

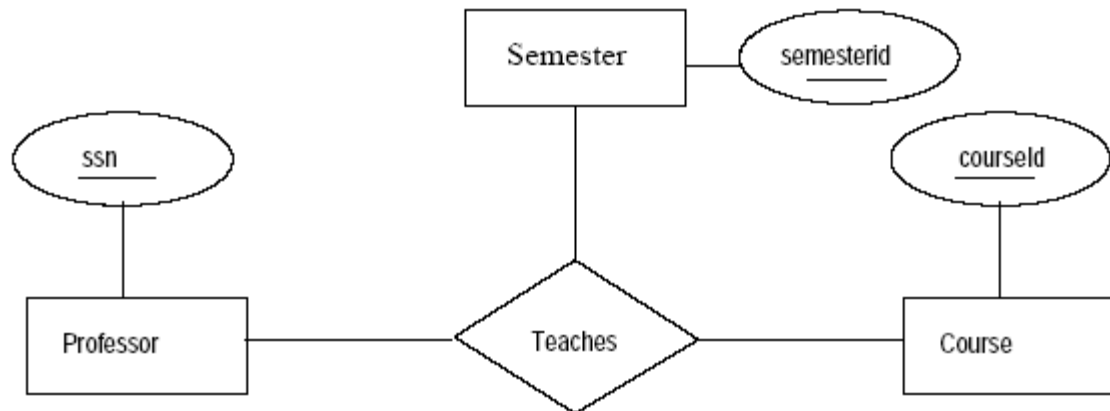
# Entity-Relationship Models

Dr. Bing Zhou

# Exercise #1

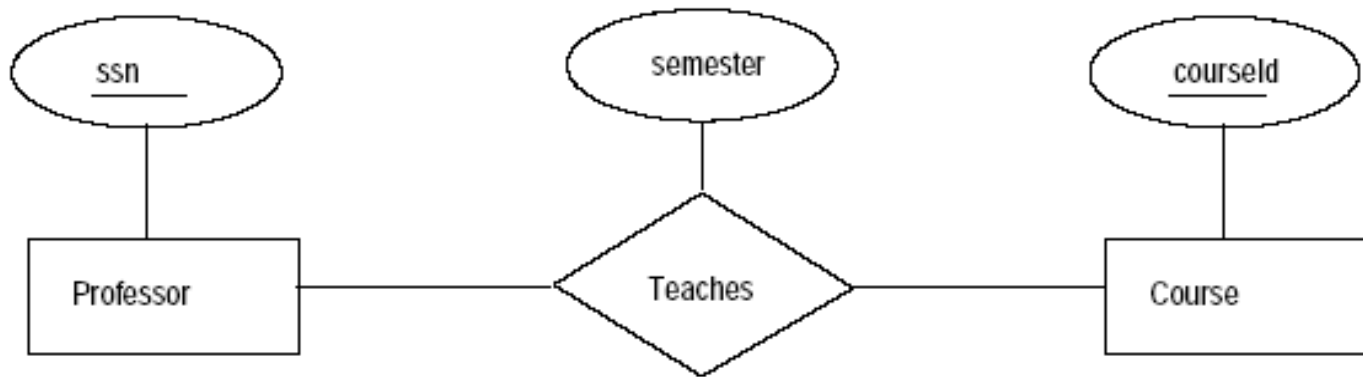
A university database contains information about **professors** (identified by social security number) and **courses** (identified by courseid). Professors **teach** courses; each of the following situations concerns the Teaches relationship set. For each situation, draw an ER diagram that describes it.

- Professors can teach the same course in several semesters, and each offering must be recorded.



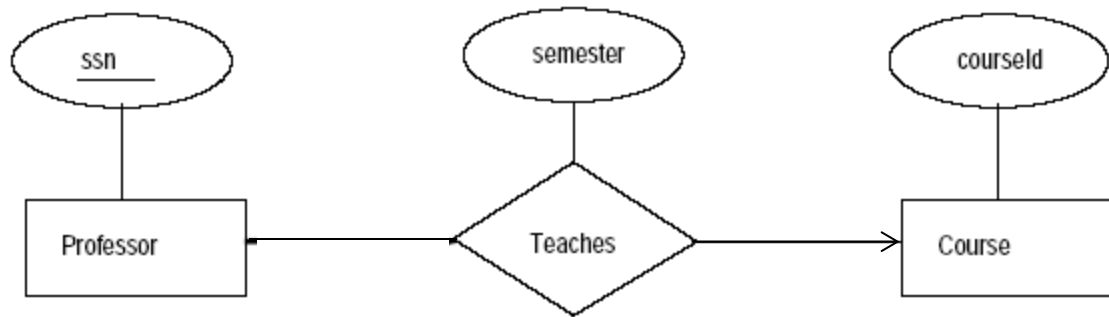
# Exercise #2

Professors can teach the same course in several semesters, and only the most recent such offering needs to be recorded.



# Exercise # 3

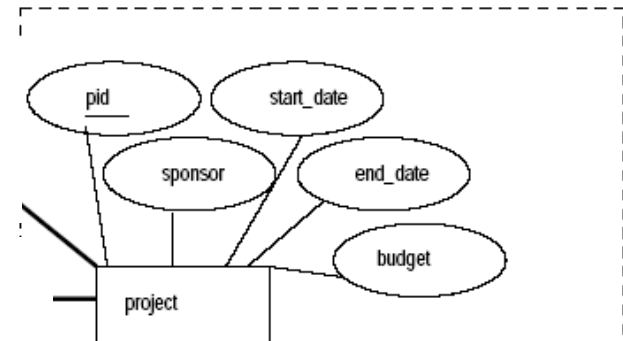
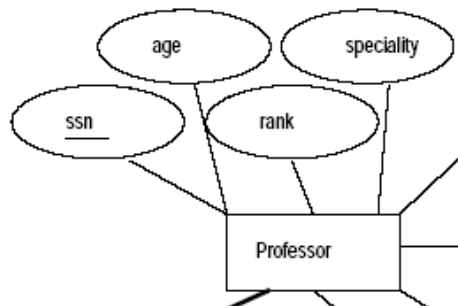
- Every professor teaches exactly one course (no more, no less)



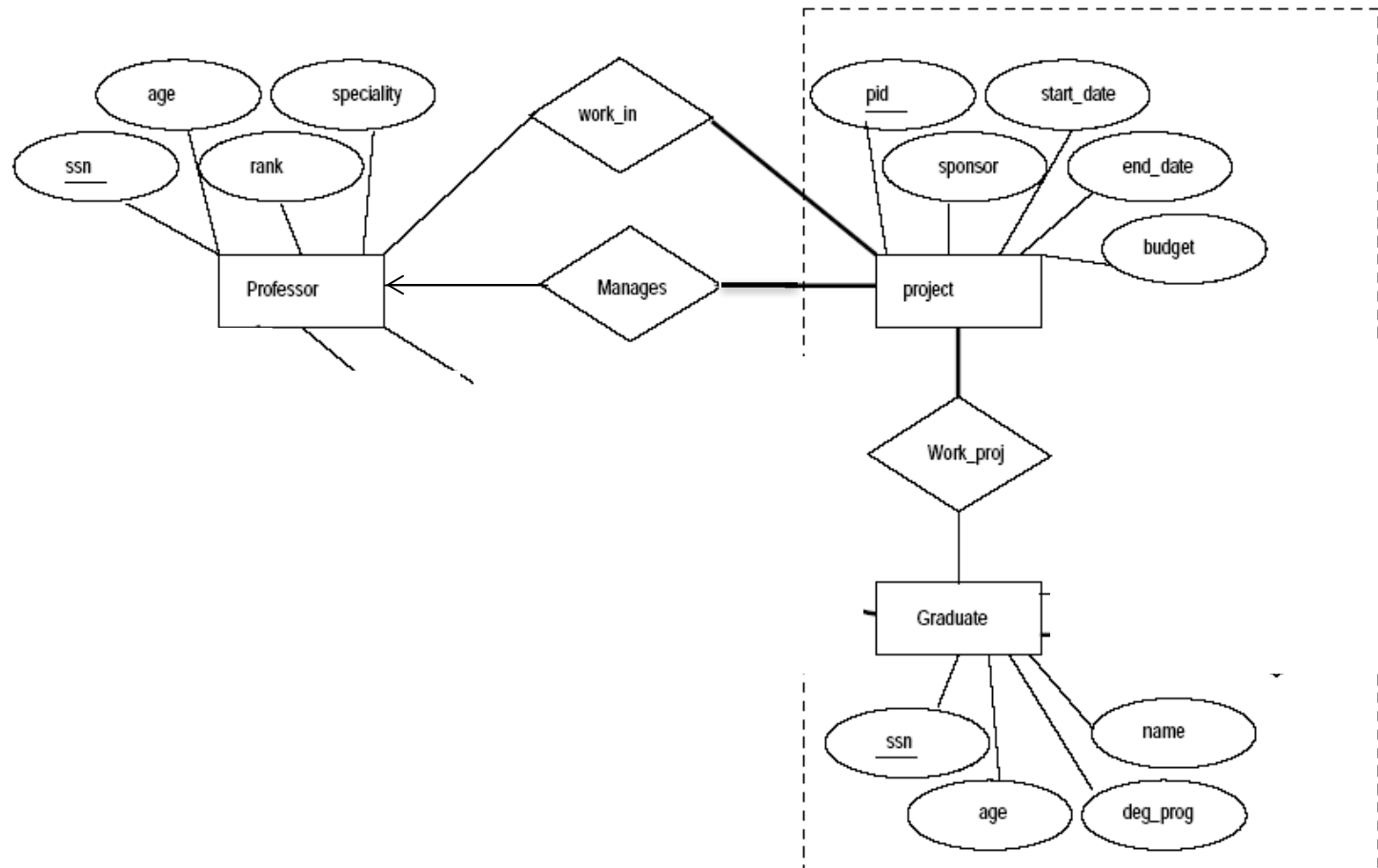
# Practice

- **Professors** have an SSN, a name, an age, a rank, and a research specialty.
- **Projects** have a project number, a sponsor name (e.g., NSF), a starting date, an ending date, and a budget.

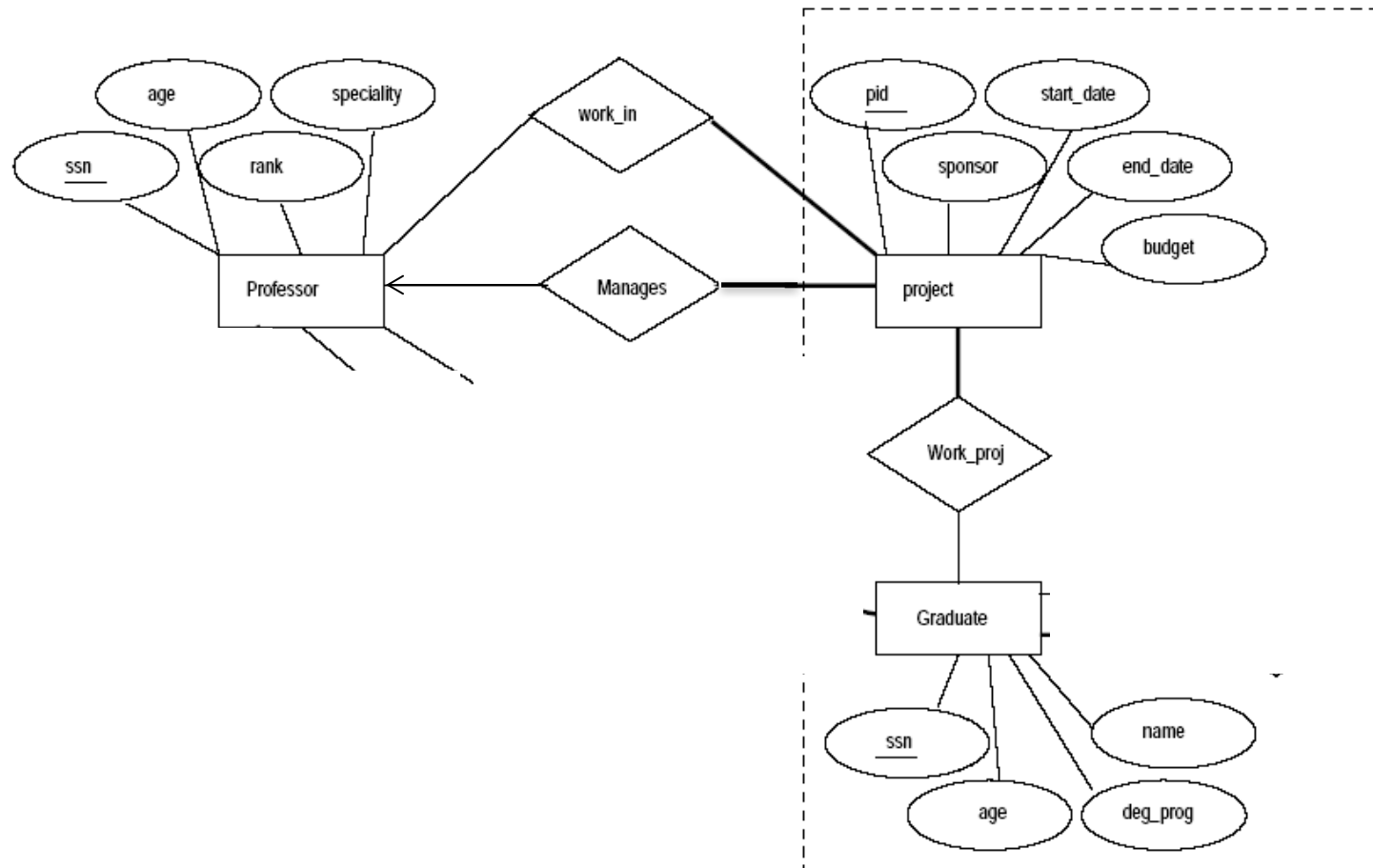
- Graduate students have an SSN, a name, an age, and a degree program
- Each project is **managed** by exactly one professor (known as PI)
- Each project is **worked in** by one or more professors (known as Co-PIs)
- Each project is **worked on** by one or more graduate students (known as RAs)



- Graduate students have an SSN, a name, an age, and a degree program
- Each project is **managed** by exactly one professor (known as PI)
- Each project is **worked in** by one or more professors (known as Co-PIs)
- Each project is **worked on** by one or more graduate students (known as RAs)

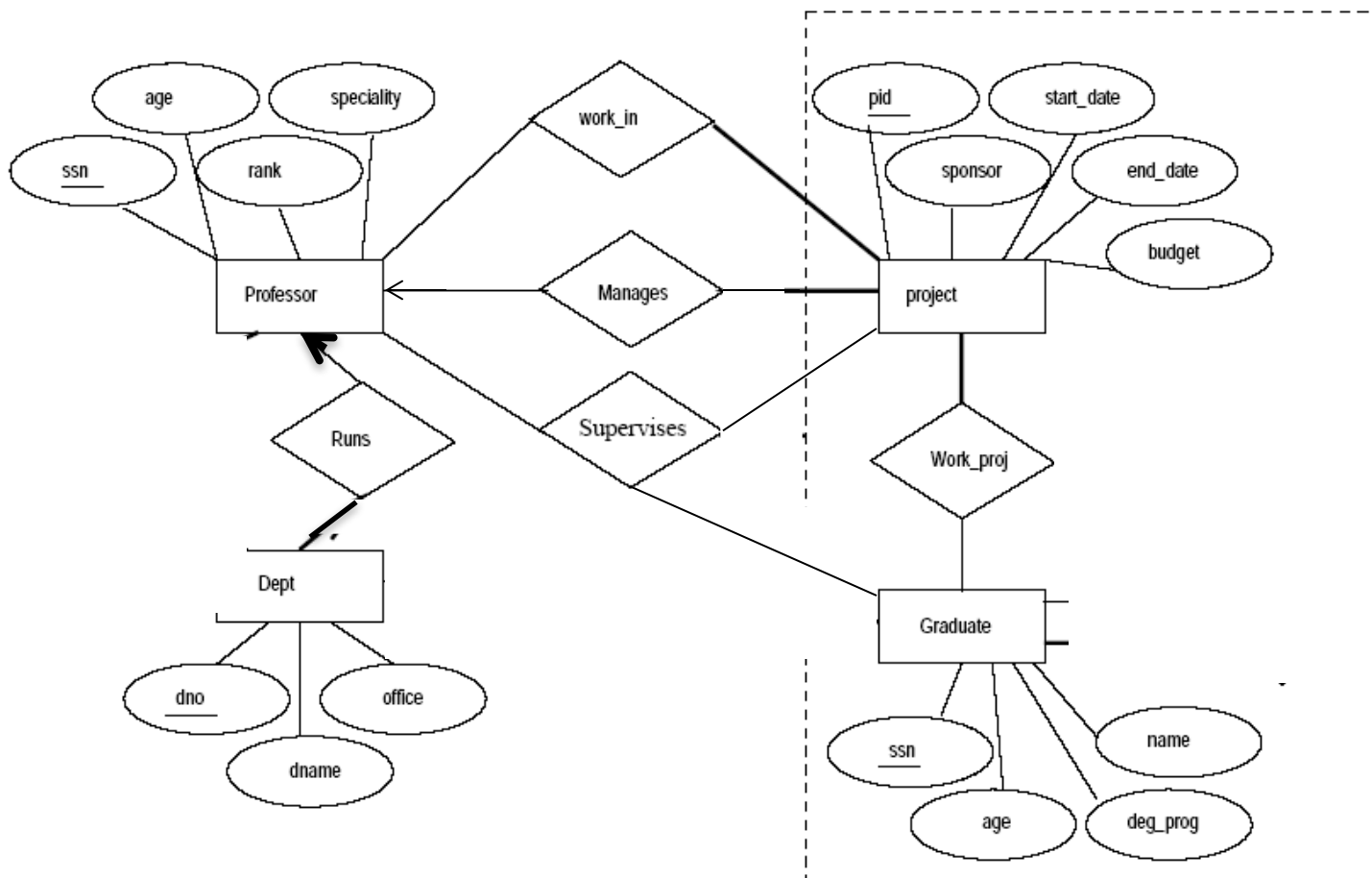


- When graduate students **work on** a project, a professor must **supervise** their work on the project. Graduate students can work on multiple projects, in which case they will have a potentially different supervisor for each project.
- Departments** have a department number, a department name, and a main office.
- Department has a professor (known as Chairman) who **runs** the department.

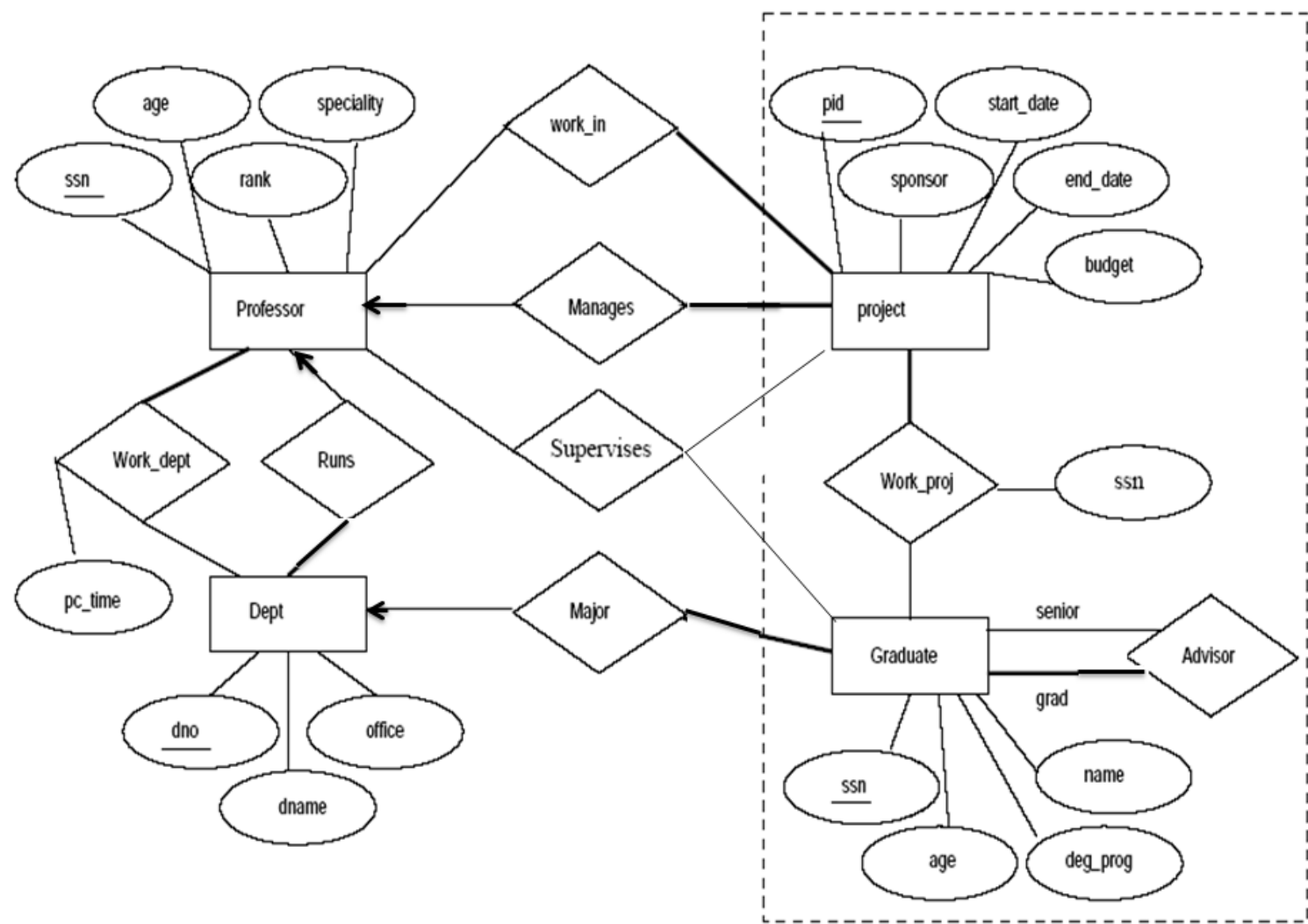


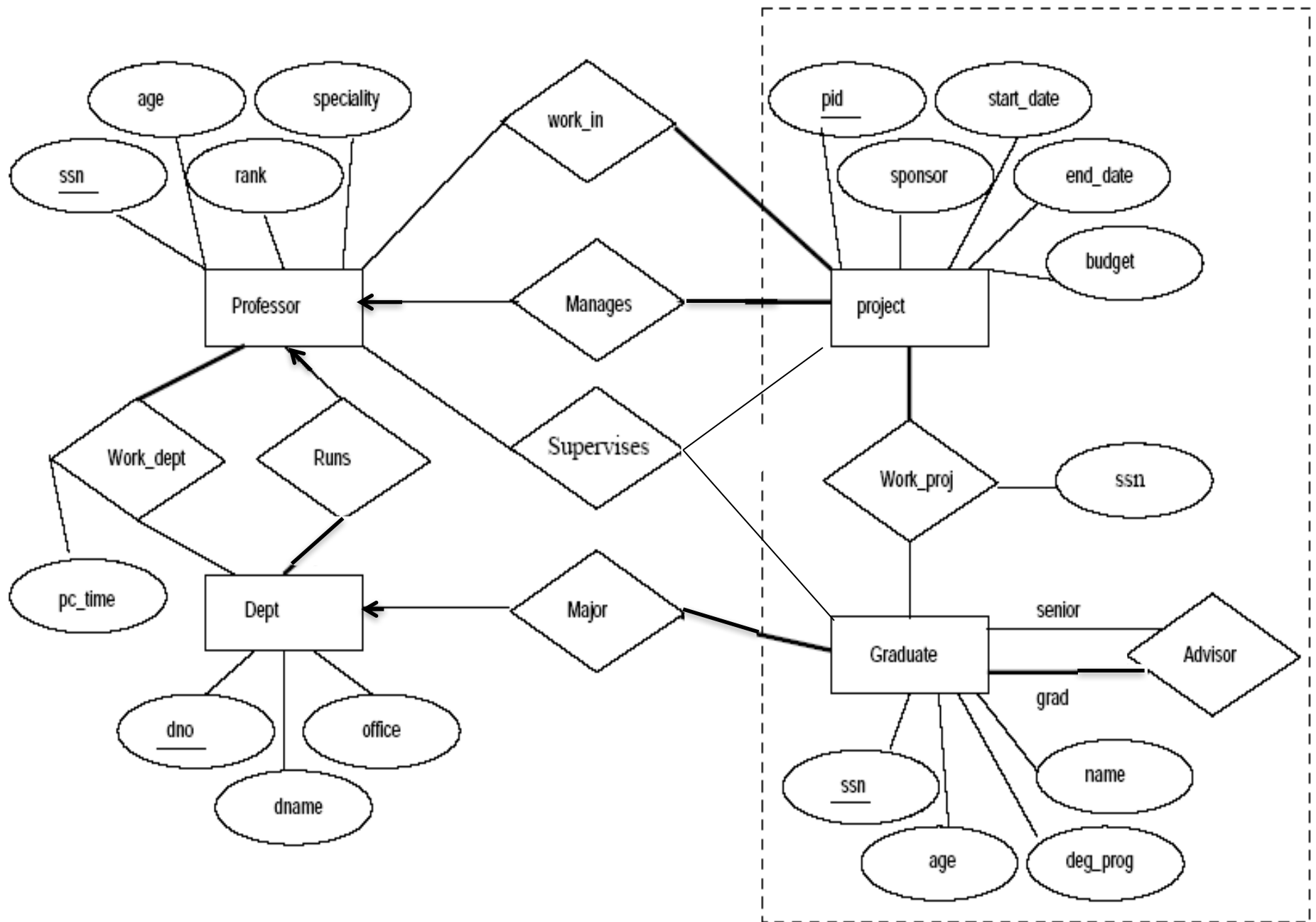


- When graduate students **work on** a project, a professor must **supervise** their work on the project. Graduate students can work on multiple projects, in which case they will have a potentially different supervisor for each project.
- Departments** have a department number, a department name, and a main office.
- Department has a professor (known as Chairman) who **runs** the department.



- Professors **work in** one or more departments, and for each department that they work in, a **time percentage** is associated with their job
- Graduate students have one **major** department in which they are working on their degree.
- Each graduate student must have another, more **senior** graduate student as an advisor.

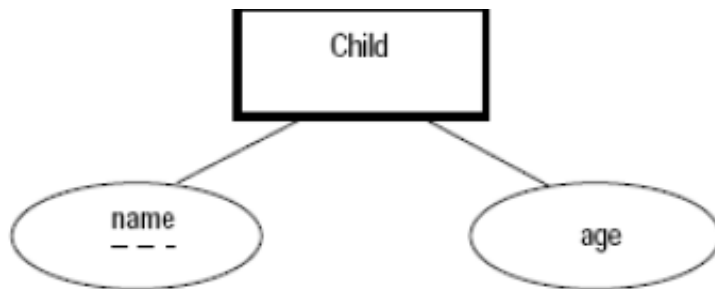
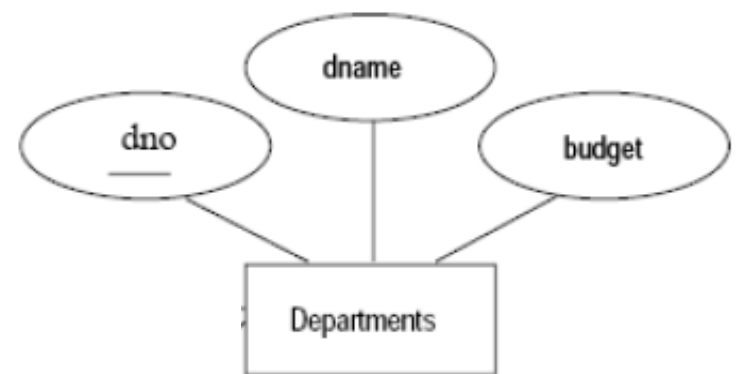




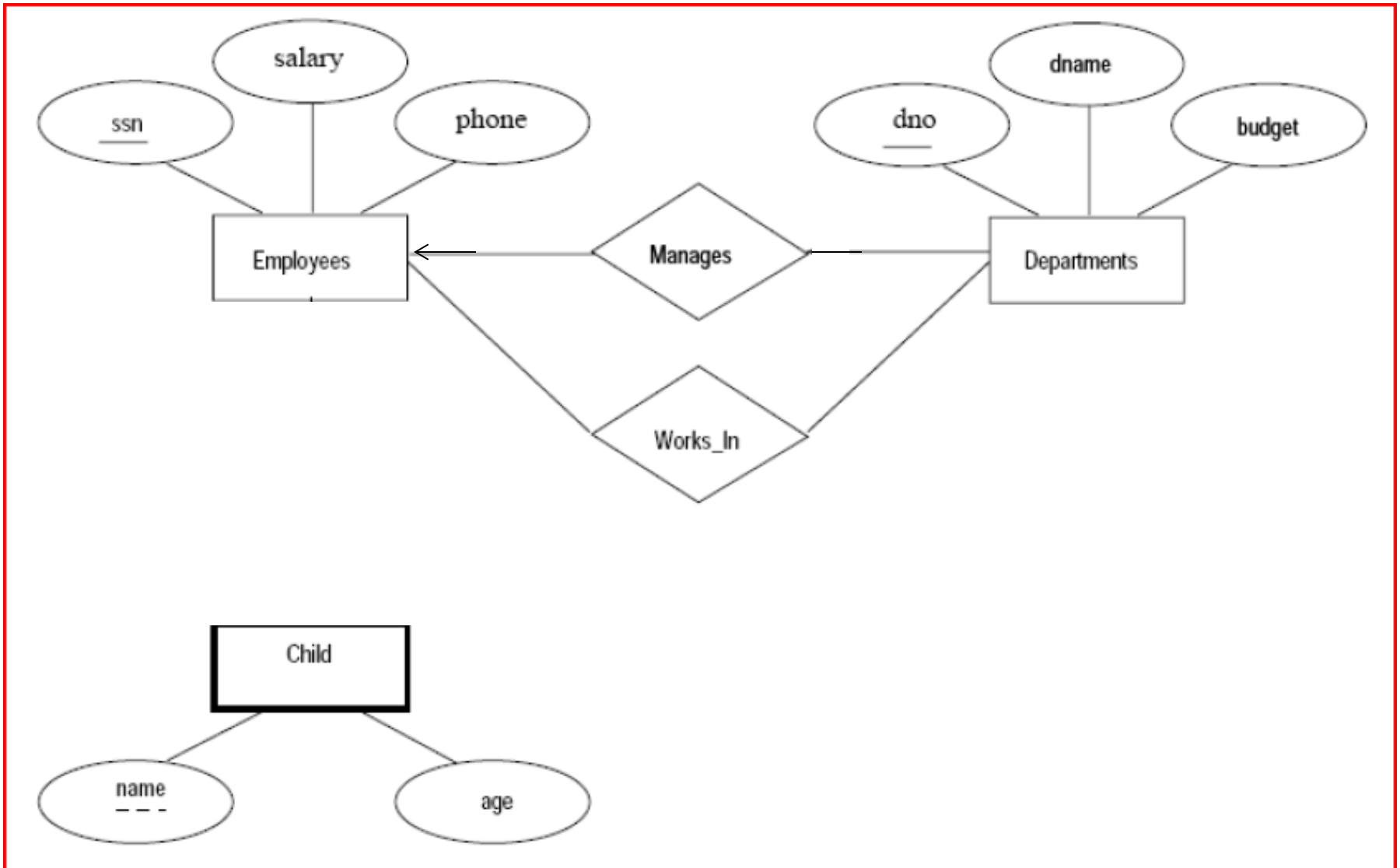
# Exercise # 5

- A company database needs to store information about employees (identified by *ssn*, with *salary* and *phone* as attributes), departments (identified by *dno*, with *dname* and *budget* as attributes), and children of employees (with *name* and *age* as attributes). Employees *work* in departments; each department is *managed* by an employee; a child must be identified uniquely by *name* when the parent (who is an employee; assume that only one parent works for the company) is known.
- Draw an ER diagram that captures this information.

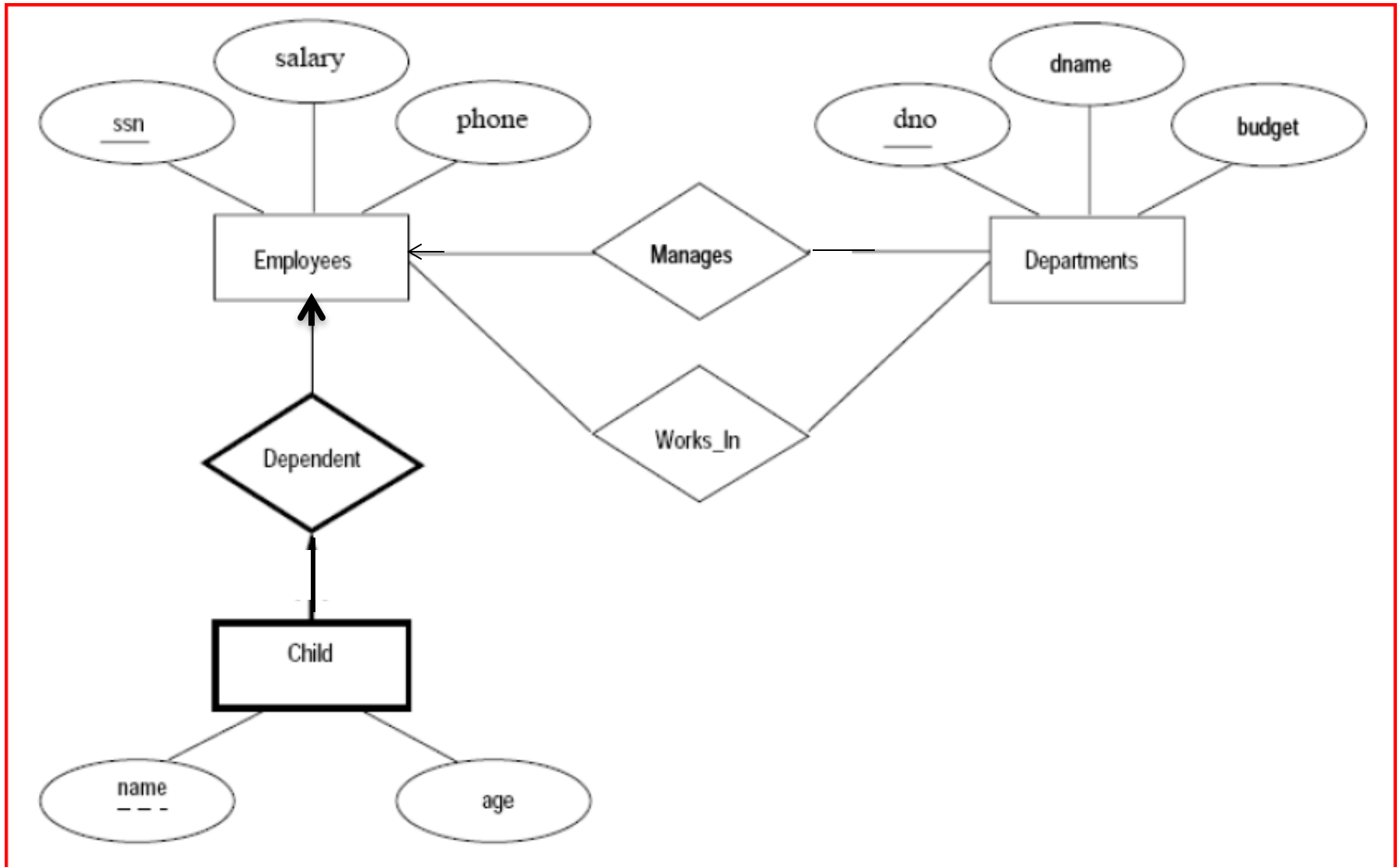
- A company database needs to store information about employees (identified by *ssn*, with *salary* and *phone* as attributes), departments (identified by *dno*, with *dname* and *budget* as attributes), and children of employees (with *name* and *age* as attributes).



- Employees *work* in departments;
- each department is *managed* by an employee;



- a child must be identified uniquely by *name* when the parent (who is an employee; assume that only one parent works for the company) is known.

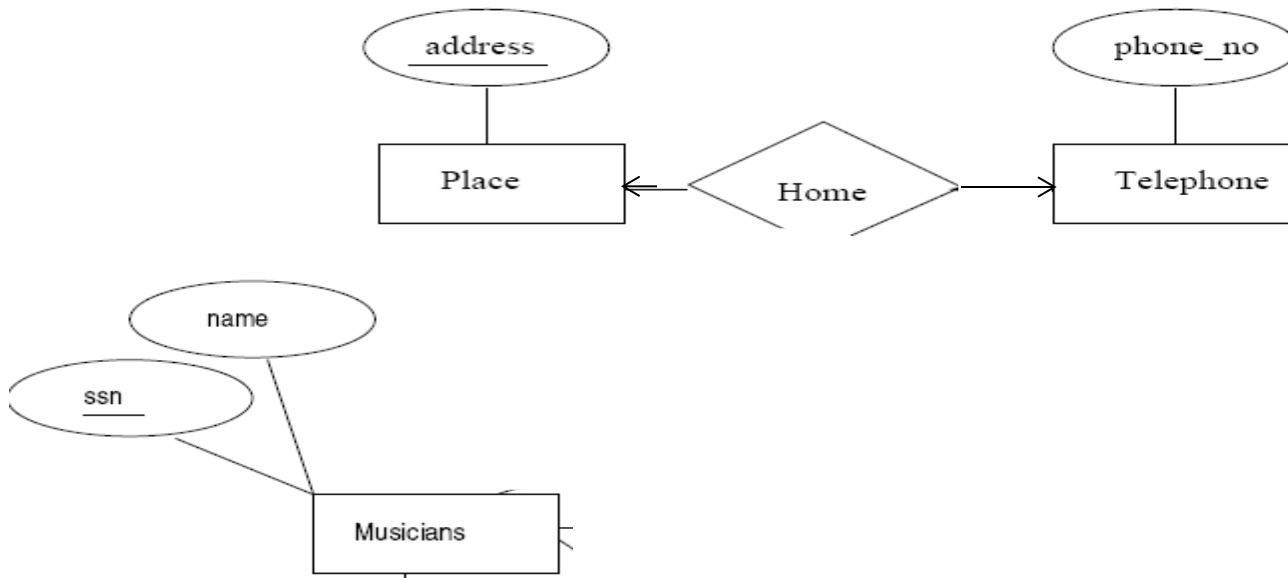


# Exercise # 6

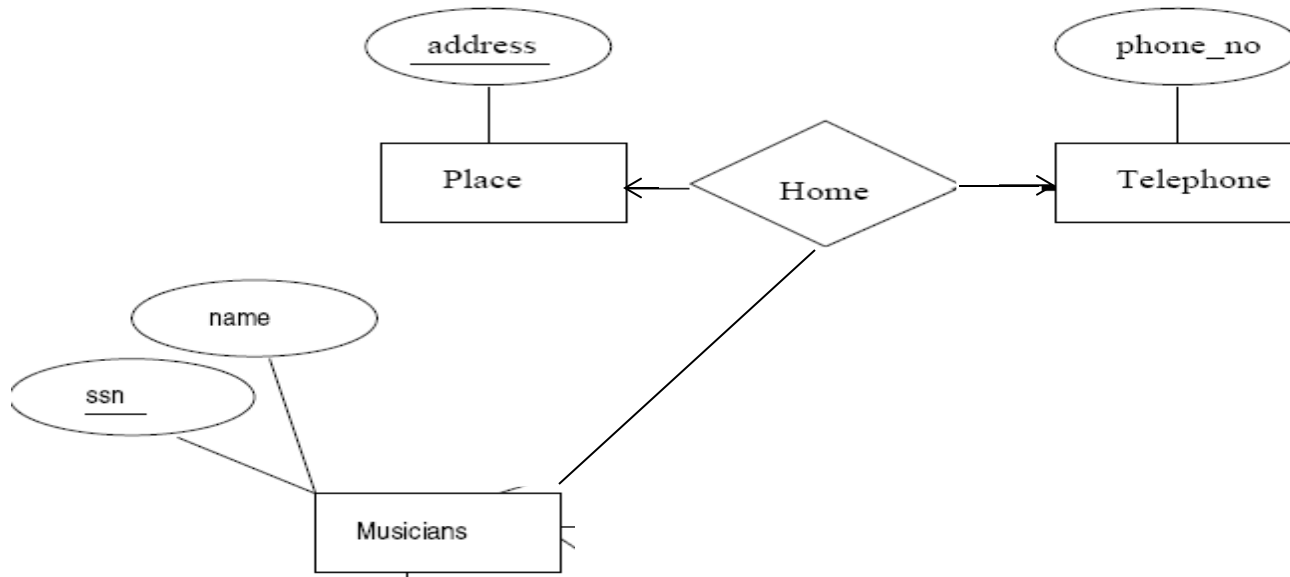
- Notown records has decided to store information on musicians who perform on their albums (as well as other company data) in a database. The company has chosen to hire you as a database designer.
  - Each musician that records at Notown has an SSN, a name, an address and a phone number. Poorly paid musicians often share the same address, and no address has more than one phone.
  - Each instrument that is used in songs recorded at Notown has a name (e.g. guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, Eflat).
  - Each album that is recorded at the Notown label has a title, a copyright date, a format (e.g., CD or MC) and an album identifier.
  - Each song recorded at Notown has an id, title and an author.
  - Each musician may play several instruments, and a given instrument may be played by several musicians.
  - Each album has a number of songs on it, but no song may appear on more than one album.
  - Each song is performed by one or more musicians, and a musician may perform a number of songs.
  - Each album has exactly one musician who acts as its producer. A producer may produce several albums.
- Draw an ER diagram for Notown.



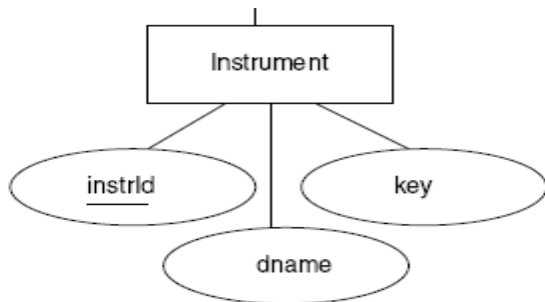
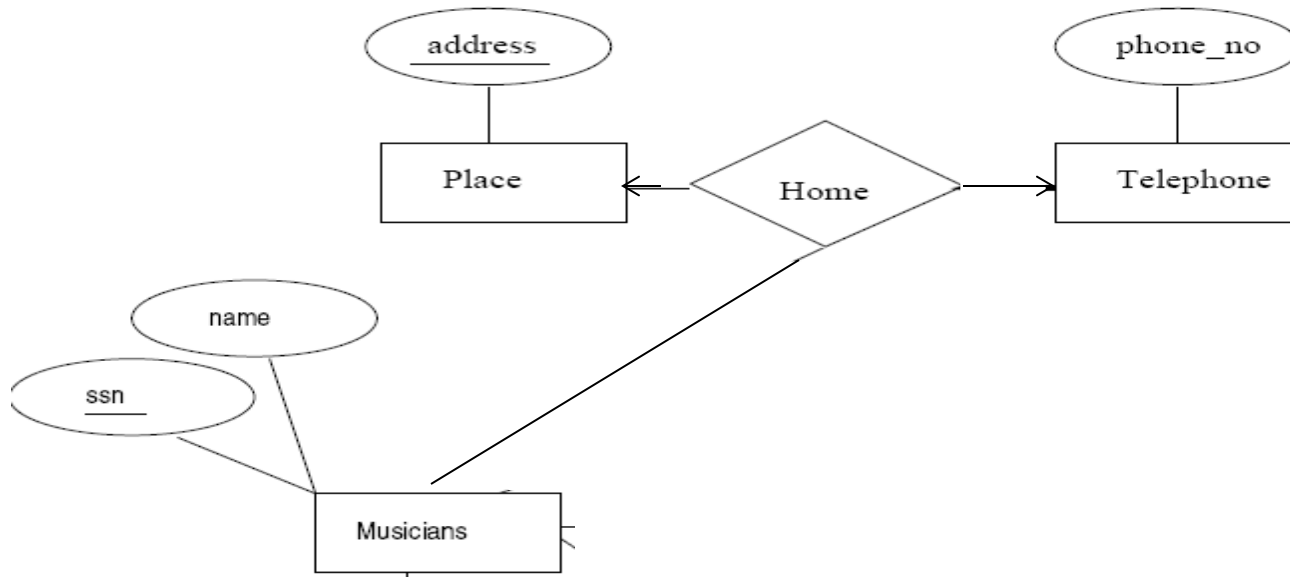
- Each musician that records at Notown has an SSN, a name, an address and a phone number. Poorly paid musicians often share the same address, and no address has more than one phone.



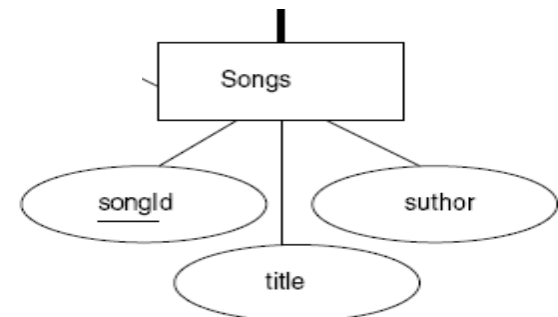
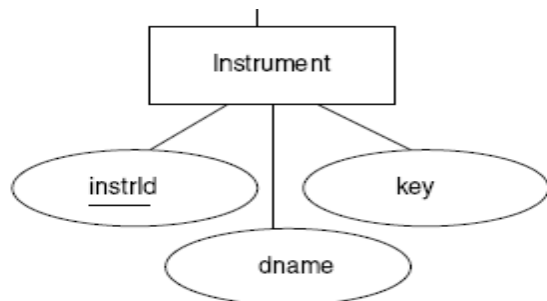
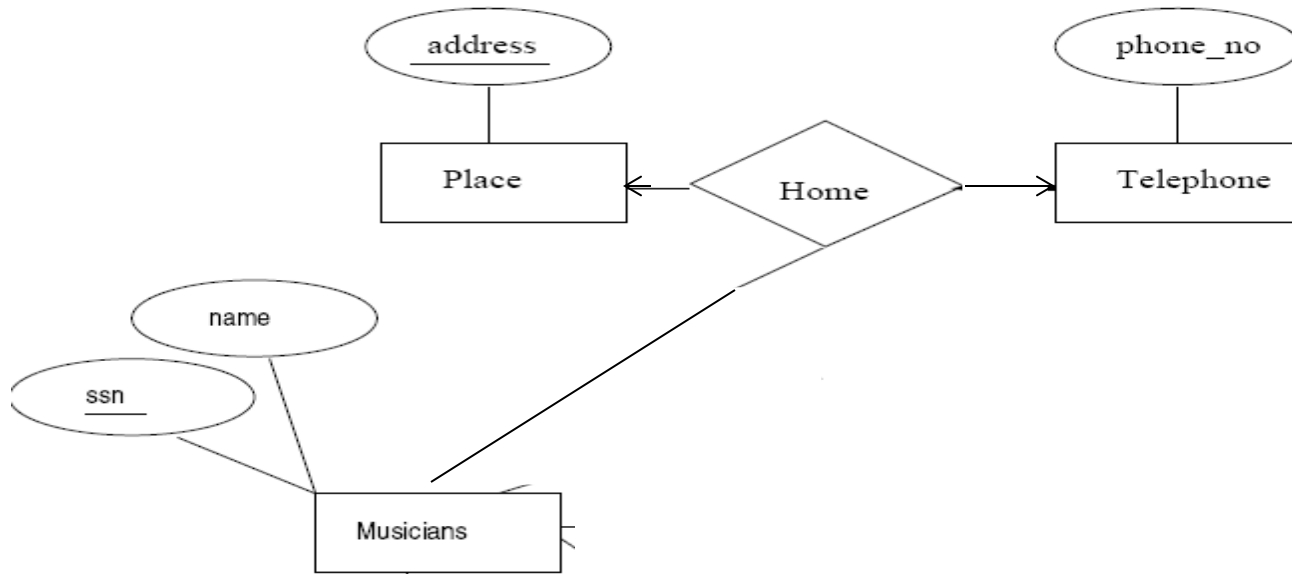
- Each instrument that is used in songs recorded at Notown has a name (e.g. guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, Eflat).



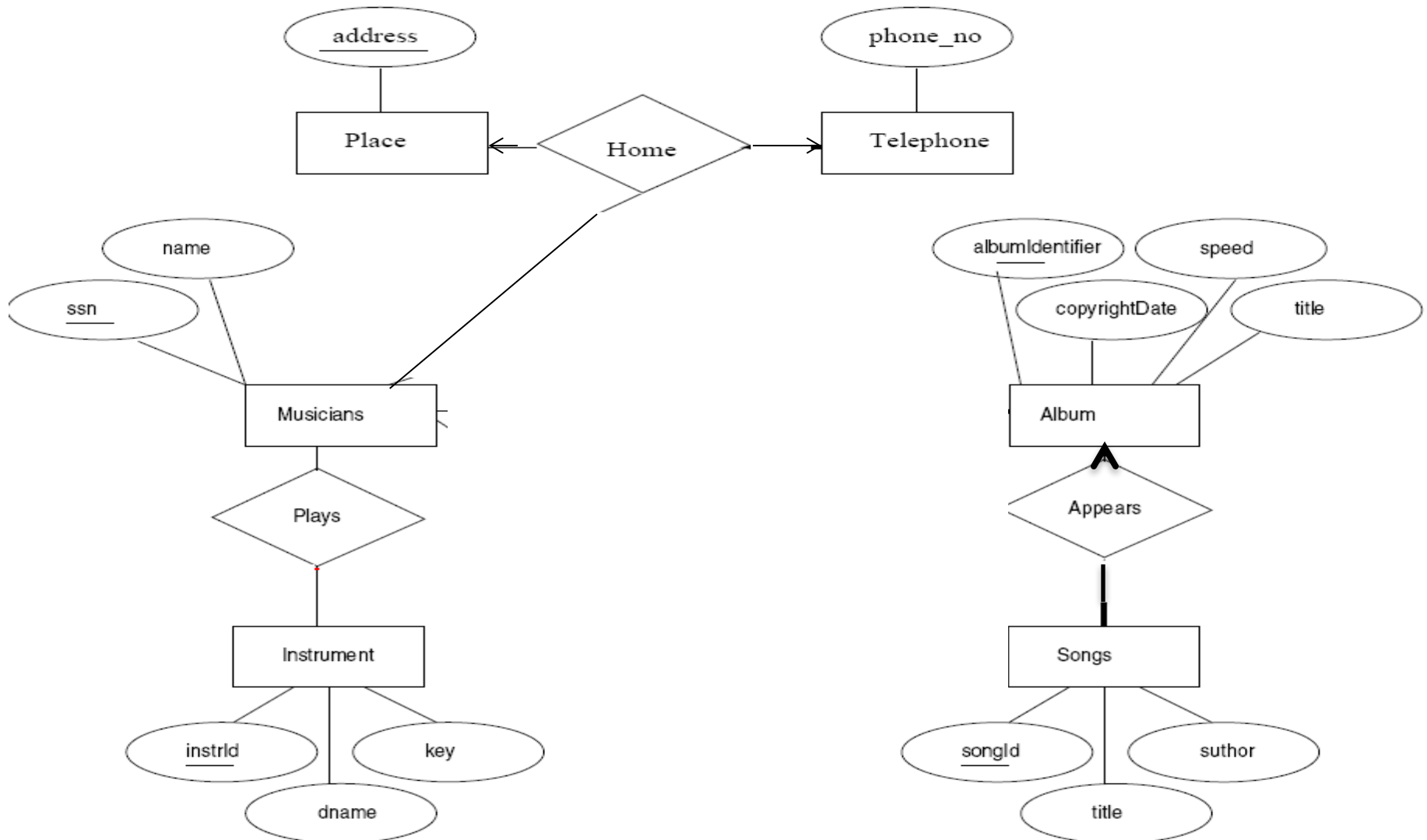
- Each song recorded at Notown has an id, title and an author.



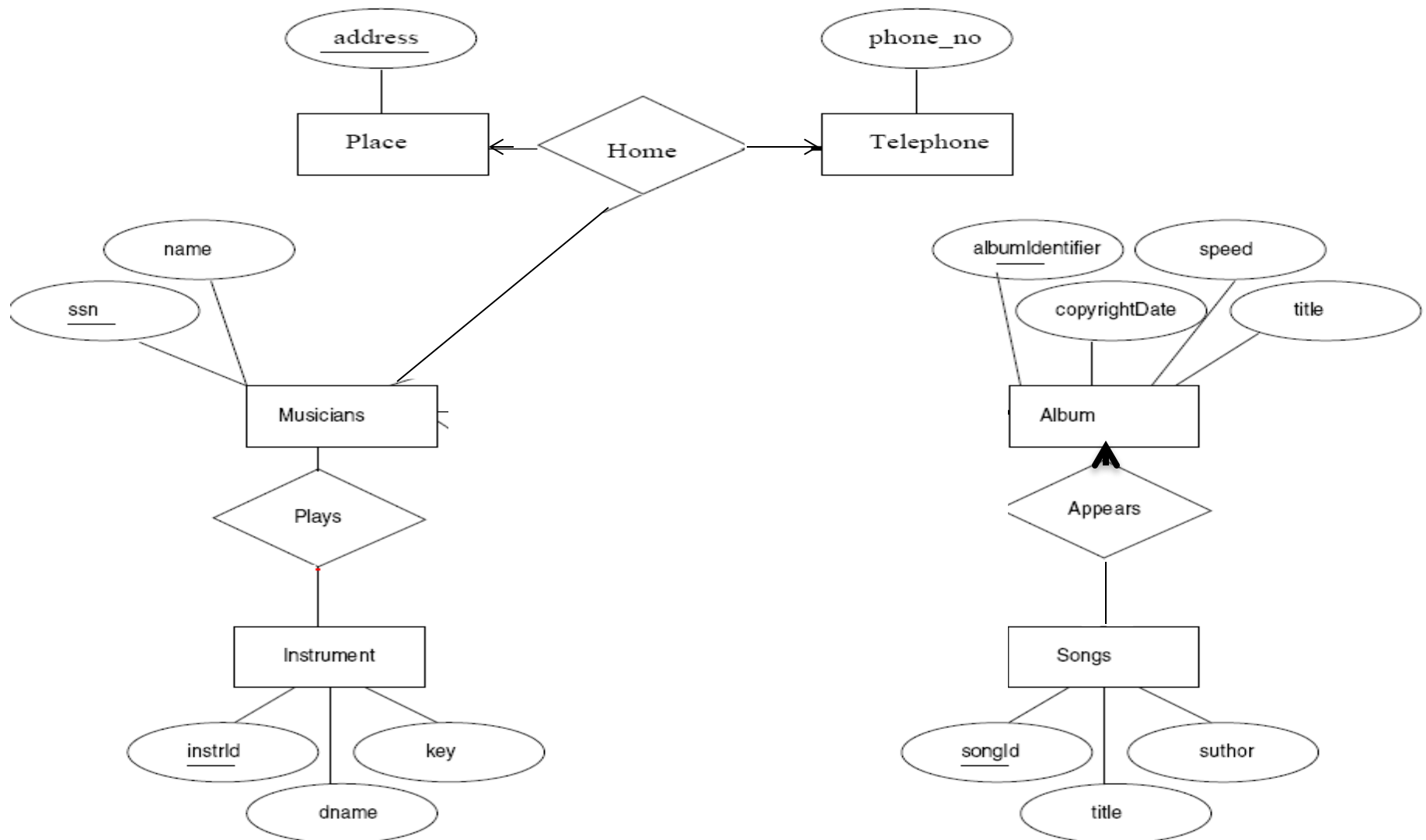
- Each musician may play several instruments, and a given instrument may be played by several musicians.



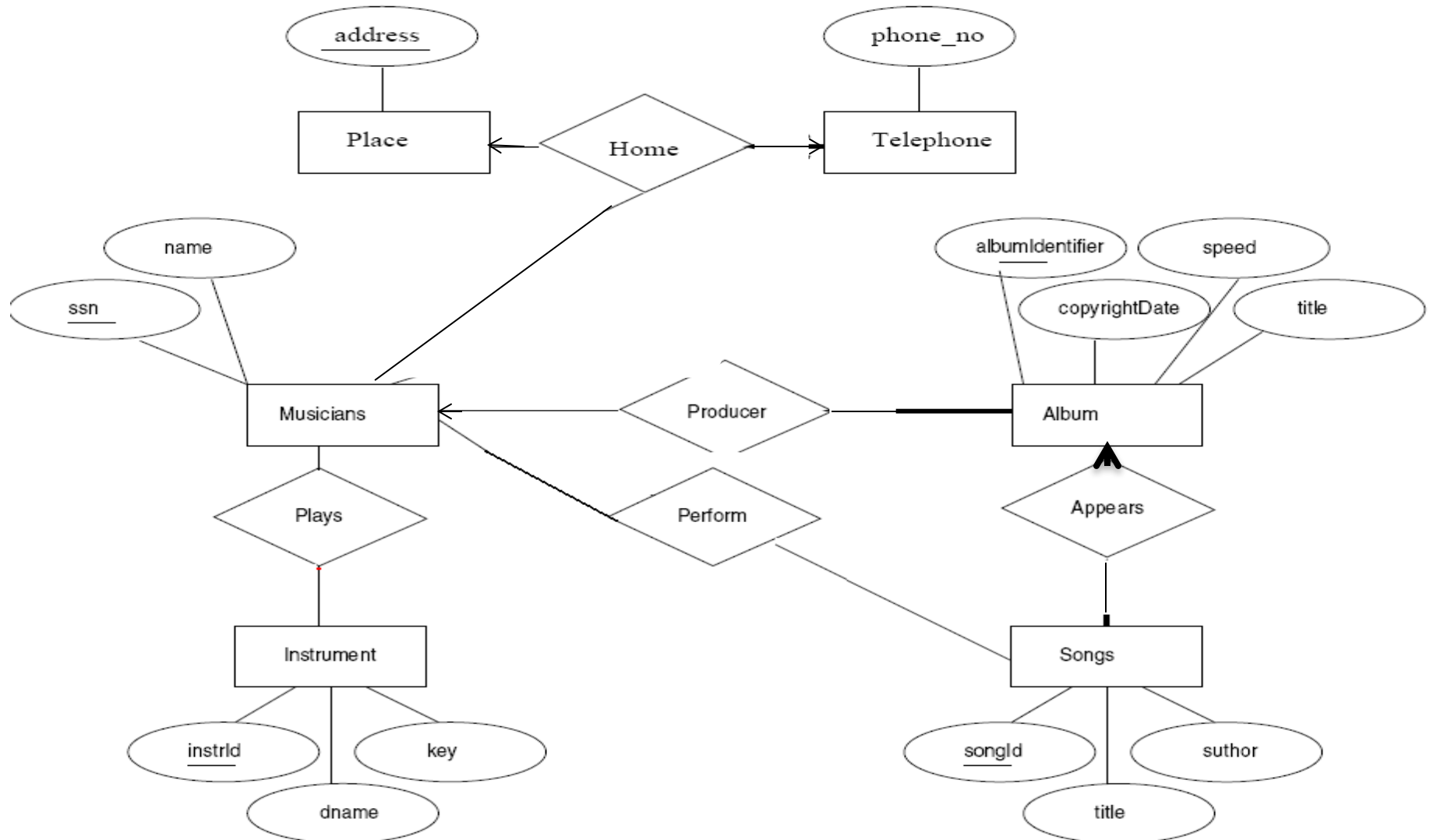
- Each album that is recorded at the Notown label has a title, a copyright date, a format (e.g., CD or MC) and an album identifier.
- Each album has a number of songs on it, but no song may appear on more than one album.

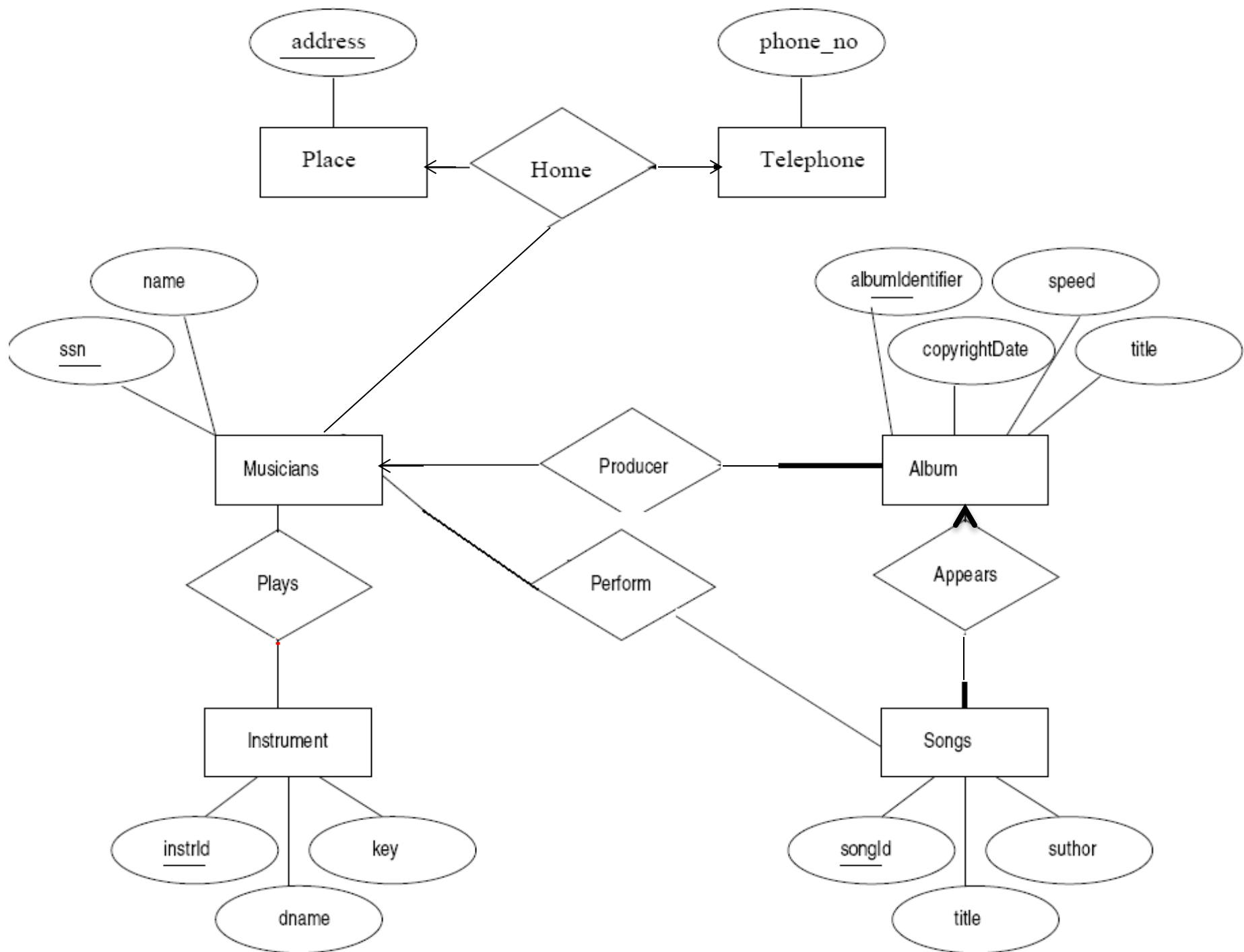


- Each song is performed by one or more musicians, and a musician may perform a number of songs.

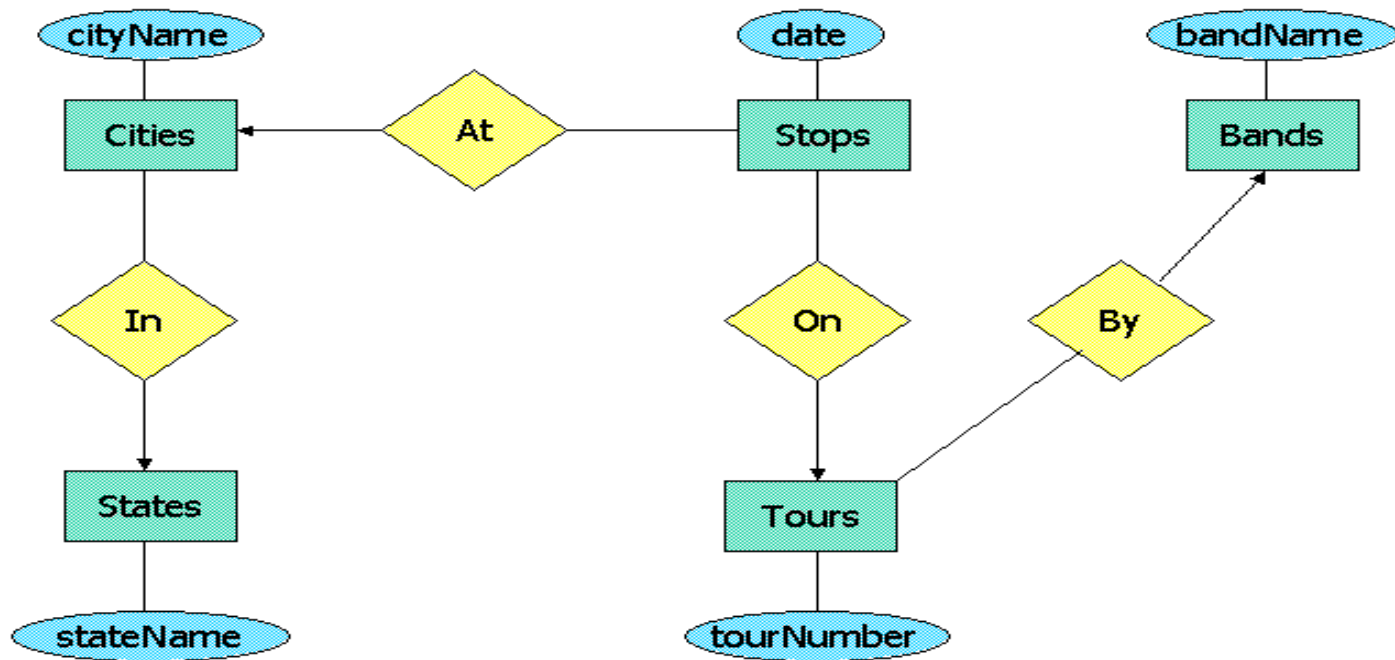


- Each album has exactly one musician who acts as its producer. A producer may produce several albums.









- The E/R Model describes a database about bands and their tours. A tour consists of a sequence of cities visited by a band. We assume that no city is visited twice on a single tour, and on one date, a band can visit only one city. Some of the entity sets are weak, and some of the relationships are supporting many-one relationships, but all double rectangles and double diamonds are **not** shown.
- Your task is to decide which of the entity sets are weak, and which relationships support them. If there is a choice, prefer to use the attributes of the entity set itself in the key, and minimize the number of supporting relationships. There may be more than one reasonable answer, especially since "tour#" can have several reasonable interpretations. You should pick from the list below the one that is most reasonable.

# Correct Answers

- Stops is weak, supported by On.
- Tours is weak, supported by By; tour# is unique only for a given band.
- Tours is not weak; tour# is a globally unique ID.
- Cities is weak, and its key is the city name and the name of the state it is In.

# INCORRECT ANSWERS (with explanation)

- Stops is weak, supported by both At and On; date is not a key attribute.
  - We are told that a tour cannot stop at two cities on the same date. Thus, once we know the date and the tour that a Stop represents, we don't also need the City to have a key for Stops.
- Stops is weak, supported by both At and On; date is a key attribute.
  - We are told that a tour cannot stop at two cities on the same date. Thus, once we know the date and the tour that a Stop represents, we don't also need the City to have a key for Stops.
- Stops is weak, supported by At.
  - A date and a city does not define a unique Stop (City/Tour pair). The reason is that several tours may stop at the same city on the same date.
- Cities is not weak.
  - The city name by itself does not determine a unique city. There can be cities of the same name in two or more different states, e.g., Portland ME and Portland OR.
- Bands is weak, supported by By.
  - A supporting relationship has to be many-one *from* the entity set it supports. In order for this diagram to make sense, we have to assume that band names are unique, because By cannot help define a band uniquely using the weak entity set construct.

- States is weak, supported by In.
  - A supporting relationship has to be many-one *from* the entity set it supports. Note also that state names are unique.
- Stops is weak, and its key is the date and the name of the band making the tour the stop represents.
  - There is no many-one relationship directly from Stops to Bands. Thus, an attribute of Bands can only contribute to the key of Stops if Tours is weak and the band name contributes to the key of Tours. But then, the tour# would also be in the key for Stops.
- Stops is weak, and its key is the date, the name of the associated city, and the associated tour#.
  - Since it is ambiguous whether tour#'s are unique, it is possible that there are two bands that use the same tour number, and these bands stop in the same city on the same date.
- Tours is weak, supported by On and By.
  - A supporting relationship has to be many-one *from* the entity set it supports. Thus, On couldn't possibly support Tours.