recipe creating/sharing and grocery list app

- users can sign into the app with their email and password
- users can create recipes with ingredients and instructions
- recipes can be marked as public or private
- users can view other people's recipes
- ingredients from recipes can be added to user's grocery lists
- users can create their own special events and assign recipes to special events

Brainstorming & Table Ideas

- User table w/ user_id, email, password,
- Recipe table ref to user_id (view other's recipes)
  - Recipe_id, instructions, public BOOL
- Ingredients table ref recipe_id
  - ingredient_id, name, amount
- Groceries list ref user_id, ref ingredient_id
  - Grocery_id, grocery, price, quantity
- Special events ref user_id, ref recipe_id
  - Event_id, name, date, attendees

Relationships

- Users (one-to-one)
  - Does not refer to any other table
- Recipes (one-to-many)
  - Refers to one table
- Ingredients (one-to-many)
  - Refers to one table
  - Recipes have many ingredients/ingredients many recipes
- Groceries (many-to-many)
  - Refers to multiple tables
  - Users have many groceries/groceries many users
  - Ingredients can have many groceries/groceries many ingredients
- Special Events (many-to-many)
  - Refers to multiple tables
  - Users can have many events/events many users
  - Recipes can have many events/events many recipes

Part 2 / Step 2

- Users:
  - user_id => primary identifier of table
  - Email => varchar(40) => emails vary in length and characters
  - Password => text => hashed pass varies in length
- Recipes:
  - Recipe_id => primary identifier of table
  - User_id => foreign key link to user table
  - Instructions => text => instructions vary largely in length
  - Public_display => boolean => public/private or false/true
- Ingredients:
  - Ingredient_id => primary identifier of table
  - Ingredient_name => varchar(55) => text
  - Amount => varchar(30) => not int because unit of measure will be included
- Grocery_Lists
  - List_id => primary identifier of table
  - User_id => foreign key link to user table
  - Grocery_name => varchar(65) => text
  - Dollar_price => integer => in dollars
  - Quantity => integer => must buy a whole of product
- Recipes_Ingredients:
  - Recipe_ingredient_id=> primary identifier of table
  - Recipe_id => foreign key link recipe
  - Ingredient_id => foreign key link ingredient
- List_Ingredients:
  - List_ingredient_id => primary identifier of table
  - List_id => foreign key link to list table
  - Ingredient_id => foreign key link to ingredient table
- Special Events:
  - Event_id => primary identifier of table
  - User_id => foreign key link user
  - Recipe_id => foreign key link recipe
  - Event_name => varchar(65) => text
  - Date => date => will prompt to enter YYYY-MM-DD
  - Attendees => integer => whole number guests

Part 3 - SQL Sandbox Input Code

```
CREATE TABLE users(
  user_id SERIAL PRIMARY KEY,
  email VARCHAR(40),
  password TEXT
);

CREATE TABLE recipes(
  recipe_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  instructions TEXT,
  public_display BOOLEAN
);

CREATE TABLE ingredients(
  ingredient_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  ingredient_name VARCHAR(55),
amount VARCHAR(30)
);

CREATE TABLE grocery_lists(
  list_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  grocery_name VARCHAR(65),
        dollar_price FLOAT,
        quantity INTEGER
);

CREATE TABLE recipes_ingredients(
  recipe_ingredient_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  recipe_id INTEGER REFERENCES recipes(recipe_id),
  ingredient_id INTEGER REFERENCES
ingredients(ingredient_id)
  );

CREATE TABLE list_ingredients(
  list_ingredient_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  list_id INTEGER REFERENCES recipes(recipe_id),
```

```
  ingredient_id INTEGER REFERENCES
ingredients(ingredient_id)
  );

CREATE TABLE special_events(
  event_id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(user_id),
  recipe_id INTEGER REFERENCES recipes(recipe_id),
  event_name VARCHAR(65),
        date DATE,
        attendees INTEGER
  );
```