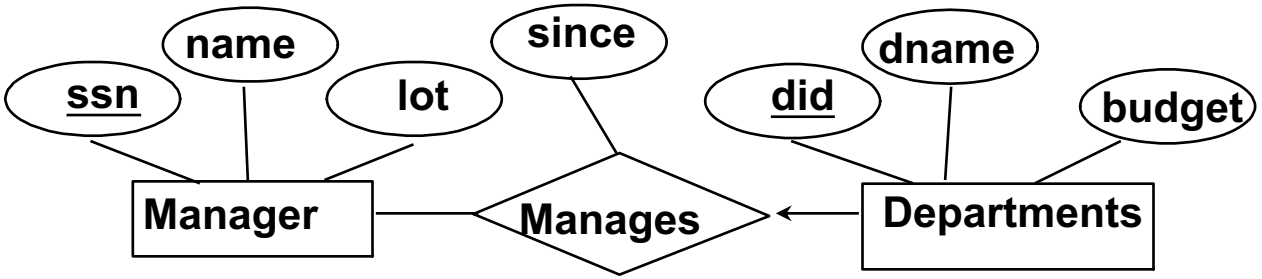
Lecture Ten – Relational Model 4

Instructor: Dr. Yifeng Gao


days
salary
id
Name
PID
Univ Account
Own
Professor
days
salary
id
Name
PID
Univ Account
Own
Professor
Name
CRN
Name
PID
Course
Teaches
Professor

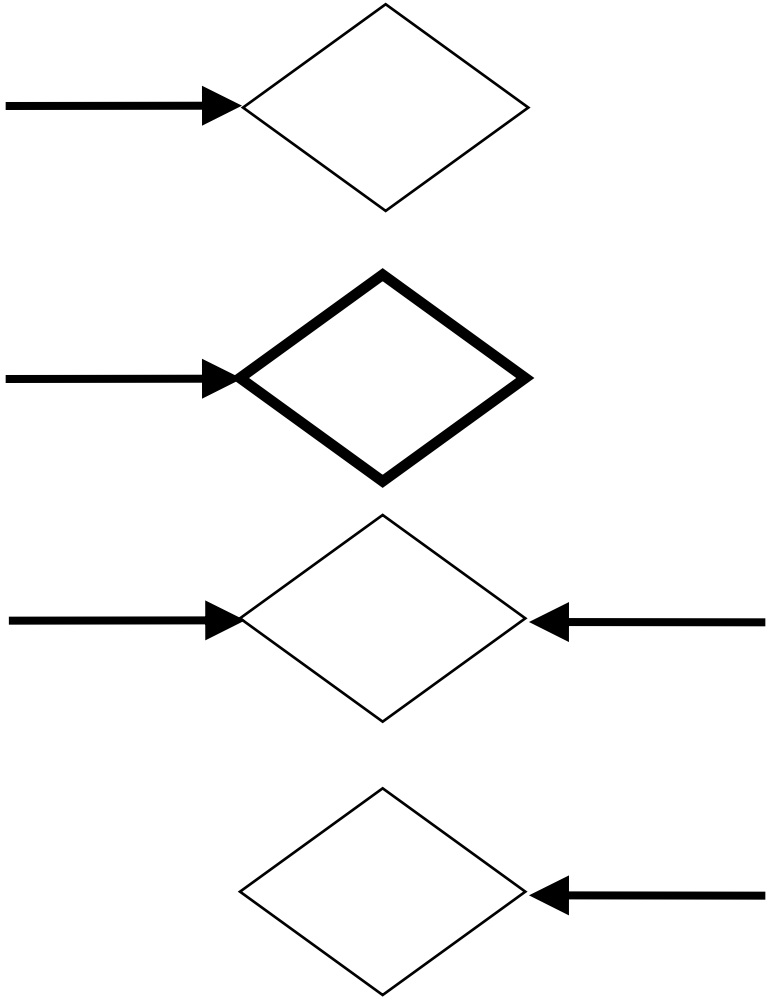

Number
Name
Course
Teaches
Name
PID
Professor


name
since
ssn
lot
dname
did
budget
Manager
Manages
Departments


name
since
ssn
lot
dname
did
budget
Manager
Manages
Departments

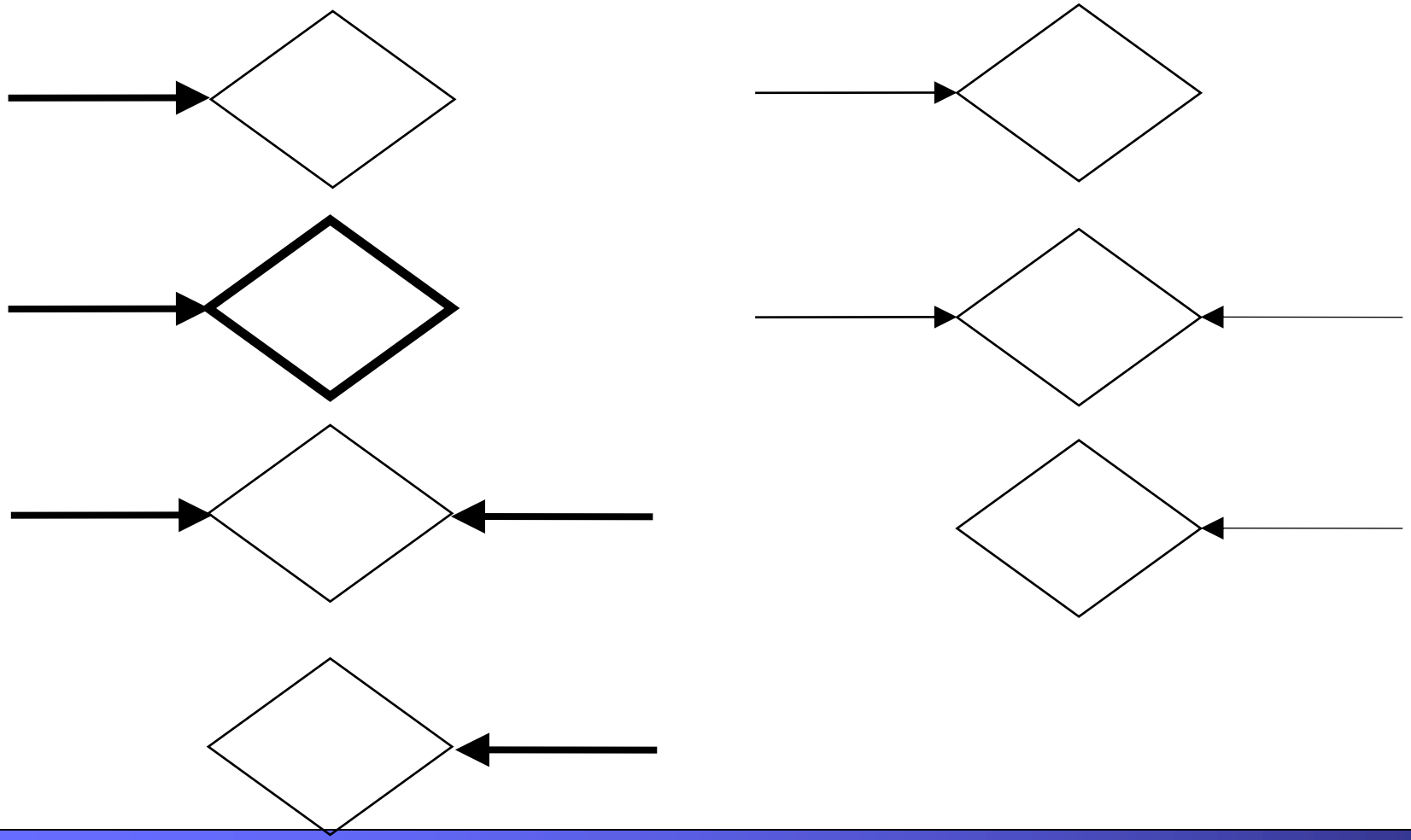
Cheatsheet

- Must Merge:

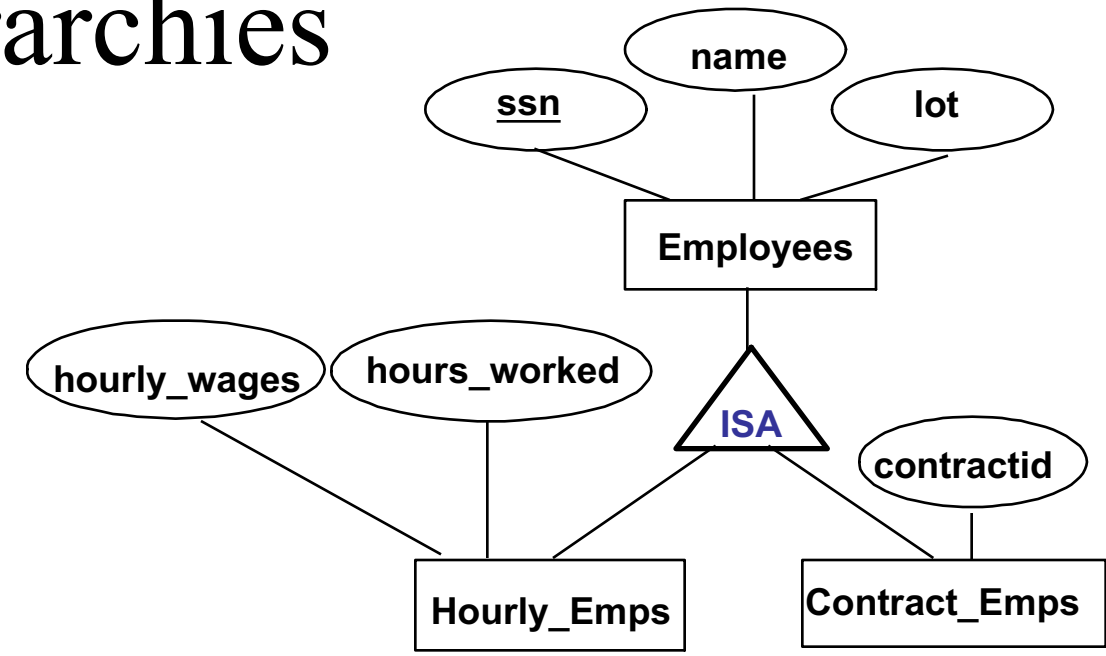


Cheatsheet

- Single Primary Key: Any ER has an arrow



ISA (‘is a’) Hierarchies



General approach:

3 relations: Employees, Hourly_Emps and Contract_Emps.

- *Employee(ssn,name,lot)*
- *Hourly_Emps*(*hourly_wages*, *hours_worked*, *ssn)*;
  - must delete Hourly_Emps tuple if referenced Employees tuple is deleted.
- Contract_Emps: Contract_Emps (contract_id,ssn);
  - must delete Contract_Emps tuple if referenced Employees tuple is deleted.

isA

- CREATE TABLE Employee(
  ssn CHAR(10),
  name CHAR(10),
  lot INTEGER,
  PRIMARY KEY (ssn)
  );

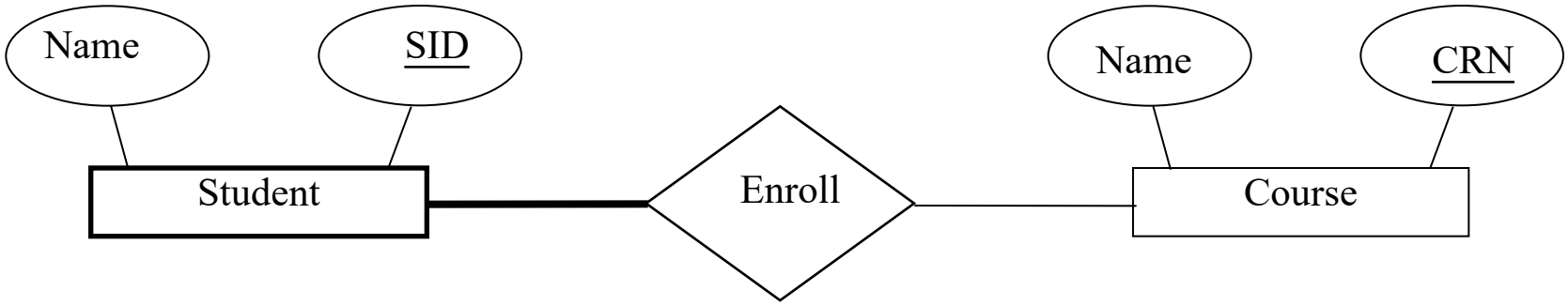
isA

- CREATE TABLE Hourly_Emps(
  ssn CHAR(10),
  hourly_wages INTEGER,
  hours_worked INTEGER,
  PRIMARY KEY (ssn)
  FOREIGN KEY (ssn) REFERENCE Employee
  ON DELETE CASCADE
  );

isA

- CREATE TABLE Contract_Emps(
  ssn CHAR(10),
  hourly_contractid CHAR(10),
  PRIMARY KEY (ssn)
  FOREIGN KEY (ssn) REFERENCE Employee
  ON DELETE CASCADE
  );

How to address this case?



Only solution so far is just combined two table….

But it indeed increase the space cost!

How to address this case?

- CREATE TABLE Student (

  Name CHAR(10),

  SID CHAR(10),

  PRIMARY KEY (SID)

  FOREIGN KEY (SID) REFERENCE Enroll);

Relational Model: Summary

- A tabular representation of data.
- Simple and intuitive, currently the most widely used.
- Integrity constraints can be specified by the DBA, based on application semantics. DBMS checks for violations.
  - Two important ICs: primary and foreign keys
  - In addition, we *always* have domain constraints.
- Powerful and natural query languages exist.
- Rules to translate ER to relational model

One More Thing!

```
CREATE TABLE Enrolled
  (sid CHAR(20),
   cid  CHAR(20),
   grade CHAR(2),
   PRIMARY KEY  (sid,cid) );
```

Oracle - Industrial level. Paid Software

```
CREATE TABLE artist(
  artistid    INTEGER PRIMARY KEY,
  artistname  TEXT
);
```

SQLite - lightweight SQL, naturally come with python

```
create table tutorials_tbl(
   tutorial_id INT NOT NULL AUTO_INCREMENT,
   tutorial_title VARCHAR(100) NOT NULL,
   tutorial_author VARCHAR(40) NOT NULL,
   submission_date DATE,
   PRIMARY KEY ( tutorial_id )
);
```

MySQL - opensource version SQL