Data to consider

Username
Password
Ingredients
Instructions
Recipes
Recipe visibility (public/private)
Grocery list
Occasion

Table Ideas

Login

- id
- Email
- Password

Recipes

- Id
- Instructions_id (fk)
- Quantity
- Login_id (fk)
- recipe _visibility (can be null)
- occasion

Ingredients

- Id
- Food

Recipes_ingredients

- Id
- Recipes_id
- Ingredients_id

Grocery_List

- id
- recipe_ingredients_id (fk)
- Login_id (fk)

Instructions

- Id
- instruction

Occasion
- Id
- occasion

Relationships

1. One-to-one

2. One-to-many
    - Login
    - Instructions
    - occasion

3. Many-to-many
    - Recipes
    - Ingredients
    - Grocery_list
    - recipes_ingredients

Columns

Ingredients
1. The string specifies what food is required in other tables.
2. Varchar because it's a text of food.

Recipes_ingredients
1. Pulls the recipes and ingredients from separate tables.
2. Integer because they are all id's. This is an association table.

Instructions
1. This stores the instructions for recipes
2. Varchar because it's a text of the instructions.

Recipes
1. This stores the instruction id, quantity, connects to the login id, occasion, and if the recipe is public or private.
2. The recipe visibility is boolean because it is set up as true or false for public or private or null. The others are integers connecting to other id's.

Grocery list
1. This table lists all the ingredients from the recipe ingredients list.
2. The data types are all integers, pulling from other table id.

Occasion

1. This table is a list of occasions.
2. Varchar because it's a text of the occasions.

Login

1. Columns in this have the email and password and id.
2. Varchar because it's the text of the user login info.

SQL

```
CREATE TABLE login(
  id SERIAL PRIMARY KEY,
  email VARCHAR(32) NOT NULL,
  password VARCHAR(30) NOT NULL
 );

CREATE TABLE occasion(
  id SERIAL PRIMARY KEY,
  occasion VARCHAR(100)
 );

CREATE TABLE ingredients(
    id SERIAL PRIMARY KEY,
    food VARCHAR(200) NOT NULL
    );

 CREATE TABLE instructions(
    id SERIAL PRIMARY KEY,
    instruction VARCHAR(200) NOT NULL
    );

 CREATE TABLE recipes_ingredients(
    id SERIAL PRIMARY KEY,
    recipes_id INT NOT NULL REFERENCES recipes(id),
    ingredients_id INT NOT NULL REFERENCES ingredients(id)
    );

CREATE TABLE recipes(
  id SERIAL PRIMARY KEY,
  instructions_id INT NOT NULL REFERENCES instructions(id),
```

```
 name VARCHAR(100) NOT NULL,
 login_id INT NOT NULL REFERENCES login(id),
 recipe_visibility BOOLEAN,
 occasion INT REFERENCES occasion(id)
 );

CREATE TABLE grocery_list(
  id SERIAL PRIMARY KEY,
  recipes_ingredients_id INT NOT NULL REFERENCES recipes_ingredients(id),
  login_id INT NOT NULL REFERENCES login(id)
  );
```