

# CMPINF 2110

Spring 2021

Homework 04

Solutions – data models

# The conceptual model captures that we have 3 main entities: Artist, Painting, and Feature

A Painting has 3 attributes: The season and episode the painting was painted in and the title of the painting. You may have called the painting the episode. That was acceptable as long as it was clear what you were referring to.



An artist has one attribute, their name, in our assignment.

A feature has one attribute, its type or name.

# The conceptual model captures the basic relationships between the entities

An artist paints a painting.

There could be many artists that paint many paintings, hence the many-to-many relationship is used.

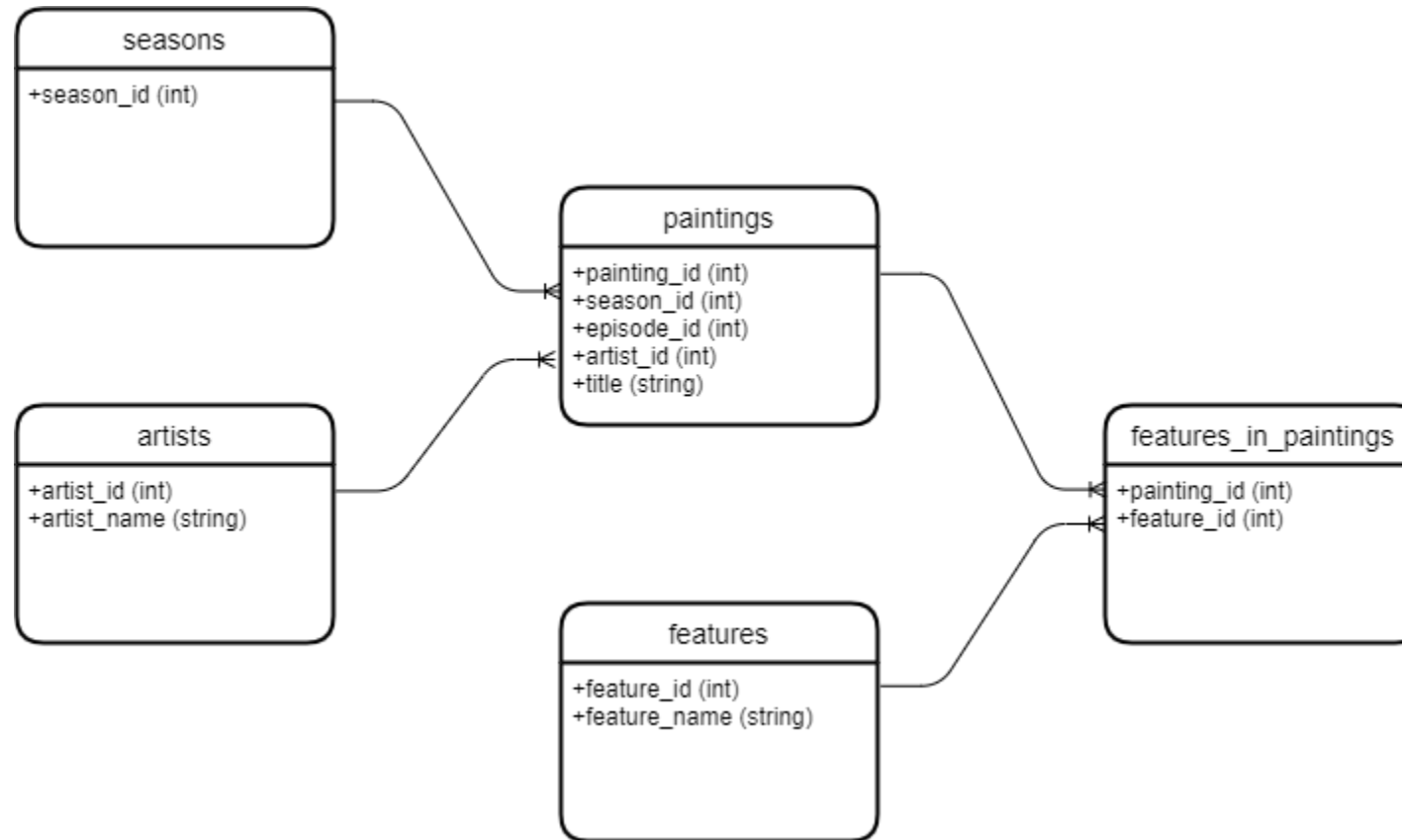


A feature is in a painting.

There could be many features in many paintings, hence the many-to-many relationship is used.

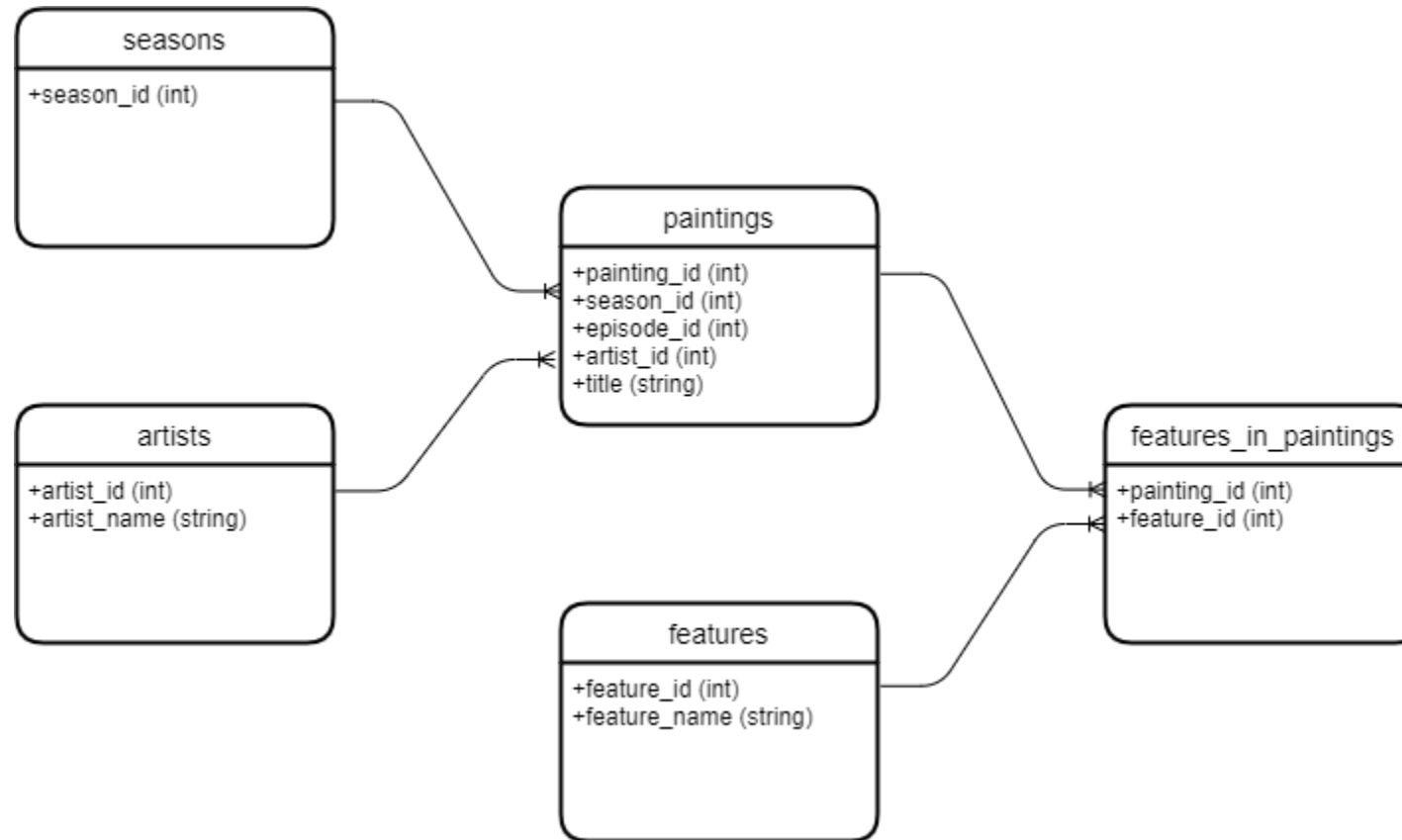
We know which feature an artist paints because know how artist relates to painting and how painting relates to feature.

The logical model expands on the conceptual model, representing how the entities (tables) will be logically related within a relational data model

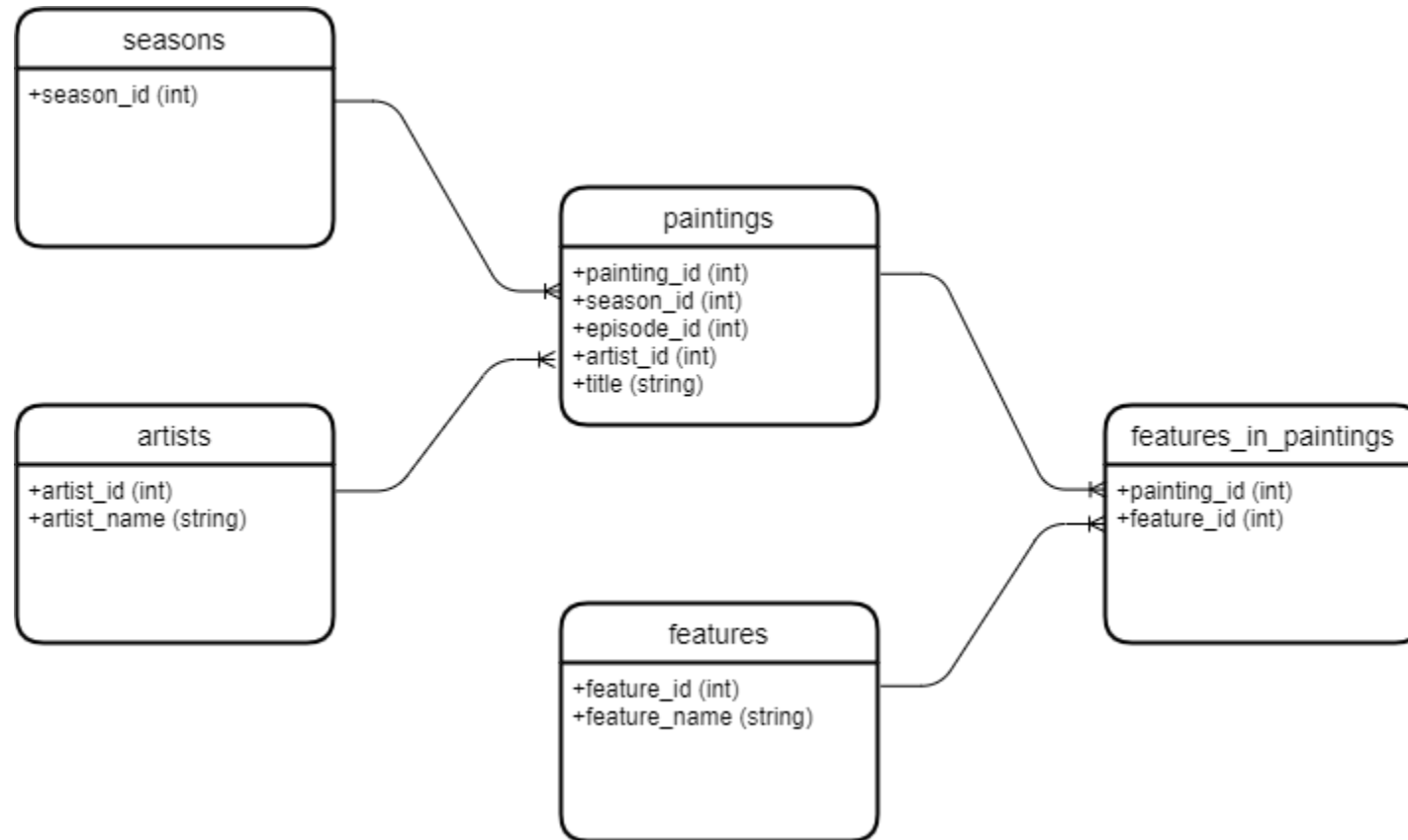


The `seasons` and `artists` tables store attributes associated with each unique Season and Artist

We are only storing the `season_id` in this example. Having a separate `seasons` table allows storing more information if required (such as produced by, filmed at, etc...)

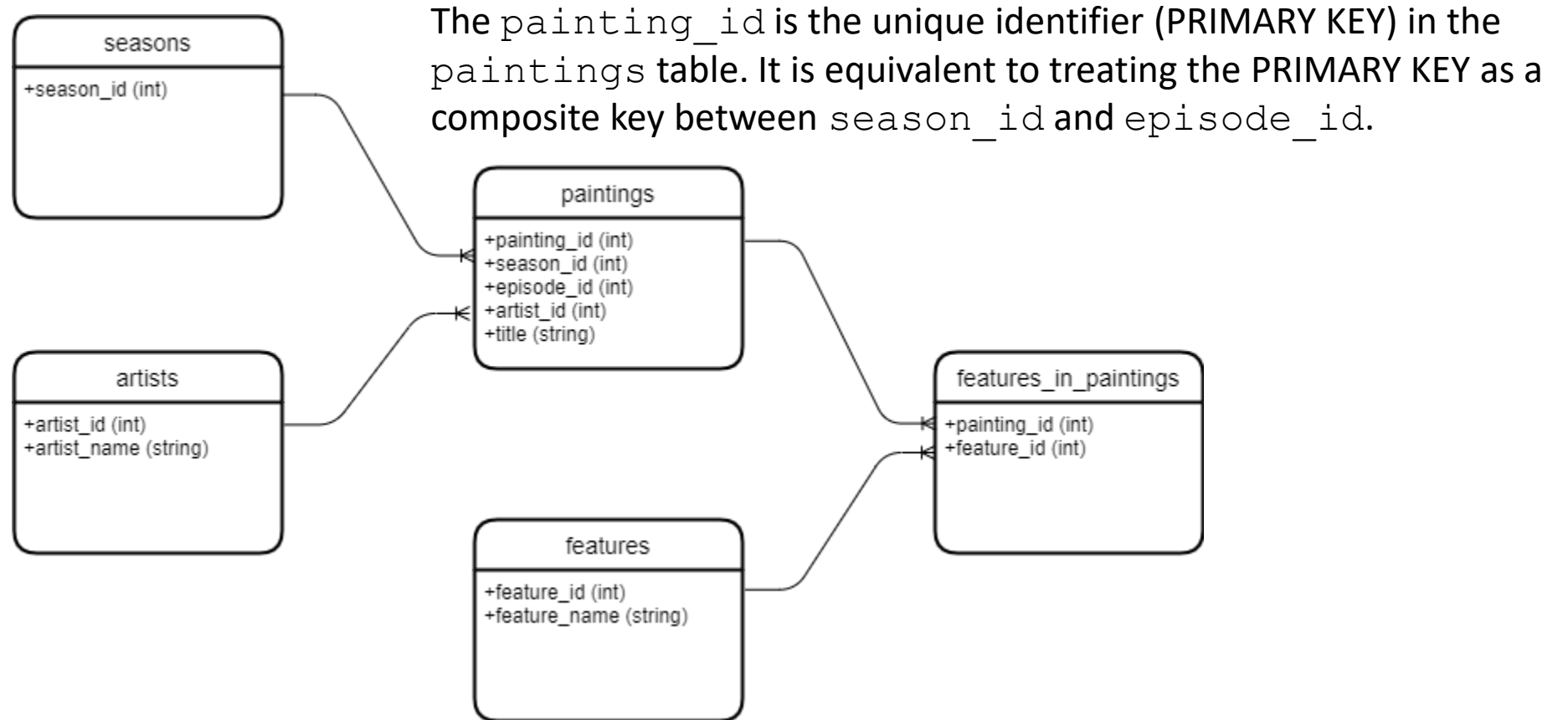


The `seasons` and `artists` tables store attributes associated with each unique Season and Artist

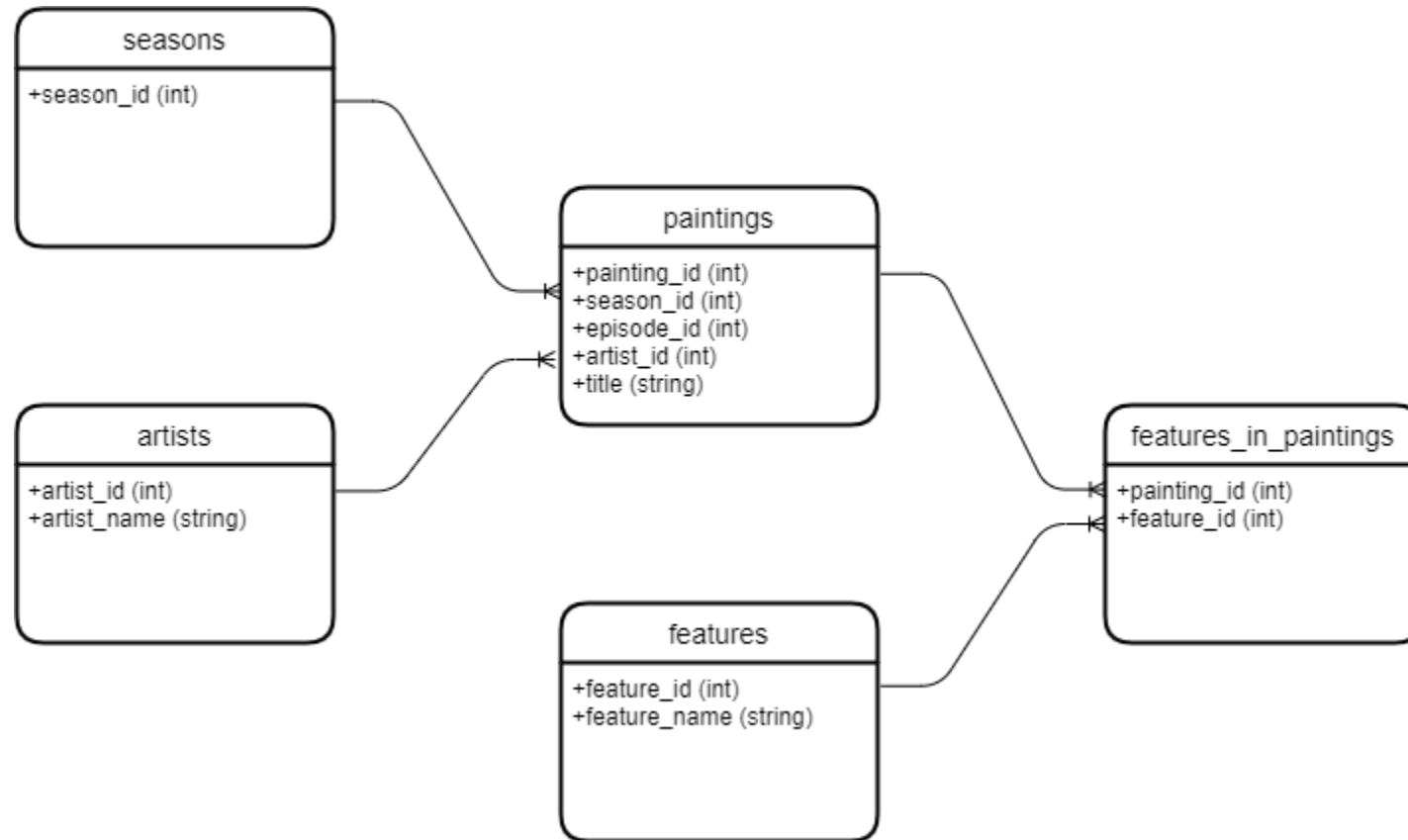


One row in the `artists` table corresponds to a specific person (artist). We are only storing their name in this example.

One row in the `paintings` table corresponds to a specific painting. The `paintings` table can be JOINED to the `artists` table via the `artist_id` column. This tells us who painted the painting in that episode of the show.

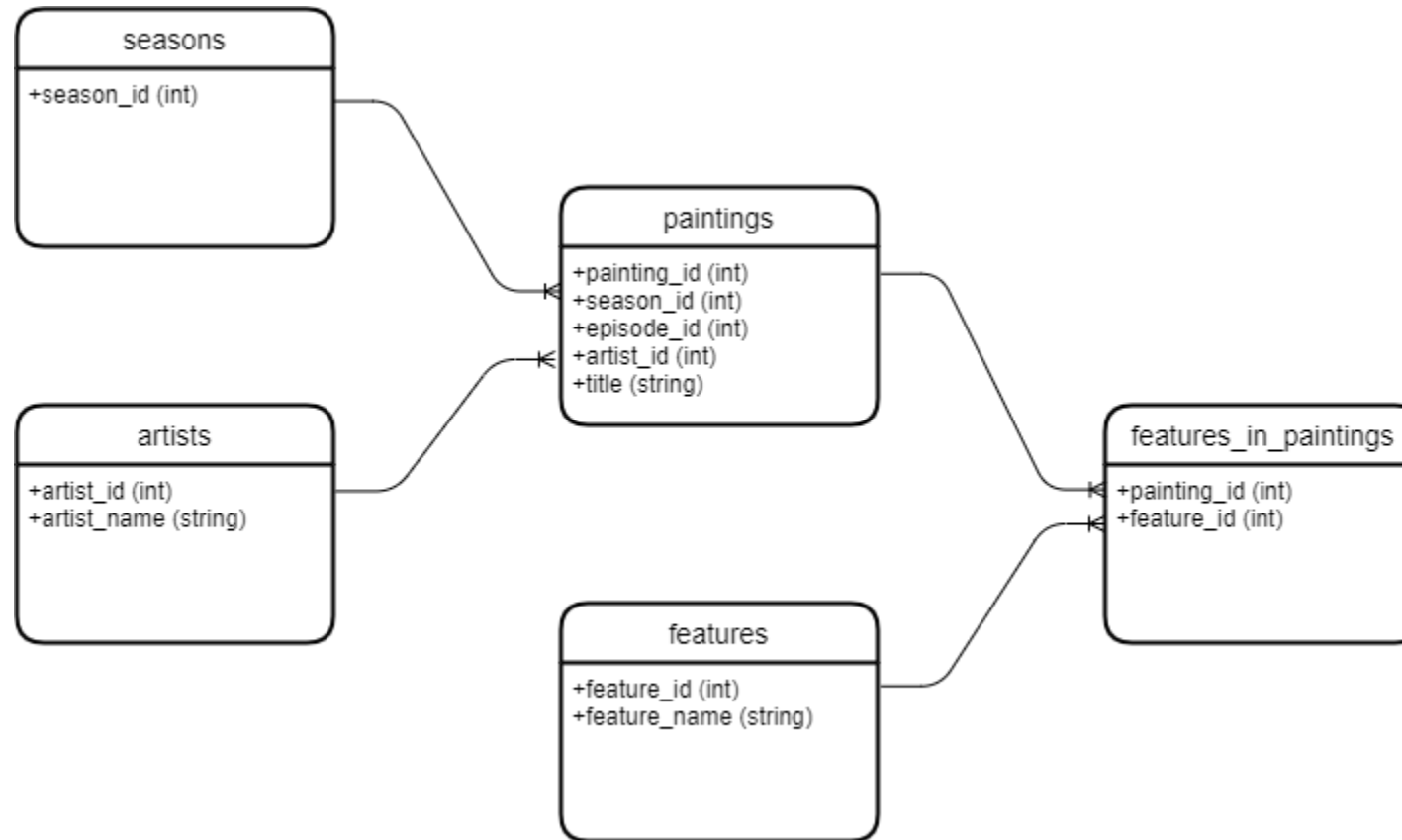


The only attribute for the feature that we are storing is the name. The PRIMARY KEY of the `features` table is the `feature_id` integer.

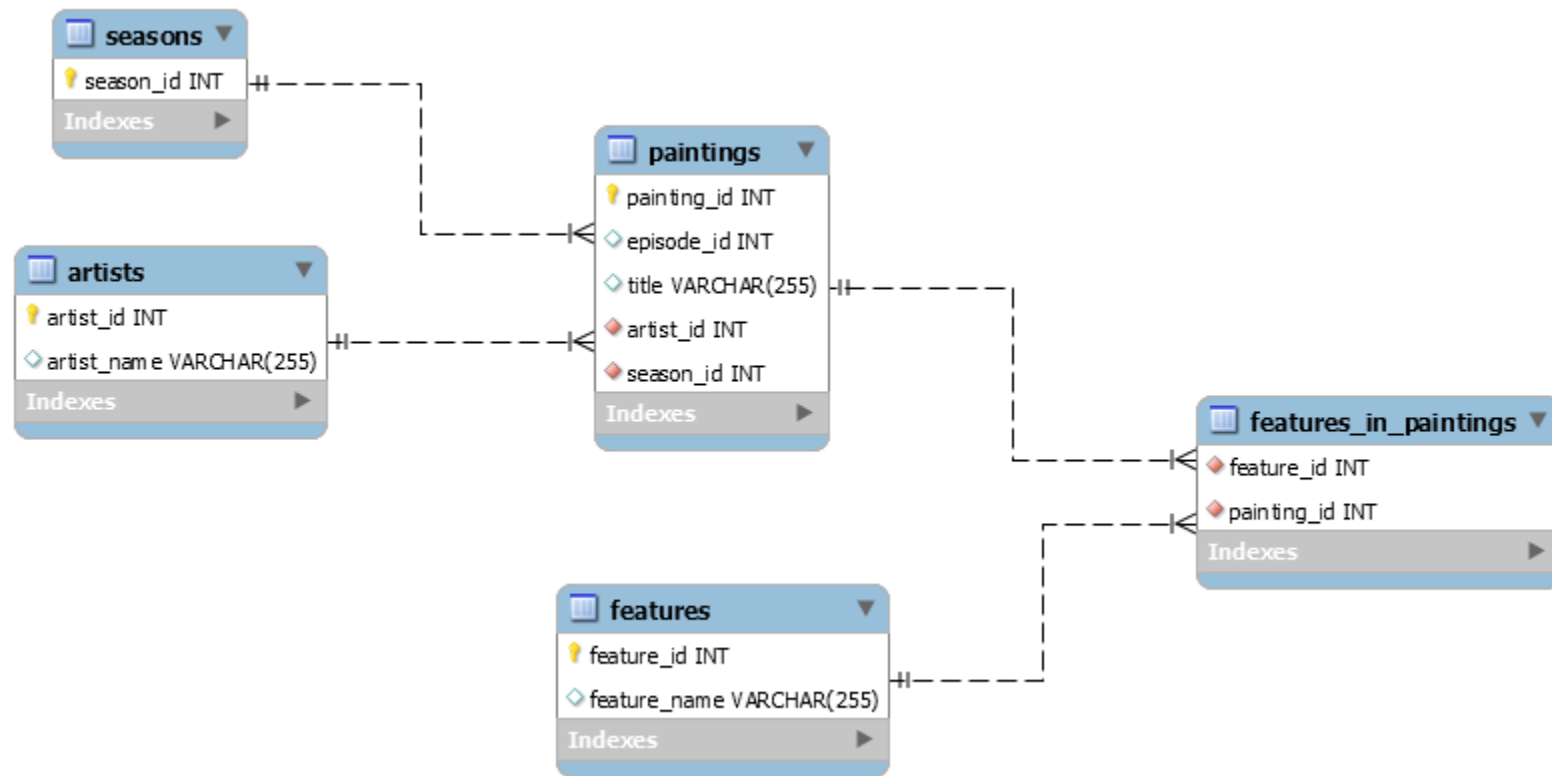




The `features_in_paintings` table is a link table that allows the many-to-many features-in-paintings relationship to be logically modeled



# Physical model – implement the logical model within a relational data base



# Create data base from physical model

- Data model created in MySQL Workbench:
  - File name: `hw04_bob_ross_physical_model.mwb`
- MySQL script auto-generated by the MySQL Workbench Forward Engineering function.
- Script named:  
`hw04_bob_ross_create_db_from_model.sql`

# Tables populated via data import wizard

- Multiple queries are demonstrated in:
  - `hw04_bob_ross_mysql_query.sql`
- Last query shows how to use a subquery to identify all paintings (season + episode) that contained at least 1 mountain.