# Session 3: ER to Relational Mapping
## *CSCI-585, Cyrus Shahabi*

- Strong entity set with attributes $a_1$, $a_2$, …, $a_n$: represent it as a table with *n* unique columns (one column per attribute). Example: ….

  Each row in this table corresponds to one entity of the entity set. We may add/delete/modify rows in the table.

- Weak entity set with attributes $a_1$, $a_2$, …, $a_n$ and an owner entity set with primary key $b_1$, $b_2$, …, $b_m$ : represent it as a table with n+m columns, one for each of { $a_1$, $a_2$, …, $a_n$} U { $b_1$, $b_2$, …, $b_m$ }.  $b_1$, $b_2$, …, $b_m$ is the foreign key of the resulting relation referring to the corresponding relation of the owner entity set. Example:

- *(Idea: keep rows unique.)*
- N-ary relationship set R with attributes $a_1$, $a_2$, …, $a_n$ among entity sets $E_i$ 's (say m entity sets): represent it as a table with n+m columns, one for each of { $a_1$, $a_2$, …, $a_n$} U {prim-key($E_1$), prim-key($E_2$), …, prim-key($E_m$)}.
- Binary relationship set R with attributes $a_1$, $a_2$, …, $a_n$ among entity sets corresponding to relations S and T:
  - If 1:1 then choose either relations (say S) and extend it with prim-key(T) U { $a_1$, $a_2$, …, $a_n$}

  - If 1:N or N:1 then choose the N-side relation (say S) and extend it with prim-key(T) U { $a_1$, $a_2$, …, $a_n$}

  - If N:M then create a new relation as:
        prim-key(S) U prim-key(T) U { $a_1$, $a_2$, …, $a_n$}

- For multivalued attribute A of entity set S, create a new relation as:  A U prim-key(S)

How to enforce Referential Integrity?
- ➢ Consider Students and Enrolled; *sid* in Enrolled is a foreign key that references Students.
- ➢ What should be done if an Enrolled tuple with a non-existent student id is inserted? (Reject *it!*)
- ➢ What should be done if a Students tuple is deleted?
  - – Also delete all Enrolled tuples that refer to it.
  - – Disallow deletion of a Students tuple that is referred to.
  - – Set sid in Enrolled tuples that refer to it to a *default sid* .
  - – (In SQL, also: Set sid in Enrolled tuples that refer to it to a special value *null,* denoting `*unknown'* or `*inapplicable'* .)
- ➢ Similar if primary key of Students tuple is updated.

SQL/ 92 supports all 4 options on deletes and updates.
– Default is NO ACTION
( *delete/ update is rejected* )
– CASCADE (also delete all tuples that refer to deleted tuple)
– SET NULL / SET DEFAULT (sets foreign key value of referencing tuple)

*CREATE TABLE Enrolled*
*(sid CHAR (20),*
*cid CHAR( 20) ,*
*grade CHAR (2),*
*PRIMARY KEY (sid, cid),*
*FOREIGN KEY (sid)*
*REFERENCES Students*
*ON DELETE CASCADE*
*ON UPDATE SET DEFAULT )*