

Assignment 1

Due Monday, Feb 19 at 11:59pm

General Instructions

- Feel free to talk to other members of the class in doing the homework. You should, however, write down your solutions yourself. *List the names of everyone you worked with at the top of your submission.*
- Keep your solutions brief and clear.
- Please use Piazza if you have questions about the homework but do not post answers. Feel free to use private posts or come to the office hours.

Homework Submission

- We DO NOT accept late homework submissions.
- We will be using Compass for collecting the homework assignments. Please submit your answers via [Compass](#). Hard copies are not accepted.
- Contact the TAs if you are having technical difficulties in submitting the assignment; attempt to submit well in advance of the due date/time.
- The homework must be submitted in **pdf** format. Scanned handwritten and/or hand-drawn pictures in your documents won't be accepted.
- Please do not zip the answer document (PDF) so that the graders can read it directly on Compass. You need to submit one answer document, named as **hw1_netid.pdf**.
- Please see the [assignments](#) page for more details. In particular, we will be announcing errata, if any, on this page.

1 Short Questions (20 pts)

Provide a short answer (4 sentences at most) for each of the following questions. You may use figures if necessary. If you are including a figure do not handdraw it. You are free to use annotation tools such as Mac Preview or Microsoft PowerPoint to draw the ER diagrams.

1. [3] Say the key for a relation comprises two attributes A and B. Then, no two tuples can have the same value for A or the same value for B. Justify or prove otherwise.

Solution: Incorrect. A set of attributes forms a key for a relation if we do not allow two tuples in a relation instance to have the same values in all the attributes of the key, so two tuples can have the same value for either A or B, just not both.

2. [8] True/False questions - If true, please justify; if false, please provide a counterexample (which does not need to be original - you can look at the slides and textbook).

- A weak entity set cannot have relations with other non-supporting entity sets.

Solution: False - see textbook page 170 figure 4.32, where the weak entity set *Courses* has an isa relationship with its subclass *Lab Courses* (any reasonable counterexample is acceptable).

- A subclass entity set cannot have relations with other non-related (non sibling nor ancestor) entity sets.

Solution: False - see textbook page 137 figure 4.10, where the subclass *Cartoons* has a *Voices* relationship with a set of *Stars* who speak but do not appear in the movie (any reasonable counterexample is acceptable).

- A weak entity set is produced only when translating a multi-way relation to a binary relation.

Solution: False - see textbook page 152 example 4.20 and 4.21, where the weak entity sets are not produced when translating multi-way relations to binary relations, but emerge due to their structural natures (any reasonable counterexample is acceptable).

- To identify entities in a child subclass, we need the key of the root entity set.

Solution: True - see textbook page 165 section 4.6.

3. a. [6] When translating a subclass hierarchy with n children entity sets in an ER model into the relational model with the O-O approach, what is the smallest number of relations one can get, as a function of n ? Justify your answer.

Solution: The basic idea of the O-O approach is to enumerate all possible subtrees of the hierarchy, so if the hierarchy is a linear chain, we will have $n + 1$ relations.

- b. [3] When translating a subclass hierarchy in an ER model into the relational model, the O-O approach always produces more relations than the straight-ER approach. Use your answer in part a, justify or prove otherwise.

Solution: Incorrect. As we have stated for part a, if the hierarchy is a linear chain, the OO approach would produce $n + 1$ relations, the same as the number of relations produced via the ER approach.

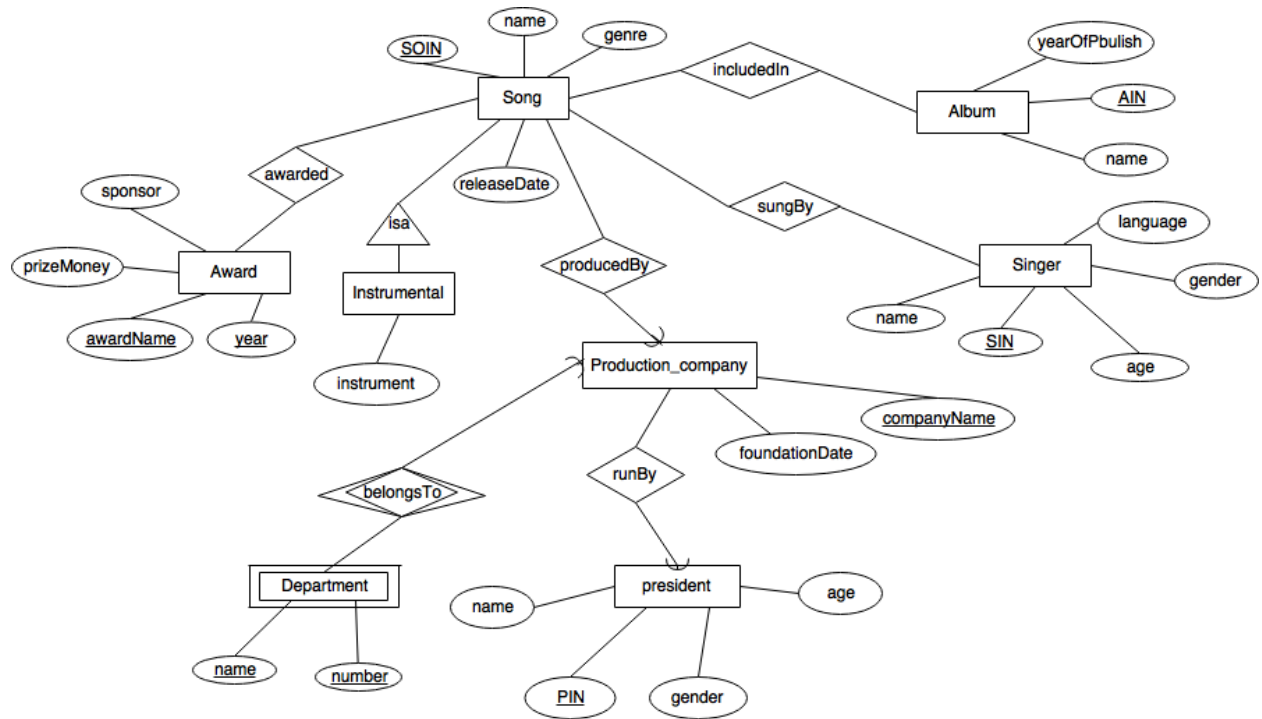
2 ER Models (35 pts)

Consider the following information about a music database.

1. There are many songs in the database, and each has a unique identification number 'SOIN', a name, a genre, and a release date. Each song is sung by one or more singers, is produced by exactly one production company, and can belong to one or more albums. In addition, some songs are purely instrumental, and in this case, they have one additional attribute, instrument.
2. Each song may end up winning zero or more awards. Each award is identified by the award name and year, and also has other attributes prize money and sponsor. The same award can be given to different songs.
3. Each production company has a unique company name, the date it was founded, and is run by exactly one president who is associated with a unique identification number 'PIN', name, gender, and age; each president can run multiple companies.
4. There are different departments in each production company, and they are uniquely identified by their department names, department numbers, as well as the company to which they belong.
5. Each singer is assigned a unique identification number 'SIN' and has a name, gender, language, and age as part of their information.
6. Each album has a unique identification number 'AIN', a name, and a year of publication. It can include one or more songs.

Design and draw an ER diagram that captures the aforementioned information. Underscore the primary key of each entity. Please do not add any additional entity set yourself by inferring.

You are free to use annotation tools such as Mac Preview or Microsoft PowerPoint to draw the ER diagrams. Please do not include scanned pictures.



3 Relational Model (35 pts)

Convert the ER model from the previous question to a relational model. For translating the subclass hierarchy, use the straight-ER approach. Please underscore the primary key of each entity, and merge relations as far as possible to minimize redundancy.

Solution: The relational model is as follows:

```
Song(SOIN, name, genre, releaseDate, companyName)
Instrumental(SOIN, instrument)
Award(awardName, year, prizeMoney, sponsor)
Production_company(companyName, foundationDate, PIN)
President(PIN, name, gender, age)
Department(name, number, companyName)
Singer(SIN, name, gender, language, age)
Album(AIN, name, yearOfPublish)
SungBy(SIN, SOIN)
Awarded(SOIN, awardName, year)
IncludedIn(SOIN, AIN)
```

4 DDL Commands (10 pts)

Consider the following relational schemas:

Professor (NetID, name, department, officeAddress, officePhone, email)

Student (NetID, name, department, graduationYear)

1. [5] Write the DDL commands that define each schema as a table.

Solution:

If the key is indicated in the first line, we do not need the last line.

```
(a)      CREATE TABLE Professor (
           NetID VARCHAR(any reasonable number) PRIMARY KEY/UNIQUE,
           name VARCHAR(any reasonable number),
           department VARCHAR(any reasonable number),
           officeAddress VARCHAR(any reasonable number),
           officePhone VARCHAR(any reasonable number),
           email VARCHAR(any reasonable number),
           PRIMARY KEY/UNIQUE (NetID)
        );
```

```
(b)      CREATE TABLE Student (
           NetID VARCHAR(any reasonable number) PRIMARY KEY/UNIQUE,
           name VARCHAR(any reasonable number),
           department VARCHAR(any reasonable number),
           graduationYear INT,
           PRIMARY KEY/UNIQUE (NetID)
        );
```

2. [5] Delete graduationYear in the Student table and add an attribute GPA, setting its default to 4.0.

Solution:

```
ALTER TABLE Student DROP graduationYear;
```

```
ALTER TABLE Student ADD GPA DEFAULT FLOAT '4.0';
```