Gabriel Janzen and Connor Shields
CS340 – step 5
May 27, 2018

No feedback from step 4 yet, so no changes.

Coffee Types Database Project Step 5
Website URL: http://flip2.engr.oregonstate.edu:2419
Fixes based on Feedback from Step 2 and 3 Step 2 feedback from the Grader: The overall structure looks good to proceed. Just a small thought that maybe Variety can also have a N to M relationship? But that's up to you.

Step 2 feedback from the Peer reviewer: A) ERD The ERD matches the description in the database outline. 1. The attributes for each entity in the ERD are the same as what is described in the outline. 2. The participation of entities in the relationships are the same as what is described in the outline. 3. The cardinality of entities in the relationships are the same as what is described in the outline. 4. Notes on improvement: The Roast entity only consists of a Roast ID and the roast Type. This may not need to be its own entity. If a coffee can only be light, medium, or dark (and not more than one of these), then it might be better to include Roast as an attribute along with name and description. B) SCHEMA 1. The relationship tables required for the many-to-many relationships are all present and correctly defined. 2. Foreign Keys are all present and correctly defined. 3. The entity attributes appear to match the outline. 4. As the Roast attribute is in a one-to-many relationship with Coffee (a Coffee only has one roast), and only consists of a roast_id and roast description, it probably does not require its own table. Roast description could be added as a column to the Coffee table.

Step 3 feedback from the Grader: All queries passed, structure looks ok, good to proceed. Step 3 feedback from the Peer reviewer: None.

What I chose to fix and how I fixed it: I did not make any changes. What I chose not to fix and why I decided so: I chose to not change the Roast entity because I believe it provides better organization to keep it separate, and the BSG example database has a similar table. And about Variety being an N to M relationship, it already is has its own relationship table. Other changes that I decided to make on my own: I did not make any other changes.

Project Outline and Database Outline - Updated Version
No new changes have been suggested and the project and database outline remain unchanged since the last step.

Project Outline I will be making a database for coffees from all over the world. There are many different varieties, roasts, and origins, and each roaster can create a unique coffee by changing each of these features. This database will store information on many different coffees in a well-organized and easy-to-understand format.

Database Outline The entities in my database are:
• Coffee – This will be the type of coffee. Coffee names are usually descriptive of the type of coffee used, but it can have a creative name chosen by the roaster as well. Coffee is the main entity in the database and is connected to all of the other entities.

o coffee_id: Each coffee has a unique id that is automatically generated upon creation. This is an auto-incremented number which is the primary key.

o name: Each coffee has a name associated with it. This is a string with a max of 100 characters, which cannot be blank and has no default.

o origin: The country of where the coffee was originally grown. This will contain the id of that origin in the database. It cannot be blank and there is no default.

o roaster: Which Roaster created this coffee. Contains the id of the Roaster, cannot be null, there is no default, and cannot be from a Roaster not in the database.

o roast: This will contain the id of which Roast is used. Cannot be null and must be one of the four Roast options in the database, and there is no default.

o variety: Contains the id of the Variety of the coffee. Must be one of the Varieties in the database and cannot be null. The default is Coffea arabica. o description: A description of what the coffee is like. A string of max 255 characters, and the default is null.

• Origin – The origin is the place that the coffee was grown.

o origin_id: Every origin has a unique id that is automatically generated upon creation. This is an auto-incremented number which is the primary key.

o country: The name of the origin which is a string of max 100 characters. It cannot be null and has no default.

o continent: The name of the continent in which the country belongs to. It is a string of max 100 characters and cannot be blank.

• Roaster – The roaster is the entity that makes each coffee.

o roaster_id: Each roaster has a unique id that is automatically generated upon creation. This is an auto-incremented number which is the primary key.

o name: The name of the roaster, either a business name or an individual. It is a string of max 100 characters and cannot be blank.

o city: The city where the roaster is located. It is a string of max 100 characters and cannot be blank.

o state: The state where the roaster is located. It is a string of 2 characters and cannot be blank.

• Roast – There are light, medium, medium-dark, and dark roasts.

o roast_id: Each roast has a unique id that is automatically generated upon creation. This is an auto-incremented number which is the primary key.

o roast: The roasting type of the coffee. A string of max 11 characters and must be either light, medium, medium-dark, or dark.

• Variety – This entity holds the different varieties of coffee plants, such as Coffea arabica.

o variety_id: Each variety has a unique id that is automatically generated upon creation. This is an auto-incremented number which is the primary key.

o name: The name of each coffee variety. Cannot be blank and must be a string of max 100 characters.

o description: Details about the coffee variety. A string of max 255 characters, and the default is null.

The relationships in my database are:

• Coffees are from origins – This is a many-to-many relationship as coffees can be blended and come from one or more origins, and many different coffees can come from each origin.

• Coffees are made by a roaster – This is a one-to-many relationship. Each coffee comes from one roaster, and each roaster can have many different coffees.

• Coffees have a roast level – This is a one-to-many relationship. Each coffee must have one of four available roasts, and many coffees can have the same roast.

• Coffees are a type of variety – This is a many-to-many relationship as coffees can be blended and consist of one or more varieties, and many different coffees can come from each variety.

See schema and outline files for the ER diagram, Schema, and Data Definition queries.