

DOMAIN MODEL TO RELATIONAL SCHEMA REDUCTION: OVERVIEW

We need to reduce:

generalizations/specializations

attributes → multivalued

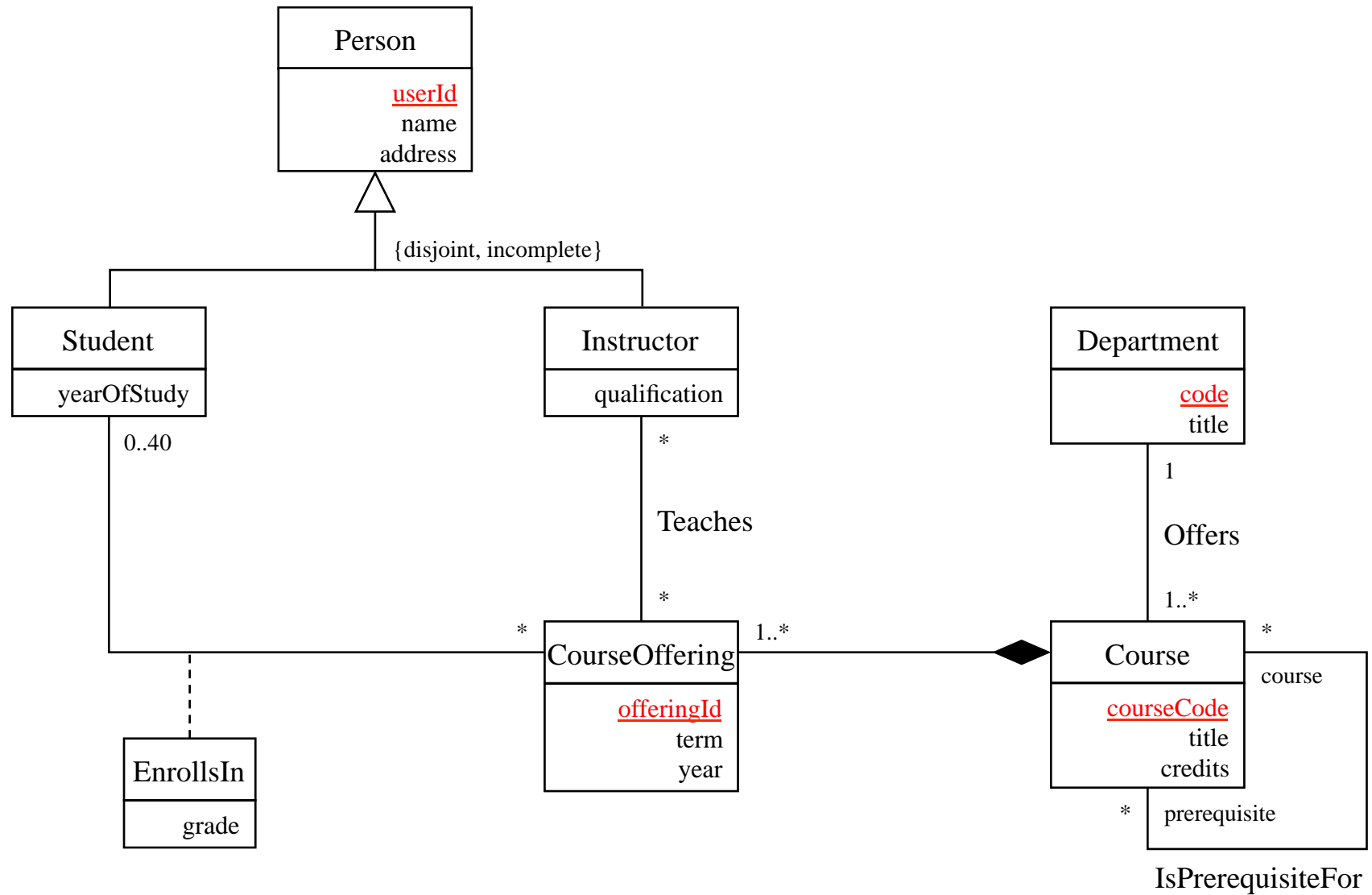
classes → independent, dependent

associations → degree (e.g., unary, binary, ternary, etc.)

→ cardinality constraints

**Cardinality constraints in the UML
are reduced to
referential integrity constraints in the relational model.**

EXAMPLE ASU DOMAIN MODEL (SIMPLIFIED)

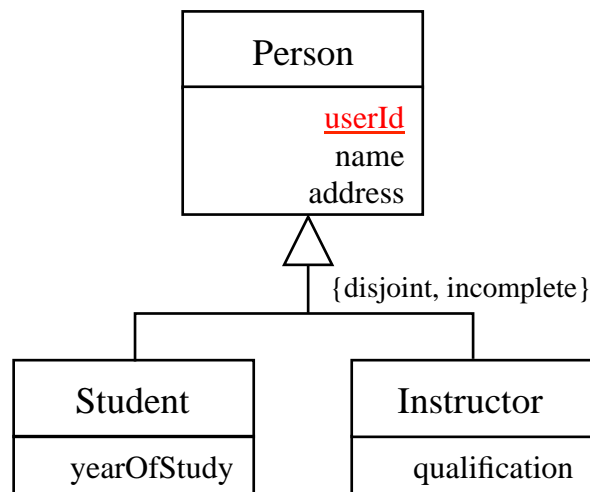


GENERALIZATIONS/SPECIALIZATIONS

Option 1: Reduce all classes to tables.

- Create a table for each class (superclass and subclasses).
- For each table created for a subclass:
 - Add the primary key of the superclass into the table as a foreign key FK_S .
 - The foreign key FK_S becomes the primary key.
 - Add a foreign key constraint: FK_S references superclass table.
 - Add a referential integrity action: on delete cascade.

Example



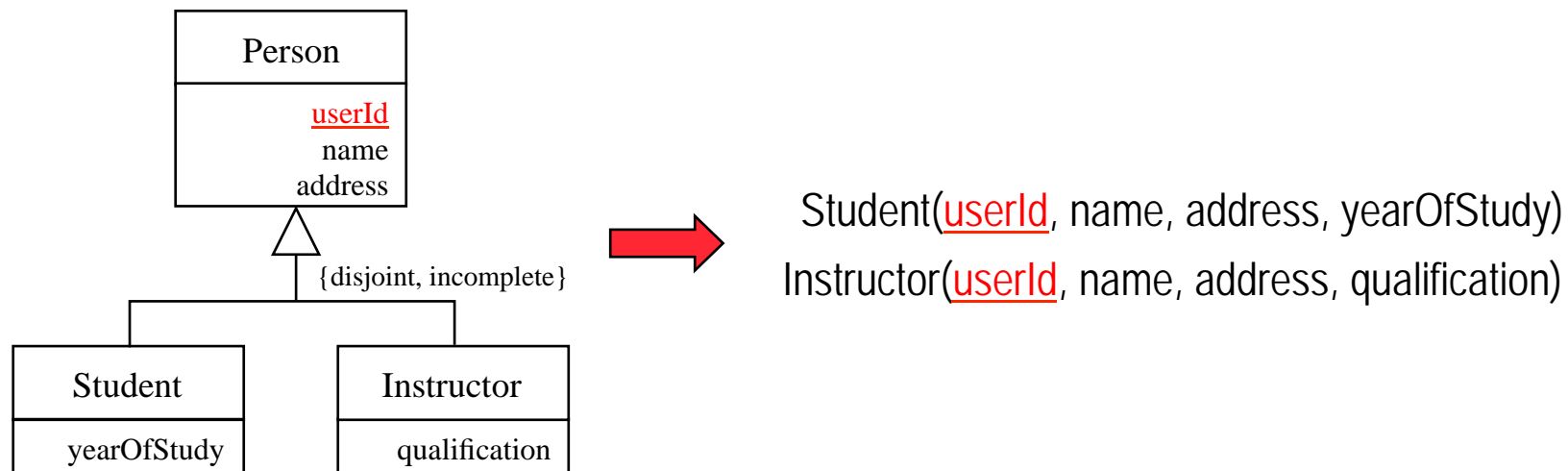
Person(userId, name, address)
Student(userId, yearOfStudy)
 userId references Person
 on delete cascade
Instructor(userId, qualification)
 userId references Person
 on delete cascade

GENERALIZATIONS/SPECIALIZATIONS (cont'd)

Option 2: Reduce only subclasses to tables.

- Create a table for each subclass.
- For each table created for a subclass:
 - Add all the attributes of the superclass.
 - The primary key is the primary key of the superclass.

Example

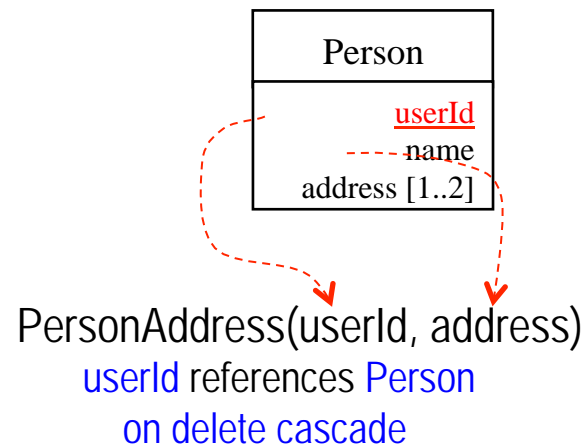


➤ **Should be used only for disjoint generalizations/specializations!**

MULTIVALUED ATTRIBUTES

For a multivalued attribute **M** in a class **S**:

- Create a table **SM** with an attribute **A** that corresponds to **M** and attribute(s), **FK_S**, corresponding to the primary key of class **S**.
- The primary key of table **SM** is the union of all its attributes.
- Add a foreign key constraint: **FK_S** references **S**.
- Add a referential integrity action: **on delete cascade**.

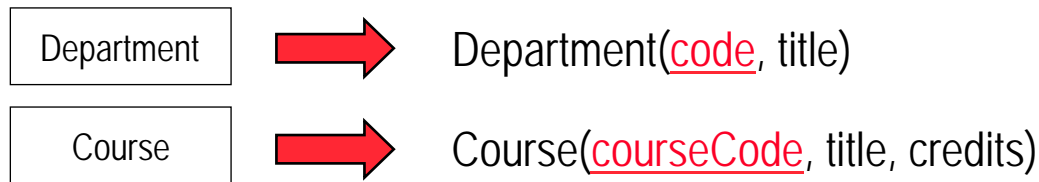


INDEPENDENT CLASSES

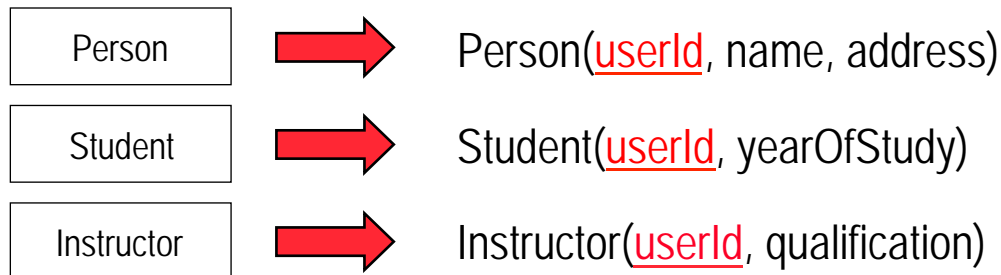
For each independent class **S** that has a primary key:

- Create a table R_S with all the attributes of class **S**.
- The primary key of table R_S is the primary key of class **S**.

Example schema reduction:



From generalization reduction we also have (using option 1):

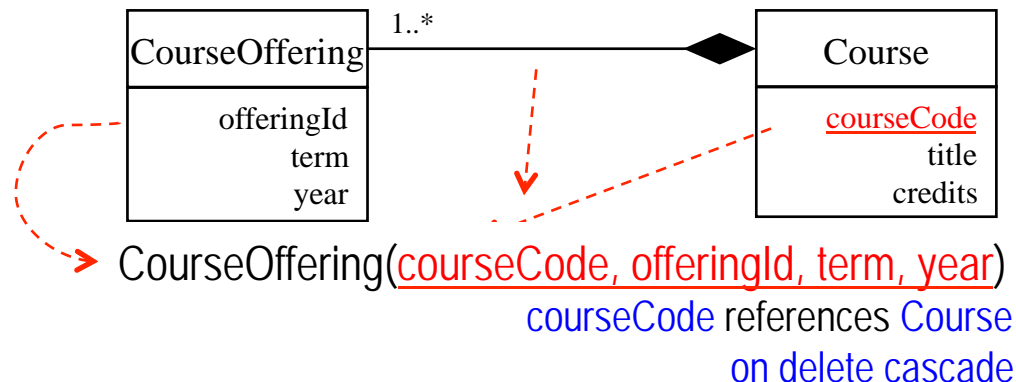


DEPENDENT CLASSES

For a dependent class **T** that depends on another class **S** (**T** has no key):

- Create a table R_T with attributes of the dependent class **T**.
- Include attributes of association **R**, if any, in table R_T .
- Include as foreign key attribute(s) FK_S in table R_T , the primary key attributes of class **S**.
- The primary key of table R_T is the **union of the foreign key attributes FK_S and the discriminator d_a** of the dependent class **T**.
- Add a foreign key constraint: FK_S references **S**.
- Add a referential integrity action: **on delete cascade**.

Example dependent class reduction:



ASSOCIATIONS

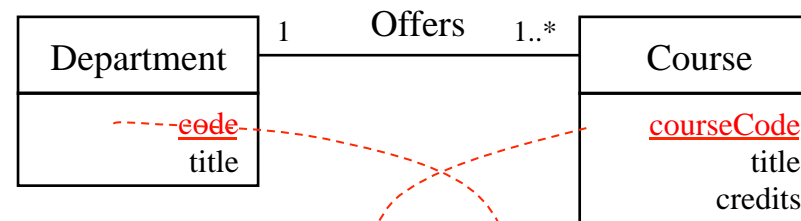
For each binary association **R**:

- Create a new table R_R .
- Include attributes of association **R**, if any, as attributes of table R_R .
- Include as foreign key attributes in table R_R the primary keys of all the classes related by association **R**.
- The primary key of table R_R is
 - *binary 1:1 association* → the primary key of **either class**.
 - *binary 1:N association* → the primary key of the **class on the N-side** of the association.
 - *binary N:M association* → the **union of the primary keys** of the two participating classes.
- Add a foreign key constraint for each foreign key.
- Add a referential integrity action for each foreign key: *on delete cascade*.

👉 **This reduction minimizes null values.**

ASSOCIATIONS: BINARY 1:N

Example binary 1:N association reduction:



Offers(courseCode, code)
courseCode references Course
on delete cascade
code references Department
on delete cascade

ASSOCIATIONS: BINARY N:M

Example binary N:M association reduction:

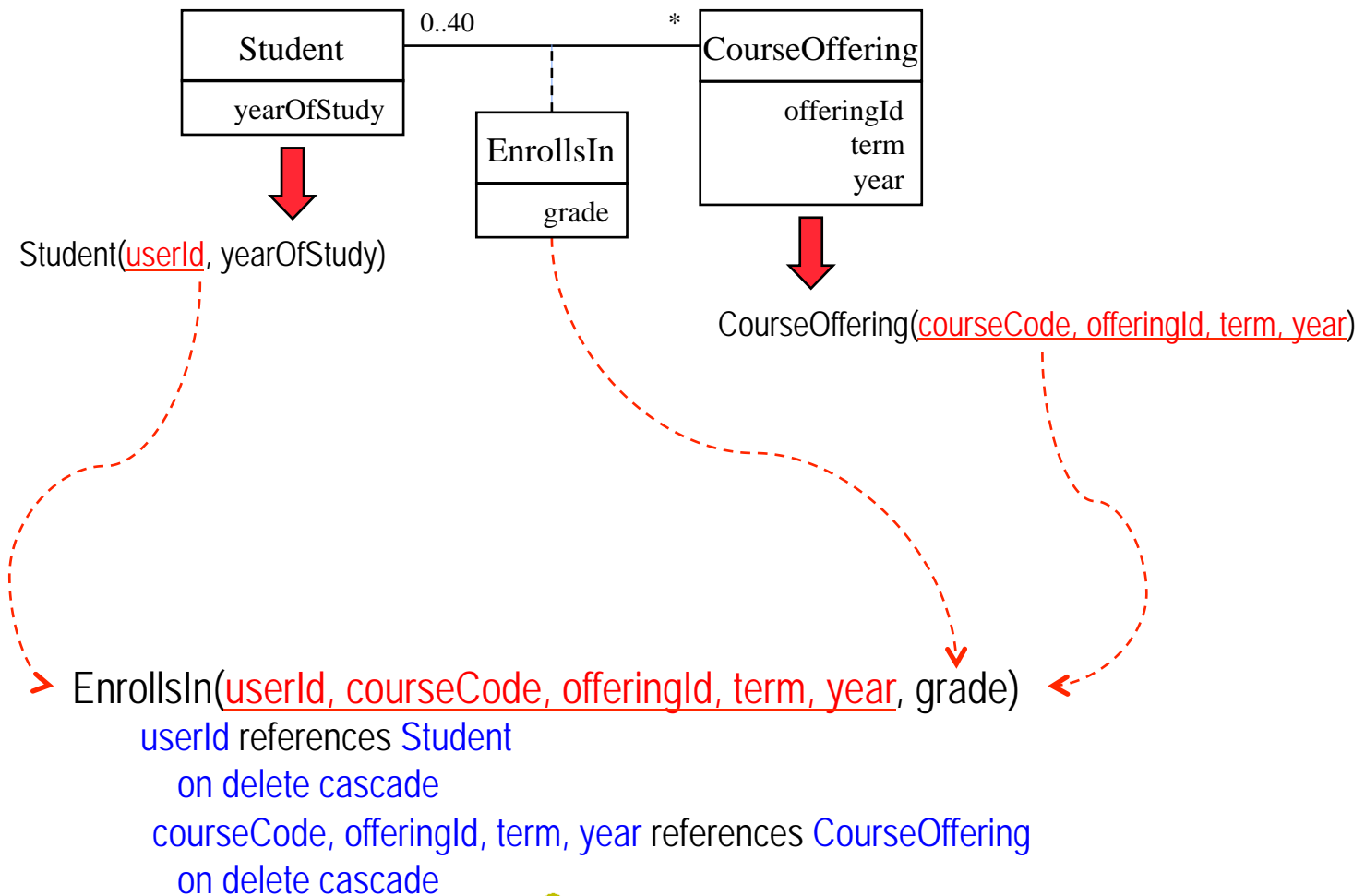
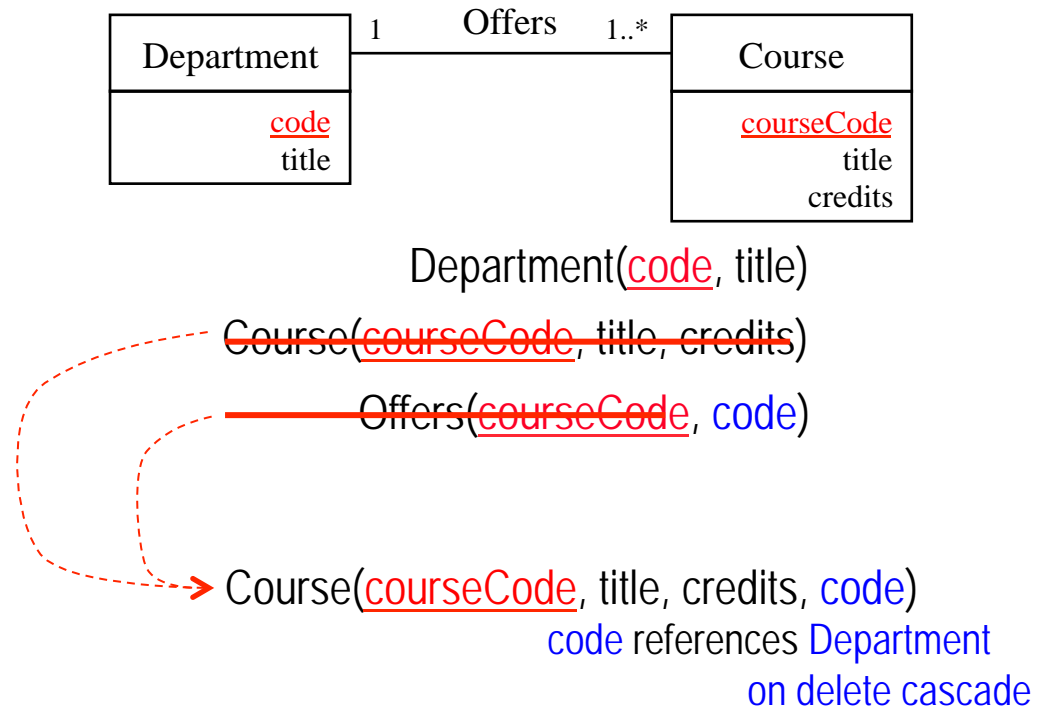


TABLE COMBINATION

- Some tables representing associations can be combined with the table for a related class to reduce the number of tables.
 - For 1:1 associations, the table for the association can be combined with the table for either class.
 - For 1:N associations, the table for the association can be combined with the table for the class on the N-side.
 - Add a foreign key constraint for the foreign key.
 - Add a referential integrity action for the foreign key that depends on the participation constraint of the class on the N-side.

TABLE COMBINATION

Example table combination



EXAMPLE DOMAIN MODEL REDUCTION

Person(userId, name, address)

Student(userId, yearOfStudy)

userId references Person
on delete cascade

Instructor(userId, qualification)

userId references Person
on delete cascade

Department(code, title)

Course(courseCode, title, credits, code)

code references Department
on delete cascade

CourseOffering(courseCode, offeringId, term, year)

courseCode references Course
on delete cascade

EXAMPLE DOMAIN MODEL REDUCTION (cont'd)

EnrollsIn(userId, courseCode, offeringId, term, year, grade)

userId references Student

on delete cascade

courseCode, offeringId, term, year references CourseOffering

on delete cascade

Teaches(userId, courseCode, offeringId, term, year)

userId references Instructor

on delete cascade

courseCode, offeringId, term, year references CourseOffering

on delete cascade

IsPrerequisiteFor(courseCode, prerequisiteCode)

courseCode references Course

on delete cascade

prerequisiteCode references Course

on delete cascade