```
/*----------------------------------------------------------------------
 * Name: Anna Smith and Jalen Tacsiat
 * File: proj5.sql
 * Date: 10/22/2020
 * Class: CPSC 321 - Databases
 * Description: Create and insert statements for our recipes database
 ----------------------------------------------------------------------*/

-- required in MariaDB to enforce constraints
SET sql_mode = STRICT_ALL_TABLES;
```

10/10

```
DROP TABLE IF EXISTS RecipesInMeals;
DROP TABLE IF EXISTS RecipeHasDietaryRestrictions;
DROP TABLE IF EXISTS CookingToolsRequired;
DROP TABLE IF EXISTS IngredientOf;
DROP TABLE IF EXISTS Instruction;
DROP TABLE IF EXISTS Ingredient;
DROP TABLE IF EXISTS Rating;
DROP TABLE IF EXISTS Recipe;
DROP TABLE IF EXISTS Meal;
DROP TABLE IF EXISTS User;
DROP TABLE IF EXISTS CookingTool;
DROP TABLE IF EXISTS DietaryRestriction;

-- #############################################################################
-- CREATE TABLE STATEMENTS
-- #############################################################################

CREATE TABLE DietaryRestriction(
  restriction_id INT UNSIGNED AUTO_INCREMENT,
  restriction_name VARCHAR(50) UNIQUE NOT NULL,
  PRIMARY KEY (restriction_id)
);

CREATE TABLE CookingTool(
  tool_id INT UNSIGNED AUTO_INCREMENT,
  tool_name VARCHAR(50) UNIQUE NOT NULL,
  PRIMARY KEY (tool_id)
);

CREATE TABLE User(
  username VARCHAR(50) CHECK (LENGTHB(password) >= 5),
  name VARCHAR(50) NOT NULL,
  password VARCHAR(50) NOT NULL CHECK (LENGTHB(password) >= 8),
  PRIMARY KEY (username)
);

CREATE TABLE Meal(
  meal_id INT UNSIGNED AUTO_INCREMENT,
  meal_name VARCHAR(50) NOT NULL,
  description VARCHAR(100),
  username VARCHAR(50),
  PRIMARY KEY (meal_id),
  FOREIGN KEY (username) REFERENCES User(username) ON DELETE SET NULL
);

CREATE TABLE Recipe(
  recipe_id INT UNSIGNED AUTO_INCREMENT,
  food_type VARCHAR(30),
  cuisine_type VARCHAR(30),
  recipe_title VARCHAR(30) NOT NULL,
  username VARCHAR(50),
  PRIMARY KEY (recipe_id),
  FOREIGN KEY (username) REFERENCES User(username) ON DELETE SET NULL
);

CREATE TABLE Rating(
  rating_id INT UNSIGNED AUTO_INCREMENT,
  score SMALLINT UNSIGNED NOT NULL CHECK (score <= 5),
  date_added DATE,
  username VARCHAR(50),
  recipe_id INT UNSIGNED NOT NULL,
  PRIMARY KEY (rating_id),
  FOREIGN KEY (username) REFERENCES User(username) ON DELETE SET NULL,
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);

CREATE TABLE Ingredient(
  ingredient_id INT UNSIGNED AUTO_INCREMENT,
  ingredient_name VARCHAR(30) UNIQUE NOT NULL,
  recipe_id INT UNSIGNED,
  PRIMARY KEY (ingredient_id),
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE SET NULL
);

CREATE TABLE Instruction(
  recipe_id INT UNSIGNED,
  step_number SMALLINT UNSIGNED,
  description VARCHAR(800) NOT NULL,
  PRIMARY KEY (recipe_id, step_number),
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);

CREATE TABLE IngredientOf(
  recipe_id INT UNSIGNED,
  ingredient_id INT UNSIGNED,
  amount DECIMAL(5, 2) UNSIGNED NOT NULL,
  measurement_units VARCHAR(20),
  PRIMARY KEY (recipe_id, ingredient_id),
  FOREIGN KEY (ingredient_id) REFERENCES Ingredient(ingredient_id) ON DELETE CAS
CADE,
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);

CREATE TABLE CookingToolsRequired(
  tool_id INT UNSIGNED,
  recipe_id INT UNSIGNED,
  PRIMARY KEY (recipe_id, tool_id),
  FOREIGN KEY (tool_id) REFERENCES CookingTool(tool_id) ON DELETE CASCADE,
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);

CREATE TABLE RecipeHasDietaryRestrictions(
  recipe_id INT UNSIGNED,
  restriction_id INT UNSIGNED,
  PRIMARY KEY (recipe_id, restriction_id),
  FOREIGN KEY (restriction_id) REFERENCES DietaryRestriction(restriction_id) ON
DELETE CASCADE,
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);
```

```
CREATE TABLE RecipesInMeals(
  recipe_id INT UNSIGNED,
  meal_id INT UNSIGNED,
  PRIMARY KEY (recipe_id, meal_id),
  FOREIGN KEY (meal_id) REFERENCES Meal(meal_id) ON DELETE CASCADE,
  FOREIGN KEY (recipe_id) REFERENCES Recipe(recipe_id) ON DELETE CASCADE
);

-- #############################################################################
-- Insert Statements
-- #############################################################################

INSERT INTO DietaryRestriction (restriction_name)
  VALUES ("vegan"),
          ("vegetarian"),
          ("lactose intolerant"),
          ("gluten free");

INSERT INTO CookingTool (tool_name)
  VALUES ("pot"),
          ("spatula"),
          ("oven"),
          ("stovetop"),
          ("grill"),
          ("blender");

INSERT INTO User (username, name, password)
  VALUES ("gordonramsay123", "gordon ramsay", "hellskitchen"),
          ("altonbrown_1", "alton brown", "iLik3toCook"),
          ("guyfieri420", "guy fieri", "triple_D!"),
          ("spongebob2", "spongebob squarepants", "krustykrab"),
          ("rat_atouille", "remy the rat", "baguette123");

INSERT INTO Recipe (food_type, cuisine_type, recipe_title, username)
  VALUES ("pizza", "italian", "pizza margherita on focaccia", "gordonramsay123")
,
          ("bread", "italian", "focaccia bread", "gordonramsay123"),
          ("pizza", "italian", "pepperoni pizza", "guyfieri420"),
          ("burger", "america", "cheeseburger", "spongebob2"),
          ("soup", "japanese", "miso soup", "altonbrown_1"),
          ("sauce", "japanese", "teriyaki sauce", "spongebob2"),
          ("ratatouille", "french", "ratatouille", "rat_atouille"),
          ("soup", "french", "provencial greens soup", "rat_atouille");

INSERT INTO Ingredient (ingredient_name, recipe_id)
  VALUES ("focaccia bread", NULL),
          ("marinara sauce", NULL),
          ("cheese", NULL),
          ("tomatoes", NULL),
          ("basil", NULL),
          ("balsamic glaze", NULL),
          ("flour", NULL),
          ("salt", NULL),
          ("yeast", NULL),
          ("water", NULL),
          ("olive oil", NULL),
          ("vegetable broth", NULL),
          ("nori", NULL),
          ("leeks", NULL),
          ("scallions", NULL),
          ("tofu", NULL),
          ("miso", NULL),
          ("pepperoni", NULL),
          ("dough", NULL),
          ("burger meat", NULL),
          ("buns", NULL),
          ("condiments", NULL),
          ("lettuce", NULL),
          ("soy sauce", NULL),
          ("rice vinegar", NULL),
          ("brown sugar", NULL),
          ("sesame oil", NULL),
          ("garlic", NULL),
          ("ginger", NULL),
          ("cornstarch", NULL),
          ("chard", NULL),
          ("pepper", NULL),
          ("lemon", NULL),
          ("egg", NULL),
          ("parmesean", NULL),
          ("eggplants", NULL),
          ("squash", NULL),
          ("zucchini", NULL),
          ("parsley", NULL),
          ("phyme", NULL);

INSERT INTO Rating (score, date_added, username, recipe_id)
  VALUES (5, '2015-04-03', "gordonramsay123", 4),
          (1, '2019-02-05', "gordonramsay123", 3),
          (4, '2017-06-12', "altonbrown_1", 2),
          (3, '2018-07-11', "guyfieri420", 2),
          (5, '2019-01-10', "spongebob2", 5),
          (5, '2020-01-04', "spongebob2", 1),
          (5, '2017-03-03', "rat_atouille", 6),
          (3, '2014-04-01', "rat_atouille", 3),
          (4, '2020-12-04', "altonbrown_1", 6),
          (2, '2020-12-04', "altonbrown_1", 3),
          (3, '2020-12-04', "altonbrown_1", 1),
          (5, '2020-12-04', "spongebob2", 7),
          (5, '2020-12-04', "spongebob2", 8),
          (5, '2020-12-04', "gordonramsay123", 7),
          (5, '2020-12-04', "guyfieri420", 8),
          (5, '2020-12-04', "altonbrown_1", 7),
          (5, '2020-12-04', "altonbrown_1", 8);

INSERT INTO IngredientOf (recipe_id, ingredient_id, amount, measurement_units)
  VALUES (2, 7, 3.25, "cups"),
          (2, 8, 1, "tablespoon"),
          (2, 9, .5, "teaspoons"),
          (2, 10, 1.75, "cups"),
          (2, 11, 4, "tablespoons");

INSERT INTO IngredientOf (recipe_id, ingredient_id, amount, measurement_units)
  VALUES (1, 1, 1, NULL),
          (1, 2, .25, "cups"),
          (1, 3, 6, "oz"),
          (1, 4, 1, "cup"),
          (1, 5, 1, "oz"),
          (1, 6, 2, "tablespoons");
```

```
INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (3, 2, 5, "cups"),
          (3, 3, 3, "oz"),
          (3, 18, 20, "slices");

INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (4, 22, 5, "cups"),
          (4, 20, 3, "oz"),
          (4, 3, 2, "slices");

-- miso soup ingredients
INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (5, 12, 6, "cups"),
          (5, 13, 1, "sheet"),
          (5, 14, 1, NULL),
          (5, 15, 3, NULL),
          (5, 16, 8, "oz"),
          (5, 17, 2, "tablespoons"),
          (5, 24, 1, "tablespoon"),
          (5, 25, 1.5, "tablespoons");

-- teriyaki sauce ingredients
INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (6, 24, .5, "cup"),
          (6, 10, .25, "cup"),
          (6, 25, 2, "tablespoons"),
          (6, 26, .25, "cup"),
          (6, 27, 1, "teaspoon"),
          (6, 28, 3, "cloves"),
          (6, 29, 1, "tablespoon"),
          (6, 30, 1, "tablespoon");

-- provencial greens soup ingredients
INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (7, 11, 2, "tablespoons"),
          (7, 14, 2, NULL),
          (7, 28, 4, "cloves"),
          (7, 8, 2, "teaspoons"),
          (7, 31, 6, "cups"),
          (7, 32, .5, "teaspoon"),
          (7, 10, 6, "cups"),
          (7, 33, 1, NULL),
          (7, 34, 2, NULL),
          (7, 35, .25, "cups");

-- ratatouille ingredients
INSERT INTO IngredientOf(recipe_id, ingredient_id, amount, measurement_units)
  VALUES (8, 36, 1, NULL),
          (8, 4, 5, NULL),
          (8, 37, 2, NULL),
          (8, 38, 2, NULL),
          (8, 2, 3, "cups"),
          (8, 28, 1, "cloves"),
          (8, 5, .25, "cup"),
          (8, 39, 2, "tablespoons"),
          (8, 40, .25, "cup");

INSERT INTO Instruction (recipe_id, step_number, description)
  VALUES (2, 1, "Whisk together the flour, kosher salt and yeast. Add the warm w
ater to the flour mixture and stir until incorporated."),
          (2, 2, "Pour 2 tablespoons oil into a medium bowl. Transfer the dough
to the bowl, cover with plastic wrap."),
          (2, 3, "Place in the refrigerator to rest for 24 hours."),
          (2, 4, "Brush the inside of a 9-by-13-inch baking sheet with oil. Remo
ve the dough from the refrigerator and transfer to the prepared pan."),
          (2, 5, "Using your hands, spread the dough out as much as possible, ad
ding oil to the dough if needed to keep it from sticking."),
          (2, 6, "Place the dough in a warm place and let rise until about doubl
ed in size. Then, spread out on sheet."),
          (2, 7, "Heat the oven to 450 degrees. Pat down the focaccia to an even
 thickness of about 1 inch."),
          (2, 8, "Drizzle it with the remaining 2 tablespoons olive oil. Sprinkl
e with salt."),
          (2, 9, "Bake, rotating once front to back, until the top is uniformly 
golden brown, 20 to 25 minutes. Cool for 20 minutes");

INSERT INTO Instruction (recipe_id, step_number, description)
  VALUES (1, 1, "Preheat oven to 375."),
          (1, 2, "Spread focaccia bread with pizza sauce, and top with mozzarell
a and tomatoes."),
          (1, 3, "Cook until cheese melts.  Broil for a minute or so until the c
heese is just starting to char on the tops of bubbles."),
          (1, 4, "Top with basil and balsamic glaze.");

INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (3, 1, "Set up Buns"),
          (3, 2, "Put condiment here"),
          (3, 3, "Put burger on bun"),
          (3, 4, "Close bun");

INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (4, 1, "Set up Dough"),
          (4, 2, "Put sauce on dough"),
          (4, 3, "Put Pepperoni on dough"),
          (4, 4, "Put cheese on dough"),
          (4, 5, "Bake");

-- miso soup instructions
INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (5, 1, "Heat vegetable broth over medium heat until simmering."),
          (5, 2, "Add nori, leeks, scallions, and tofu. Let simmer 5 more minute
s."),
          (5, 3, "In a small bowl, stir miso paste with just enough water to dil
ute."),
          (5, 4, "Add contents of bowl to broth and let simmer a minute or so mo
re.");

-- teriyaki sauce instructions
INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (6, 1, "In a small saucepan over medium heat, add soy sauce, water, ric
e vinegar, sugar, sesame oil, ginger, and garlic and bring to a simmer."),
          (6, 2, "Reduce heat and let simmer until reduced by about a third, abo
ut 10 minutes."),
          (6, 3, "In a small bowl, whisk together cornstarch slurry. Pour into s
auce slowly while continuously whisking."),
          (6, 4, "Cook for a minute or so more until it thickens just a little b
it more.");

-- provencial greens soup instructions
```

```
INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (7, 1, "Heat olive oil in a large sauce pot over medium heat."),
          (7, 2, "Add the leeks and cook, until tender, 3 to 5 minutes."),
          (7, 3, "Add the garlic and salt, and cook until the garlic is fragrant
, about 1 minute."),
          (7, 4, "Add greens and pepper, and cook until they begin to wilt."),
          (7, 5, "Add water, lemon juice and parmesan rind. Reduce heat and simm
er, partially covered, for 15 to 20 minutes."),
          (7, 6, "In a separate bowl, whisk eggs and stir in a ladle of soup bro
th. Slowly add mixture back into the soup while stirring."),
          (7, 7, "Serve topped with parmesan shavings.");

-- ratatouille instructions
INSERT INTO Instruction(recipe_id, step_number, description)
  VALUES (8, 1, "Preheat oven to 375 °F"),
          (8, 2, "Add the leeks and cook, until tender, 3 to 5 minutes."),
          (8, 3, "Add the garlic and salt, and cook until the garlic is fragrant
, about 1 minute."),
          (8, 4, "Add greens and pepper, and cook until they begin to wilt."),
          (8, 5, "Add water, lemon juice and parmesan rind. Reduce heat and simm
er, partially covered, for 15 to 20 minutes."),
          (8, 6, "In a separate bowl, whisk eggs and stir in a ladle of soup bro
th. Slowly add mixture back into the soup while stirring.");

INSERT INTO CookingToolsRequired(tool_id, recipe_id)
  VALUES (5, 4),
          (2, 4),
          (3, 3),
          (3, 2),
          (1, 5),
          (3, 1),
          (6, 1),
          (6, 2);

INSERT INTO RecipeHasDietaryRestrictions (recipe_id, restriction_id)
  VALUES(2, 2),
        (1, 2),
        (5, 1),
        (5, 2),
        (6, 1),
        (6, 2);

INSERT INTO Meal(meal_name, description, username)
  VALUES ("Burgers and pizza", "A burger and a pizza", "gordonramsay123"),
          ("Pizza and Soup", "A pizza and some Japanese Miso Soup", "guyfieri420
"),
          ("Burgers with Focaccia Bread", "Burgers using Focaccia Bread", "spong
ebob2");

INSERT INTO RecipesInMeals (recipe_id, meal_id)
  VALUES (3, 1),
          (4, 1),
          (1, 2),
          (5, 2),
          (4, 3),
          (2, 3);

-- ################################################################################
-- QUERIES
-- ################################################################################
-- 1) Most basic search for recipes, this is the default list that is displayed
-- selects all recipe_id, recipe name, user's name, avg(review)
SELECT re.recipe_id, re.recipe_title, re.username, AVG(ra.score)
FROM Recipe re JOIN Rating ra USING (recipe_id)
GROUP BY re.recipe_id
ORDER BY AVG(ra.score);

-- ##########################################################################
-- 2) Queries to get recipe details to populate single recipe page in the GUI.
--    Recipe pages display list of instructions, list of ingredients,
--    list of dietary restrictions, list of required cooking tools,

-- 2.1) select step number and desciption of all instructions corresponding to a
--      specific recipe_id
SELECT i.step_number, i.description
FROM Instruction i JOIN Recipe re USING (recipe_id)
WHERE re.recipe_id = 4;

-- 2.2) select ingredient name, amount, and measurement unit, recipe_id for each
 ingredient used
-- in a specific recipe
SELECT i.ingredient_name, ig.amount, ig.measurement_units
FROM IngredientOf ig JOIN Ingredient i using (ingredient_id)
WHERE ig.recipe_id = 1;

-- 2.3) select restriction name that correspond to recipe_id
SELECT d.restriction_name
FROM RecipeHasDietaryRestrictions rd JOIN DietaryRestriction d USING (restrictio
n_id)
WHERE rd.recipe_id = 1;

-- 2.4) select cooking tool names that correspond to recipe_id
SELECT c.tool_name
FROM CookingToolsRequired cr JOIN CookingTool c
WHERE cr.recipe_id = 4;

-- 2.5) selects title, cuisine type, food type, avg rating, user's name
SELECT re.recipe_title, re.cuisine_type, re.food_type, AVG(ra.score), re.usernam
e
FROM Recipe re JOIN Rating ra USING (recipe_id)
WHERE recipe_id = 1
ORDER BY AVG(ra.score);

-- ##########################################################################
-- 3) All queries below will be used in our search for recipes, these are variou
s
-- options a user can use to search by. Some of these will later be incorporated
-- into sub-queries

-- 3.1) selects all usernames where username is in specified list
-- replace ("gordonramsay123", "Alton Brown") with variable holding list of name
s
-- this will allow user to put in a list of users they want to find recipes of,
-- where they can either use the user name or actual name to search
SELECT u1.username, r1.recipe_title, r1.recipe_id
FROM User u1, Recipe r1
WHERE u1.username IN ("gordonramsay123", "Alton Brown") OR name IN ("gordonramsa
y123", "Alton Brown");
```

```
-- 3.2) selects users with an average rating higher than a certain number (in this case, 4)
SELECT u1.username
FROM User u1, Recipe r1, Rating r2
WHERE u1.username = r1.username AND r1.recipe_id = r2.recipe_id
GROUP BY u1.username
  HAVING AVG(r2.score) >= 4;

-- selects users who have made at least a certan number of
-- recipes (in this case, at least 2 recipes)
SELECT u1.username
FROM User u1 JOIN Recipe r1 USING (username)
GROUP BY u1.username
  HAVING COUNT(r1.recipe_id) >= 2;

-- selects ingredient IDs given a list of keywords
SELECT ingredient_id, ingredient_name
FROM Ingredient i
WHERE ingredient_name IN ("Flour", "Salt", "Yeast");

-- selects most common ingredients for some cuisine type
-- (in this case, selects top 3 ingredients used in Japanese Cuisine)
SELECT COUNT(i1.ingredient_id), i2.ingredient_name
FROM IngredientOf i1 JOIN Ingredient i2 ON (i1.ingredient_id = i2.ingredient_id)
  JOIN Recipe r ON (r.recipe_id = i1.recipe_id)
WHERE r.cuisine_type = "Japanese"
GROUP BY i1.ingredient_id
ORDER BY COUNT(i1.ingredient_id) DESC
LIMIT 3;

-- selects recipes that have less than a certain number of ingredients
-- in this case, selects recipes with less than or equal to 5 ingredients
SELECT recipe_id, recipe_title, COUNT(i.ingredient_id) AS "Amount of Ingredients"
FROM Recipe r JOIN IngredientOf i USING (recipe_id)
GROUP BY i.recipe_id
  HAVING COUNT(i.ingredient_id) <= 5;

-- selects recipes with an average rating higher than some number, in this case,
-- rating must be higher than 4 stars
SELECT r.recipe_title AS "Recipe", AVG(ra.score) AS "Rating"
FROM Recipe r JOIN Rating ra USING (recipe_id)
GROUP BY ra.recipe_id
  HAVING AVG(ra.score) > 4
ORDER BY AVG(ra.score)

-- #############################################################################
-- 4) Queries for meals


-- 3) query for list of meals
-- select name of meal for specific username (current user)
SELECT m.meal_name
FROM Meal m JOIN User u USING (username)
WHERE username = "gordonramsay123";
-- 4) query to get more info on meals

-- 4.1) select meal name, description, user's name given a meal_id
SELECT meal_name, description, username
FROM Meal
```

```
WHERE meal_id = 1;


-- 4.2) select all recipes corresponding to specific meal
SELECT *
FROM Recipe r JOIN RecipesInMeals re USING (recipe_id)
WHERE meal_id = 1;
```