

Homework 1

3.2 There are 22 distinct tuples are in a relation instance with cardinality ≥ 2 , because a relation is defined as a set of tuples.

3.4 Primary key is a unique and non-null key that identify a recording uniquely in the table.

A table can have only one primary key, but a table can have multiple candidate keys.

Candidate key may or may not be a primary key. A superkey is a set of attributes that contains a key.

3.5 1. The example of non-candidate key: Name, Age.

GPA can not be declared as an non-candidate key from the only figure 3.1.

2. We can't deduce any example of an attribute is a candidate key from the figure 3.1.

Because, the same relation may have an instance that contains a different set of tuples. we can't determine a key of a relation given only one instance of the relation. Also, the instance is "legal" is not important in this question.

3.6 • A foreign key is a column or combination of columns that is used to establish and enforce a link between the data in two tables to control the data that can be stored in foreign key table. A foreign key constraint is a statement of the form that one or more fields of relation.

• The foreign key constraints are important because they provide safeguards for insuring the integrity of data. Users are alerted / thwarted when they try to do something that does not make sense. This can help minimize errors in application programs or in data-entry.

• Referential integrity is a property of data stating that all its references are valid. which means all foreign key constraints are enforced.

3.7 1. Students, Faculty, Courses, and Rooms are the most basic relation, these must be free-standing.

Given: The primary key of Students, Faculty, Courses, Rooms respectively are sid, fid, cid, and rno.

- Student enrolled in Courses. \Rightarrow sid and cid should both have foreign key constraints place on them.
- Faculty teach students. \Rightarrow fid and cid should both have foreign key constraints place on them.
- Course meets in Room. \Rightarrow cid and rno should both have foreign key constraints place on them.

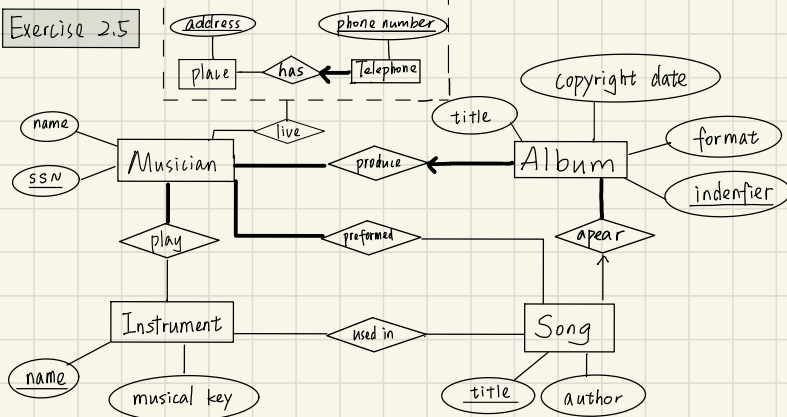
2. We could add the prerequisites, max and min credits per semester to enrolled relationship set. In this way, different students can be restricted in enrolling in the course.

3.12

- <1> Professor (ssn: varchar(10), primary key (ssn))
 Courses (course id: int, primary key (course id))
 Semester (semester id: varchar(10), primary key (semester id))
 Teaches (ssn: varchar(10), course id: int, semester id: varchar(10),
 primary key (ssn, course id, semester id),
 foreign key (ssn) references Professor,
 foreign key (course id) references Course,
 foreign key (semester id) references semester)
- <2> Professor (ssn: varchar(10), primary key (ssn))
 Courses (course id: int, primary key (course id))
 Teaches (ssn: varchar(10), course id: int, semester: varchar(10),
 primary key (ssn, course id),
 foreign key (ssn) references Professor,
 foreign key (course id) references Course)
- <3> Same as <2>
- <4> Professor (ssn: varchar(10), primary key (ssn))
 Courses (course id: int, primary key (course id))
 Teaches (ssn: varchar(10), course id: int, semester: varchar(10),
 primary key (ssn),
 foreign key (course id) references Course)
 foreign key (ssn) references Professor)
- <5> Same as <4>
- <6> Professor (ssn: varchar(10), primary key (ssn))
 Courses (course id: int, primary key (course id))
 Group (group id: varchar(10), primary key (group id))
 Member of (ssn: char(10), group id: varchar(10),
 primary key (ssn, group id),
 foreign key (ssn) references Professor,
 foreign key (group id) references Group)
 Teaches (group id: varchar(10), course id: int, semester: varchar(10),
 primary key (group id, course id),
 foreign key (group id) references Group,
 foreign key (course id) references Course)

3.15

Exercise 2.5



Musicians (ssn: varchar(10), name: varchar(20), primary key (ssn))

Instrument (name: varchar(20), musical key: varchar(10), primary key (name))

Album (identifier: int, format: varchar(5), title: varchar(20), copyright date: date, primary key (identifier))

Song: (title: varchar(20), author: varchar(20), primary key (title, author))

Place: (address: varchar(40), primary key (address))

Telephone: (phone number: int, primary key (phone number))

Has: (address: varchar(40), phone number: int, primary key (phone number), foreign key (address) reference place, foreign key (phone number) references Telephone.

Play: (ssn: varchar(10), name: varchar(20), primary key (ssn, name), foreign key (ssn) reference Musicians, foreign key (name) reference Instrument)

Used in: (name: varchar(20), title: varchar(20), primary key (name, title), foreign key (name) reference Instrument, foreign key (title) reference Song)

performed: (ssn: varchar(10), title: varchar(20), primary key (ssn, title), foreign key (title) references Song, foreign key (ssn) references Musicians)

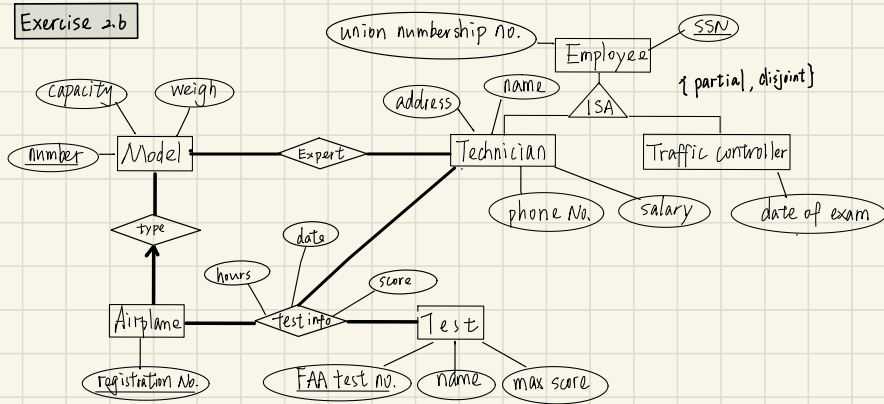
Produce: (ssn: varchar(10), identifier: int, primary key (identifier), foreign key (ssn) references Musicians)

Appear: (title: varchar(20), identifier: int, primary key (title), foreign key (identifier) references Album, foreign key (title) references Song)

Live: (ssn: varchar(10), phone number: int, primary key (ssn, phone number), foreign key (ssn) references Musicians, foreign key (phone number) references Home)

3.16

Exercise 2.b



Model : (number : varchar(10), capacity : int, weight : varchar(15), primary key (number))

Employee : (ssn : varchar(10), union membership no. : varchar(10), primary key (ssn))

Technician : (ssn : varchar(10), phone no : int, salary : varchar(10), address : varchar(20), name : varchar(20), primary key (ssn), foreign key (ssn) references Employee)

Traffic controller : (ssn : varchar(10), date of exam : date, primary key (ssn), foreign key (ssn) references Employee)

Airplane : (registration no. : int, primary key (registration no.))

Test : (FAA test no. : int, name : varchar(20), max score : int, primary key (FAA test no.))

Test info : (registration no. : int, FAA test no. : int, ssn : varchar(10), hours : int, date : date, score : int, primary key (registration no., FAA test no., ssn))

foreign key (registration no.) references Plane, foreign key (FAA test no.) references Test, foreign key (ssn) references Technician)

Type : (number : varchar(10), registration no. : int, primary key (registration no.))

foreign key (number) references Model, foreign key (registration no.) references Plane.

Expert : (number : varchar(10), ssn : varchar(10), primary key (number, ssn),

foreign key (number) references Model, foreign key (ssn) reference Technician)

We still can not modify the SQL statements defining the relations obtained by mapping the ER diagram to check this constraint. The Expert relation and the type relation must be equivalent.