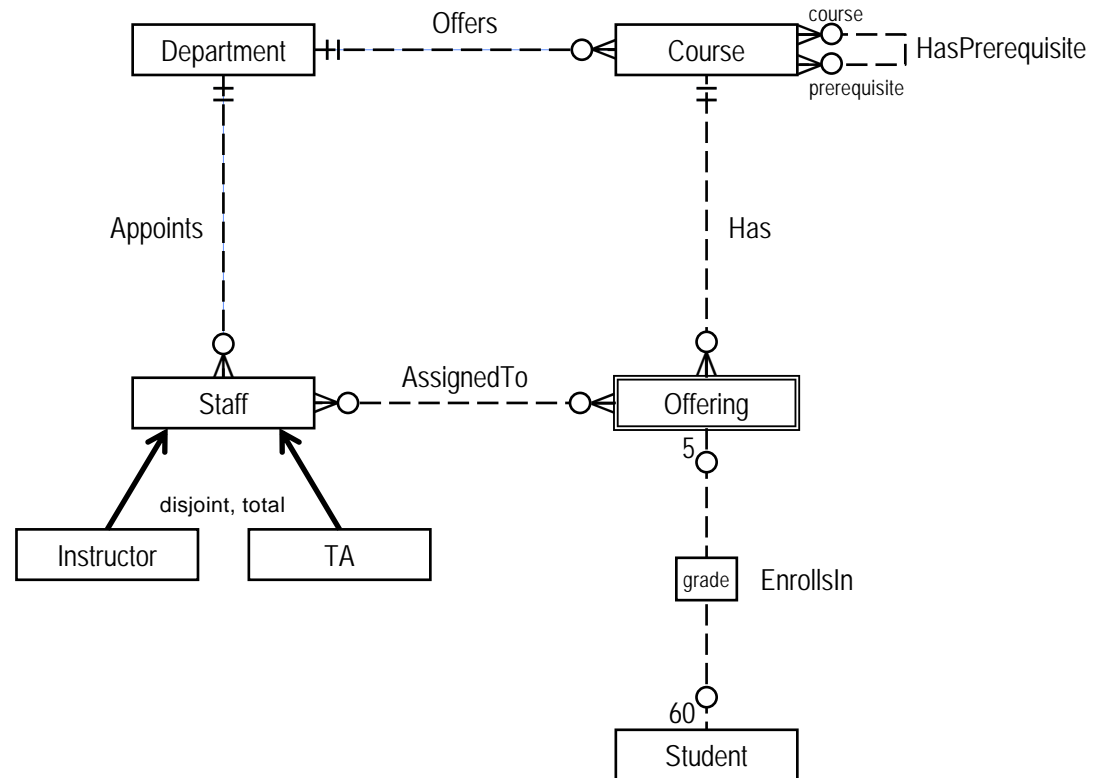


COMP 3311 DATABASE MANAGEMENT SYSTEMS

LECTURE 4 EXERCISES RELATIONAL MODEL AND RELATIONAL DATA BASE DESIGN

EXERCISE 1: REDUCE UNIVERSITY SCHEMA

Reduce the university E-R schema to relation schemas. Use schema combination where possible to reduce relationships. Specify all referential integrity constraints.



Student
<u>studentId</u>
name
{major}

Department
<u>code</u>
name

Course
<u>courseId</u>
name

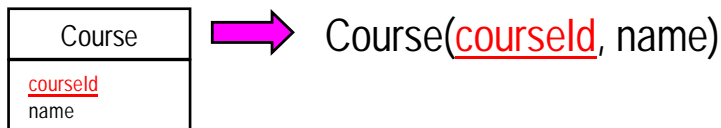
Offering
<u>section</u>
<u>semester</u>
<u>year</u>

Staff
<u>hkid</u>
name
officeNumber

Instructor
title

TA

EXERCISE 1: REDUCE STRONG ENTITIES



How do we reduce the strong entities?

⇒ Create a relation for each strong entity with the same attribute as the entity.

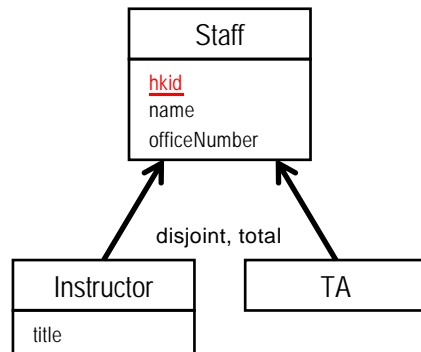
What are the keys of these relations? ⇒ Same as the entities.

What are the foreign key constraints? ⇒ None.

What are the referential integrity actions? ⇒ None.

EXERCISE 1: REDUCE GENERALIZATIONS

Option 1: Reduce *all entities* to relation schemas.



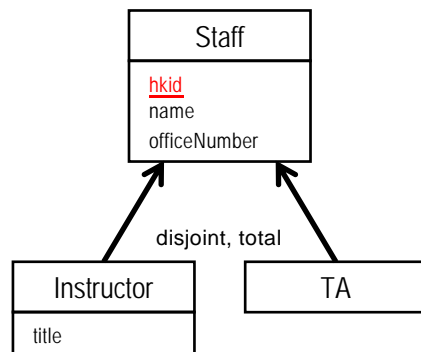
➡ Staff(hkid, name, officeNumber)

Instructor(hkid, title)
foreign key (hkid) references Staff(hkid)
on delete cascade

TA(hkid)
foreign key (hkid) references Staff(hkid)
on delete cascade

Which option to select?

Option 2: Reduce *only subclass entities* to relation schemas.



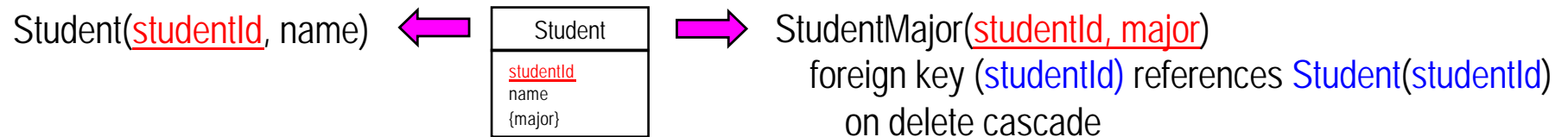
➡ Instructor(hkid, name, officeNumber, title)

TA(hkid, name, officeNumber)

Select Option 1 since Staff has a relationship to other entities and some subclass entities have their own attributes.

EXERCISE 1: REDUCE COMPOSITE/ MULTIVALUED ATTRIBUTES

Multivalued attributes: major



How do we reduce the multivalued attribute major?

⇒ Create a relation StudentMajor and include studentId, the key of Student, and the attribute major.

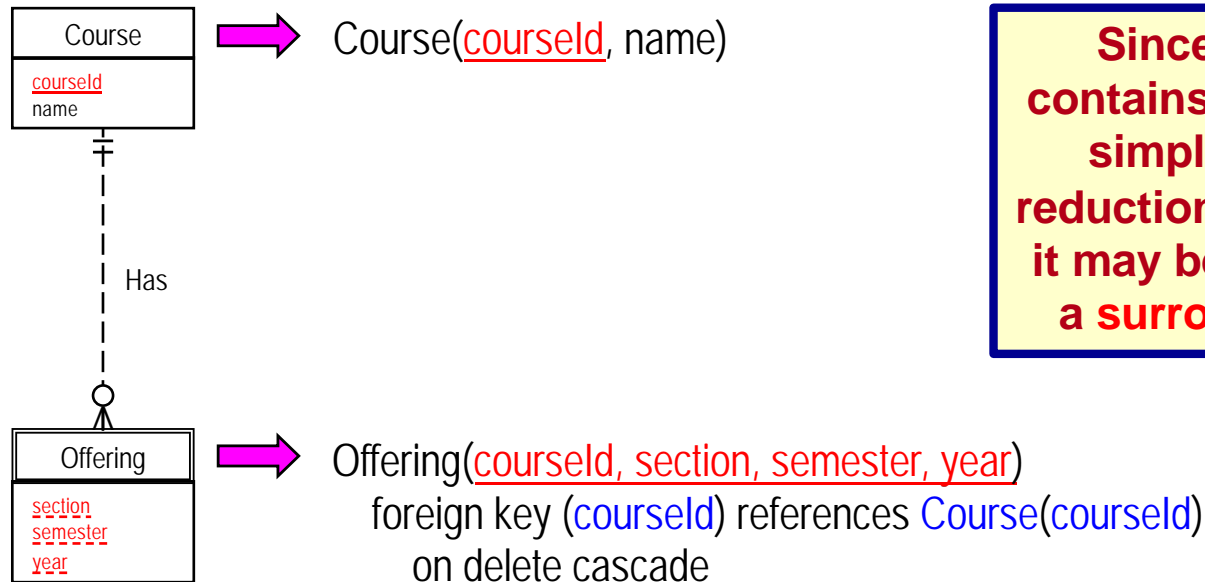
What is the key of this relation?

What is the foreign key constraint?

What is the referential integrity action?

EXERCISE 1: REDUCE WEAK ENTITIES

Offering entity



Since the key of **Offering** contains several attributes, to simplify the subsequent reductions that involve **Offering**, it may be helpful to introduce a surrogate key for **Offering**.

How do we reduce this entity?

⇒ Create a relation from **Offering** and include **courseId**, the key of **Course**, as a foreign key.

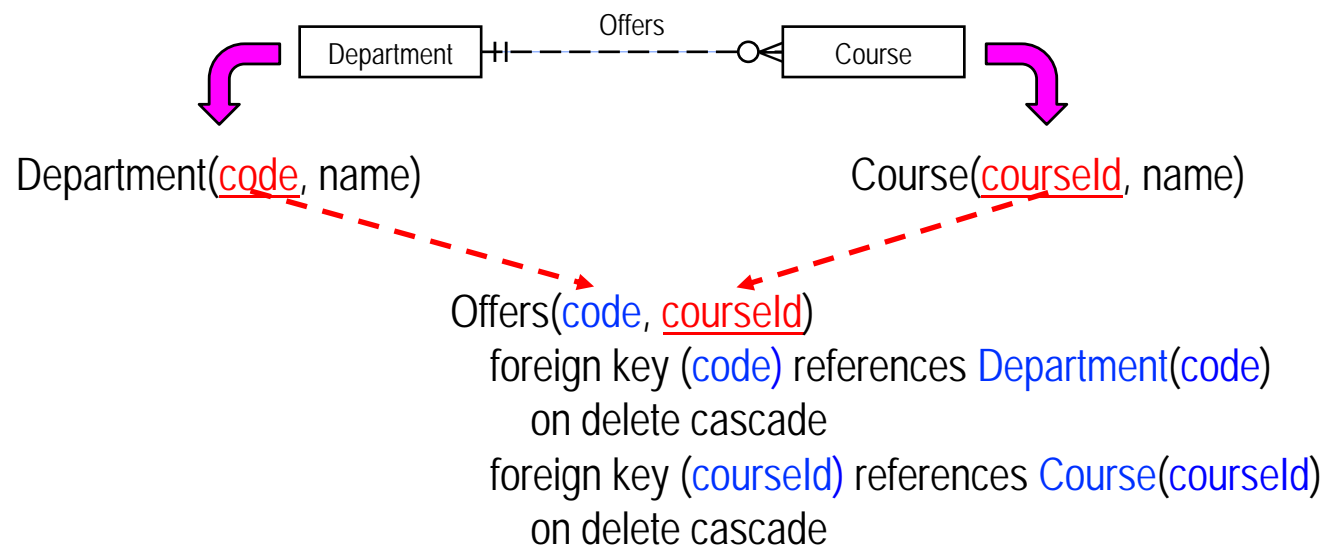
What is the key of this relation?

What is the foreign key constraint?

What is the referential integrity action?

EXERCISE 1: REDUCE 1:N RELATIONSHIPS

Offers relationship between **Department** and **Course**



How do we reduce this relationship?

⇒ Create a relation, **Offers**, containing the keys of **Department** and **Course**.

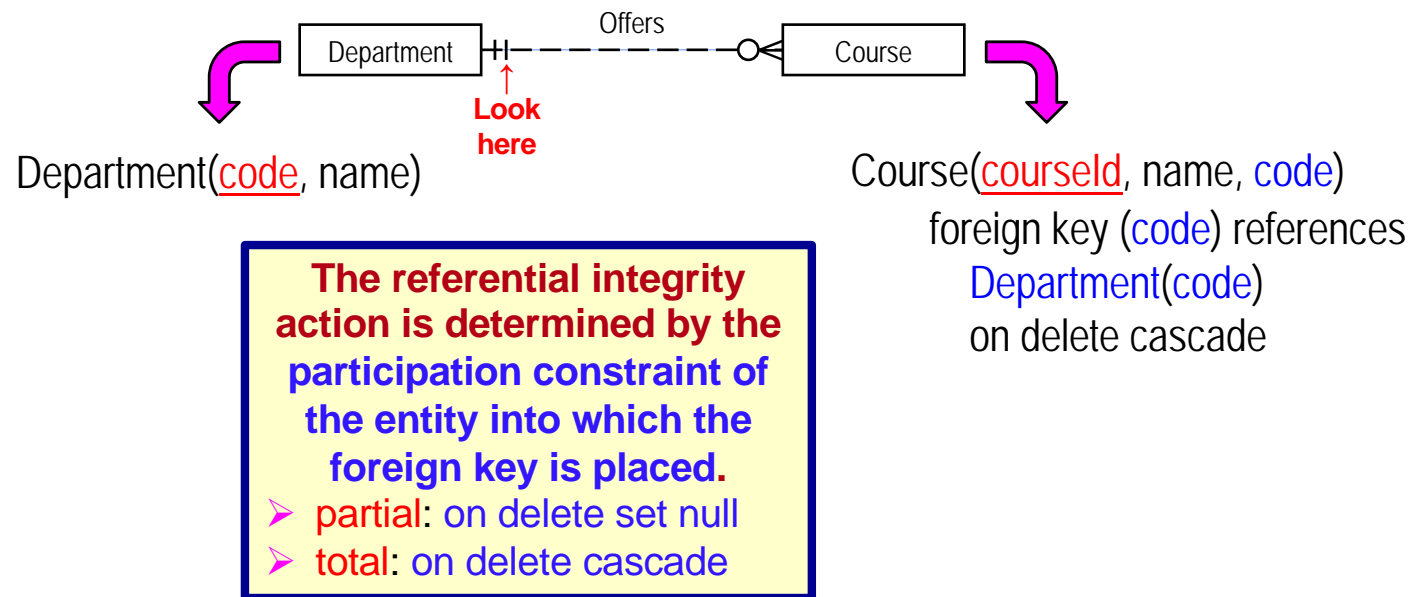
What is the key of the relation?

What are the foreign key constraints?

What are the referential integrity actions?

EXERCISE 1: REDUCE 1:N RELATIONSHIPS

Offers relationship between **Department** and **Course** (using schema combination)



Which relation do we use?

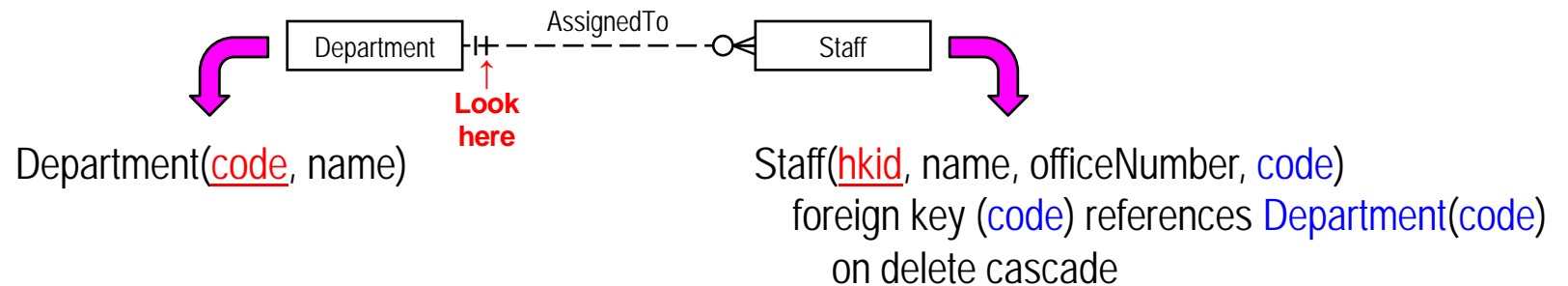
⇒ Course (Add code, the key of Department, as a foreign key.)

What is the foreign key constraint?

What is the referential integrity action?

EXERCISE 1: REDUCE 1:N RELATIONSHIPS

Appoints relationship between **Department** and **Staff** (using schema combination)



Which relation do we use?

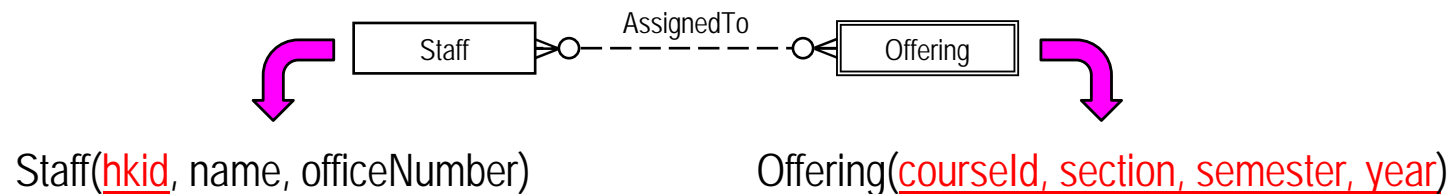
⇒ Staff (Add code, the key of Department, as a foreign key.)

What is the foreign key constraint?

What is the referential integrity action?

EXERCISE 1: REDUCE N:M RELATIONSHIPS

AssignedTo relationship between Staff and Offering



AssignedTo(hkid, courseId, section, semester, year)

foreign key (hkid) references Staff
on delete cascade

foreign key (courseId, section, semester, year) references Offering
on delete cascade

How do we reduce this relationship?

⇒ Create a relation, AssignedTo, containing the keys of the Staff and Offering relations.

What is the key of the relation?

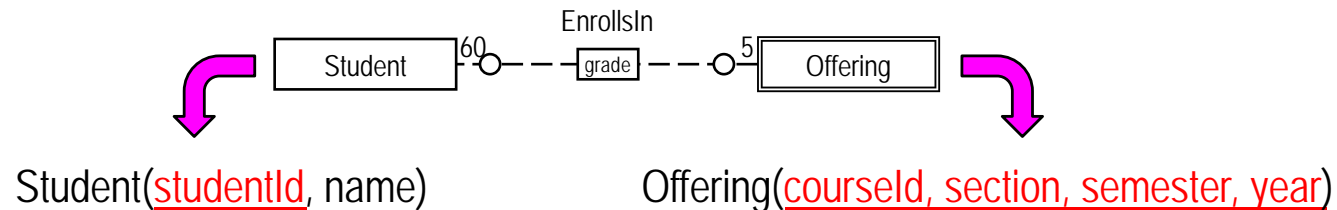
What are the foreign key constraints?

What are the referential integrity actions?

For a relation that represents a relationship, the referential integrity action is always on delete cascade.

EXERCISE 1: REDUCE N:M RELATIONSHIPS

EnrollsIn relationship between Student and Offering



EnrollsIn(studentId, courseId, section, semester, year, grade)
foreign key (studentId) references Student(studentId)
on delete cascade
foreign key (courseId, section, semester, year) references
Offering(courseId, section, semester, year)
on delete cascade

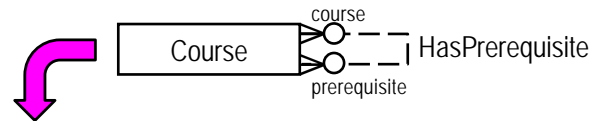
How do we reduce this relationship?

⇒ Create a relation, EnrollsIn, containing the keys of the Student and Offering relations.

Anything else? ⇒ Add the attribute grade to the relation.

EXERCISE 1: REDUCE N:M RELATIONSHIPS

HasPrerequisite relationship between Course and Course



Course(code, name)

HasPrerequisite(courseId, prerequisiteId)

foreign key (courseId) references Course(courseId)
on delete cascade

foreign key (prerequisiteId) references Course(courseId)
on delete cascade

How do we reduce this relationship?

⇒ Create a relation, HasPrerequisite, containing the key of the Course relation (twice).

What is the key of the relation?

EXERCISE 1: UNIVERSITY SCHEMA REDUCTION

Staff(hkid, name, officeNumber, code)

foreign key (code) references Department(code)
on delete cascade

Instructor(hkid, title)

foreign key (hkid) references Staff(hkid)
on delete cascade

TA(hkid)

foreign key (hkid) references Staff(hkid)
on delete cascade

Student(studentId, name)

Course(courseld, name, code)

foreign key (code) references Department(code)
on delete cascade

Department(code, name)

StudentMajor(studentId, major)

foreign key (studentId) references Student(studentId)
on delete cascade

Offering(courseld, section, semester, year)

foreign key (courseld) references Course(courseld)
on delete cascade

AssignedTo(hkid, courseld, section, semester, year)

foreign key (hkid) references Staff(hkid)
on delete cascade

foreign key (courseld, section, semester, year) references
Offering(courseld, section, semester, year)
on delete cascade

EnrollsIn(studentId, courseld, section, semester, year, grade)

foreign key (studentId) references Student(studentId)
on delete cascade

foreign key (courseld, section, semester, year) references
Offering(courseld, section, semester, year)
on delete cascade

HasPrerequisite(courseld, prerequisiteld)

foreign key (courseld) references Course(courseld)
on delete cascade

foreign key (prerequisiteld) references Course(courseld)
on delete cascade

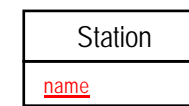
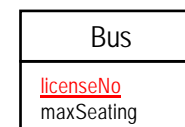
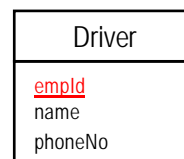
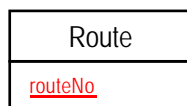
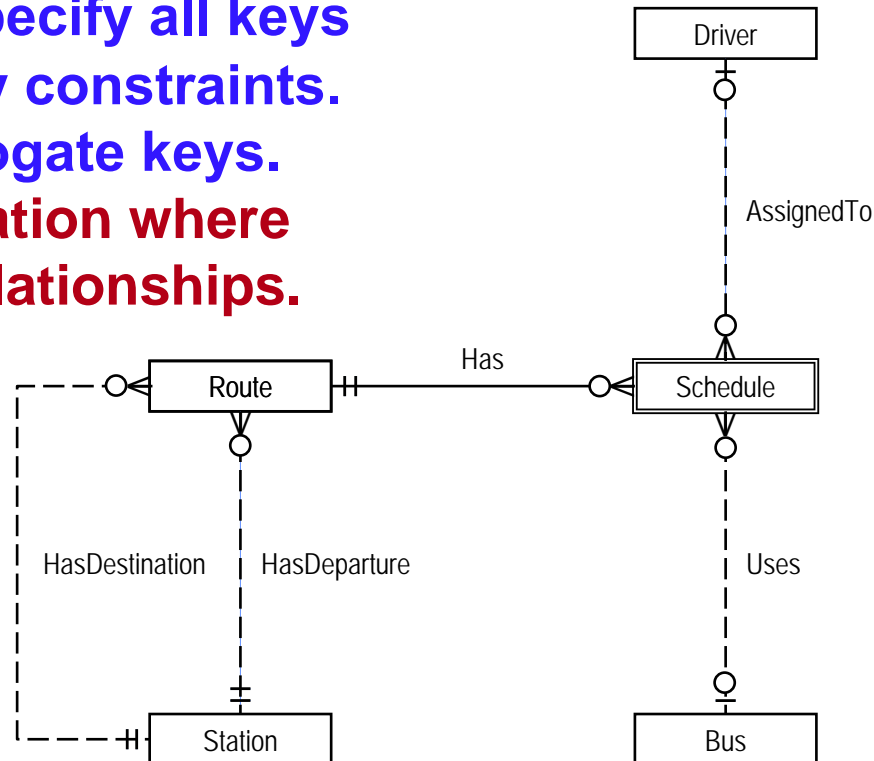


EXERCISE 2: REDUCE BUS COMPANY SCHEMA

Reduce the bus company E-R schema to relation schemas. Specify all keys and referential integrity constraints.

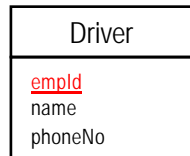
Do not add any surrogate keys.

Use schema combination where possible to reduce relationships.

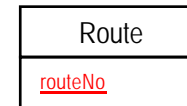


EXERCISE 2: REDUCE ENTITIES

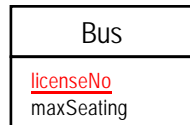
Strong Entities



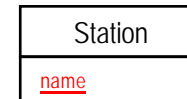
Driver(empld, name, phoneNo)



Route(routeNo)

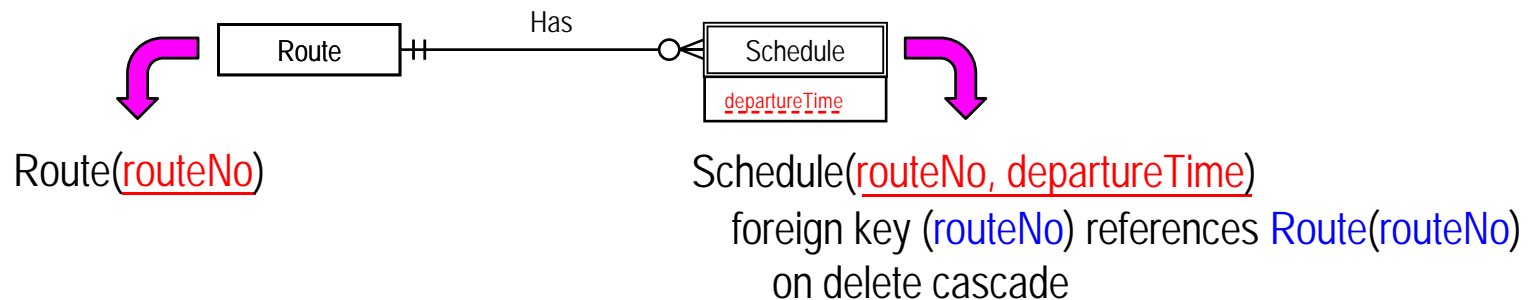


Bus(licenseNo, maxSeating)



Station(name)

Weak Entities



EXERCISE 2: REDUCE 1:N RELATIONSHIPS

AssignedTo relationship between
Driver and **Schedule** (using schema combination)

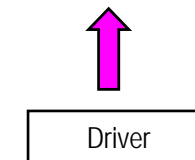
Schedule(routeNo, departureTime, empld)
foreign key (empld) references Driver(empld)
on delete set null

Schedule(routeNo, departureTime)

Uses relationship between
Bus and **Schedule** (using schema combination)

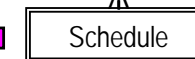
Schedule(routeNo, departureTime, licenseNo)
foreign key (licenseNo) references Bus(licenseNo)
on delete set null

Driver(empld, name, phoneNo)

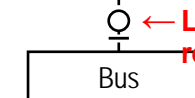


← Look here for the referential integrity action

AssignedTo



Uses



← Look here for the referential integrity action

Bus(licenseNo, maxSeating)

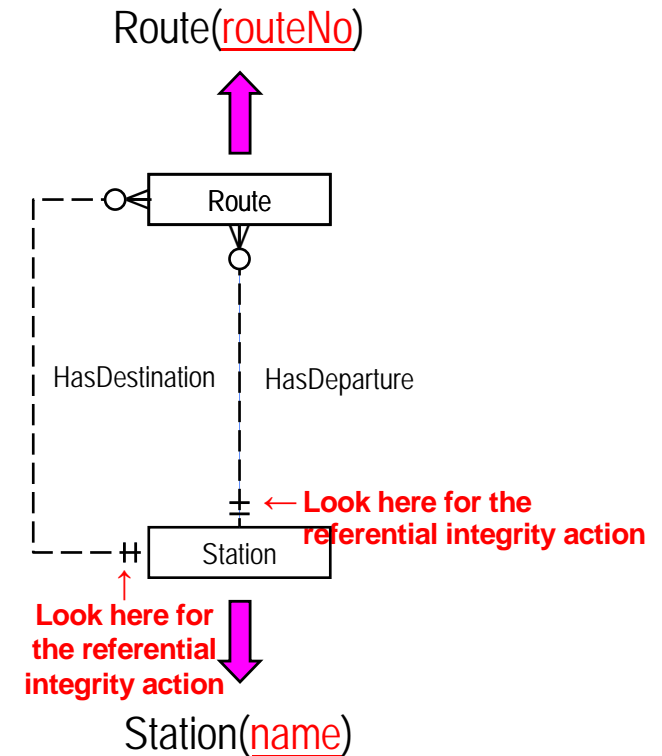
EXERCISE 2: REDUCE 1:N RELATIONSHIPS

HasDeparture relationship between
Route and **Station** (using schema combination)

Route(routeNo, departureStationName)
foreign key (departureStationName) references Station(name)
on delete cascade

HasDestination relationship between
Route and **Station** (using schema combination)

Route(routeNo, departureStationName, destinationStationName)
foreign key (destinationStationName) references Station(name)
on delete cascade



EXERCISE 2: BUS COMPANY SCHEMA REDUCTION

Driver(empld, name, phoneNo)

Bus(licenseNo, maxSeating)

Route(routeNo, departureStationName, destinationStationName)

foreign key (departureStationName) references Station(name)

on delete cascade

foreign key (destinationStationName) references Station(name)

on delete cascade

Station(name)

Schedule(routeNo, departureTime, empld, licenseNo)

foreign key (routeNo) references Route(routeNo)

on delete cascade

foreign key (empld) references Driver(empld)

on delete set null

foreign key (licenseNo) references Bus(licenseNo)

on delete set null