

## Homework #3

**Complete By:** Wednesday, September 18<sup>th</sup> @ 5:00pm  
**Submission:** submitted digitally through Gradescope

### Database Domain Design

For this assignment you will be performing the design of a database in the domain of your choice. Think of an area in which data is stored and accessed. Your database design should satisfy the following criteria.

- Your database should have at least 4 tables
- Your database should have at least one of each of these types of relationships (partial, total, to one, to many)
- Each table in your database should have a primary key labelled if appropriate and multiple attributes
- At least 1 relationship should have 1 additional attribute.
- Each entity should have at least 1 relationship with another entity (no entities should be entirely standalone)

To help you get started, you can answer the following questions. If you have your handout from day one of class you can reuse your answers. You will not be graded on these answers only the diagram and schema on the following pages.

Choice of Domain:

TV Show

What do you store data about in this domain?

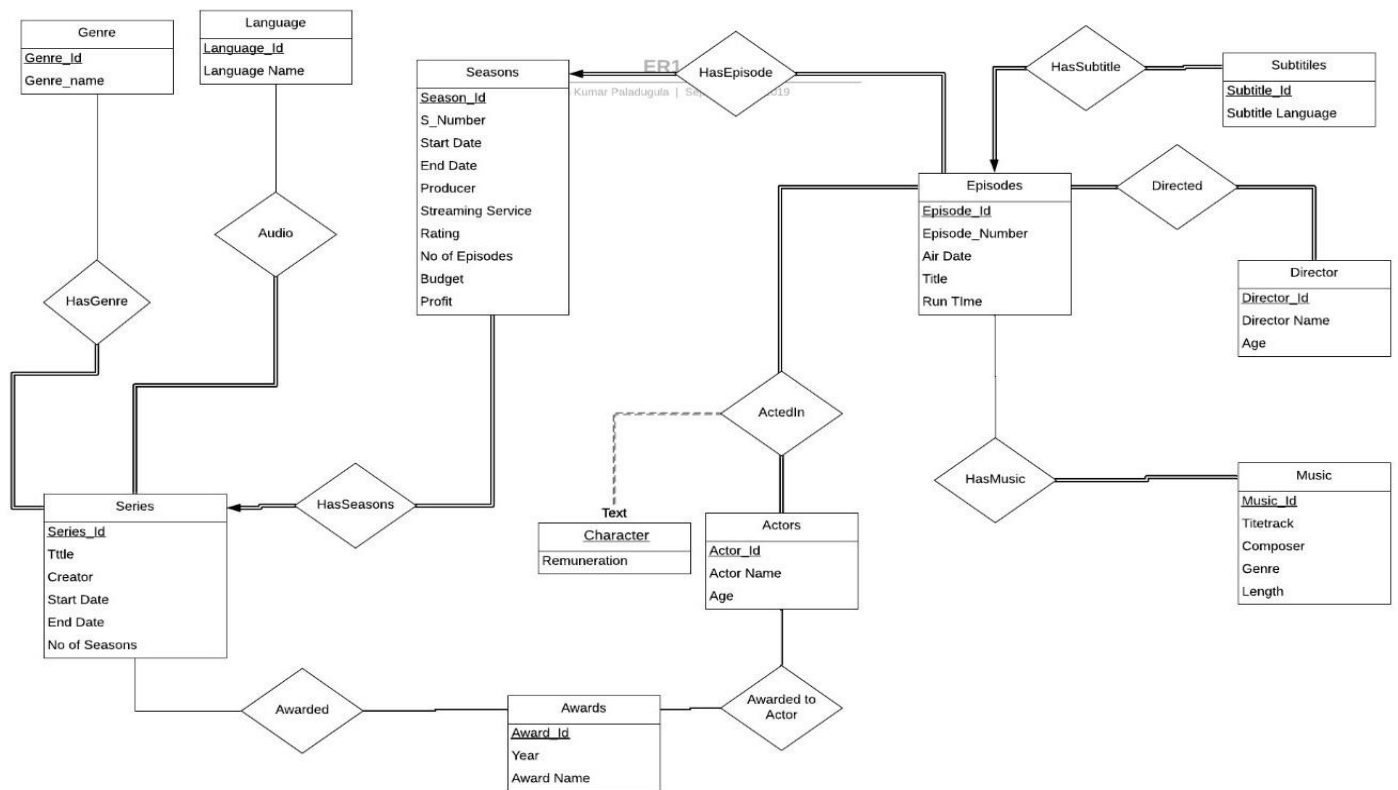
Seasons, Episodes, actors, directors, composers, awards etc

How is data currently stored or recorded?

How is the data connected?

What implicit knowledge do experts in the domain have that isn't contained in the raw data?

1. Construct an Entity Relationship diagram representing the domain of your choice and satisfying the conditions on the previous page. You may either draw the diagram on the page or include a picture taken of a physical drawing or image created by another program such as lucidchart like we did in class.



2. Now convert your Entity Relationship diagram into a relational schema following the guidelines in chapter 7. Be sure to include foreign keys where relevant. Write the schema out in the form of  
TableName(primary key fields, other attributes) attributes foreign key referencing other tables  
like we did in class. You may include a schema diagram for clarity, but we will be grading the text version.

Series (Series\_Id, Title, Start Date, End Date, No of seasons, Creator)

Genre (Genre\_Id, Genre\_Name)

HasGenre (Series\_Id, Genre\_Id) Series\_Id is a foreign key here referencing Series;

Genre\_Id is a foreign key here referencing Genre

Language (Language\_Id, Language Name)

HasLanguage (Series\_Id, Language\_Id) Series\_Id is a foreign key here referencing Series;

Language\_Id is a foreign key here referencing Languages

Seasons (Season\_Id, Season Number, Start Date, End date, No of Episodes, Streaming Service, Rating, Budget,  
Profit, Series\_Id) Series\_Id is a foreign key here referencing Series;

Episodes (Episode\_Id, Title, Episode Number, Air Date, Run time, Season\_Id)

Season\_Id is a foreign key here referencing Seasons

Music (Music\_Id, Titletrack, Genre, Composer, length)

HasMusic (Episode\_Id, Music\_Id) Episode\_Id is a foreign key here referencing Episodes;

Music\_Id is a foreign key here referencing Music

Actors (Actor\_Id, Actor Name, age)

Actedin (Episode\_Id, Actor\_Id, Character, Remuneration)

Episode\_Id is a foreign key here referencing Episodes;

Actor\_Id is a foreign key here referencing Actors

Awards (Award\_Id, Award Name, year)

Awarded (Series\_Id, Award\_Id) Series\_Id is a foreign key here referencing Series;

Award\_Id is a foreign key here referencing Awards

Awarded to Actor (Actor\_Id, Award\_Id)

Actor\_Id is a foreign key here referencing Actors;

Award\_Id is a foreign key here referencing Awards

Directors (Director\_Id, Director Name, age)

Directed (Episode\_Id, Director\_Id) Episode\_Id is a foreign key here referencing Episodes;

Director\_Id is a foreign key here referencing Directors

Subtitles (Subtitle\_Id, Subtitle Language, Episode\_Id)

Episode\_Id is a foreign key here referencing Episodes