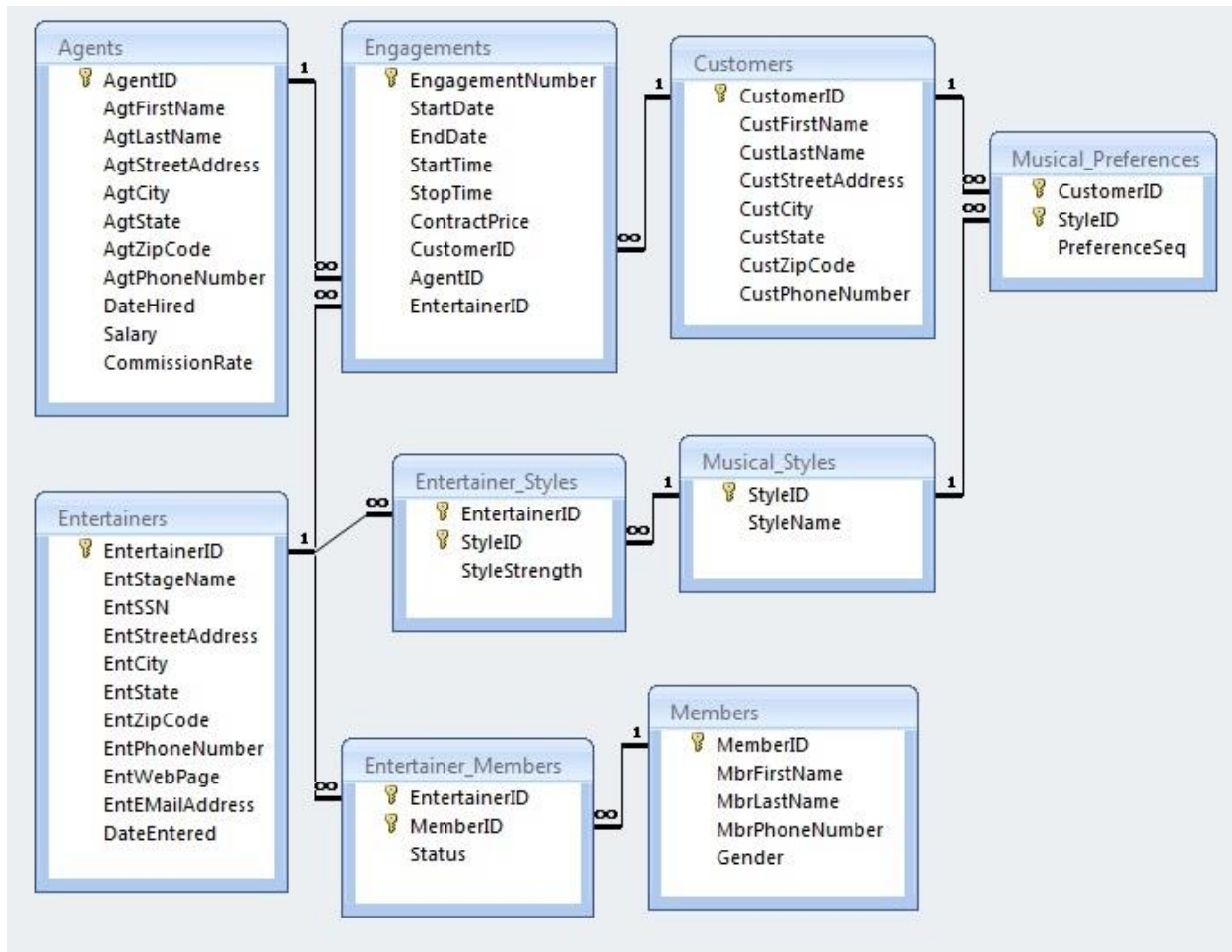


Homework 5 – Data Modeling

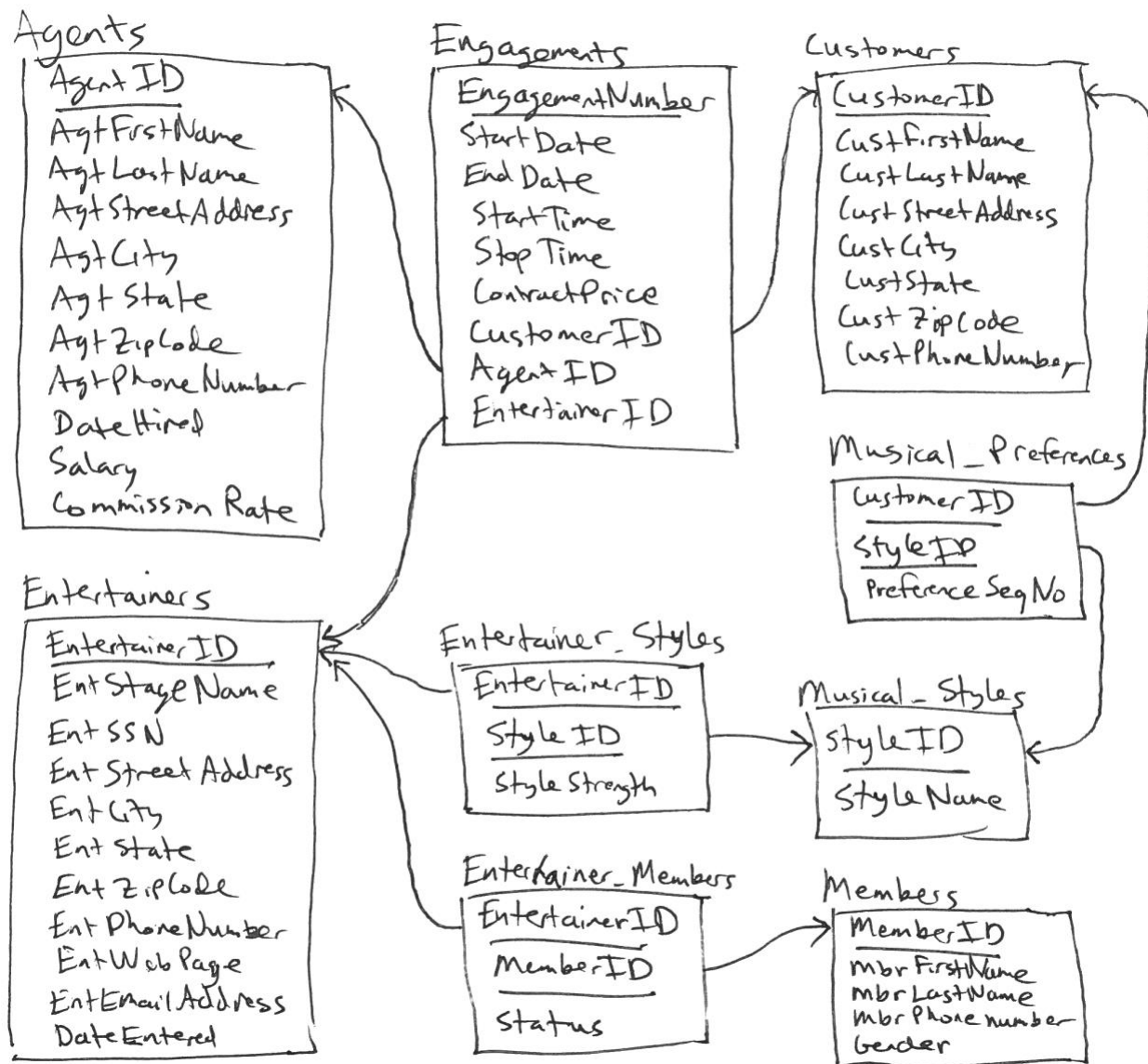
Part 1:



The diagram uses slightly different conventions than what I described in class. In particular:

- It uses yellow key icons to represent primary keys (instead of underlining), and
- Foreign keys are represented with lines that start with an infinity sign and end with a "1". My diagrams instead use an arrow that starts at the foreign key and points to the referenced column.

Following the conventions described in class, the diagram can be rewritten as follow:



Let's also assume that all of the columns are required (there can be no NULL values).

The questions below refer to the diagram above.

1. How do we know that this schema design requires **exactly one customer** per engagement?
2. How do we know that a single customer can belong to many different engagements?
3. If an entertainer has three different styles, how many rows for her would be present in the Entertainers table, and how many rows for her in the Entertainer_Styles table?
4. Which of these tables are highest-level parents?
5. Why is there a separate Musical_Preferences table? In other words, why not just create a PreferredStyle column in the Customers table?
6. Why is a **composite primary key** used in the Entertainer_Styles table (instead of a single-column primary key)?

7. What are the many-to-one relationships in this schema (if any)?
8. What are the many-to-many relationships in this schema (if any)?
9. What are the one-to-one relationships in this schema (if any)?
10. How would we change this schema to prevent Entertainer Stage Names (EntStageName) from being repeated by different entertainers?
11. Let's say we wanted to remove the AgentID column from the Agents table and instead use AgtPhoneNumber as the primary key in this table. What else would have to change in the schema? What data would have to change if an Agent suddenly changed his/her phone number?
12. Let's change the schema as follows. Assume that most engagements are free and therefore they have no *ContractPrice*, *CustomerID*, nor *AgentID*. However, some engagements do have these three values. What would be the best way to represent this?

Part 2:

Define a relational data model using the "tables and arrows" diagram style. You can pick any topics in business, social, scientific, or industrial phenomenon. You can also use the same data domain as you chose for your project. The complexity of your data model should be similar to examples discussed in Lecture 13. It should also satisfy:

- Have at least four tables
- Underline the primary keys
- Draw arrows between columns from foreign key columns to the columns that they refer to.
- Classify each table as either an object, event, or relationship
- Have at least one linking table (having at least two foreign keys)
- List the data type of each column (int, float, text, or epoch time)