DATA MODELING - AFTERNOON LAB

BRAINSTORMING:

User Information
Indv. Ingredients
Public Recipes → contains author_id, indv. ingredients , private boolean false
Private Recipes → contains author_id indv. Ingredients, private boolean true
Grocery List → populated by Indv. Ingredients
Occasion List → populated by Public or Private Recipes

TABLES:

- USER TABLE
    - Id
    - First name
    - Last name
    - Email
    - Password hash
- INGREDIENTS TABLE
    - Id
    - Name
- RECIPE TABLE
    - Recipe_id
    - Dish_name
    - Author_id → **user_id**
    - Ingredients → **recipe-ingredients-table**
    - Cooking_instructions
    - Private, boolean set to: true/false
- RECIPE INGREDIENTS TABLE
    - Recipe_ing_id
    - recipe_id → **recipe_table**
    - Ingredients → **ingredients_id**
    - quantity
- GROCERY LIST TABLE
    - Id
    - Author_id → **user_id**
    - Ingredients added → **ingredients_id**
- OCCASION TABLE
    - Occasion_id
    - Occasion_name
    - Author_id → **user_id**
    - Recipe_id → **recipe_id**

RELATIONSHIPS

One to One:
User → Grocery List
One to Many:
User → Recipes
Occasions → Recipes
Many to Many:
Recipes ←→ Ingredients
Grocery List ←→ Ingredients

CODE: (exported from DB DIAGRAM after original code was submitted for diagram construction)

```
CREATE TABLE "users" (
  "user_id" SERIAL PRIMARY KEY,
  "first_name" VARCHAR(25),
  "last_name" VARCHAR(25),
  "user_email" VARCHAR(50),
  "password_hash" VARCHAR(500)
);

CREATE TABLE "ingredients" (
  "ingredient_id" SERIAL PRIMARY KEY,
  "ingredient_name" VARCHAR(50)
);

CREATE TABLE "recipe" (
  "recipe_id" SERIAL PRIMARY KEY,
  "recipe_name" VARCHAR(50),
  "author_id" INT NOT NULL,
  "ingredients_list" INT NOT NULL,
  "cooking_instructions" VARCHAR(2000),
  "private" boolean
);

CREATE TABLE "recipe_ingredients" (
  "recipe_ingredients_id" SERIAL PRIMARY KEY,
  "recipe_id" INT NOT NULL,
  "ingredients" INT NOT NULL,
  "quantity" INT NOT NULL
);
```

```
CREATE TABLE "grocery_list" (
  "grocery_list_id" SERIAL PRIMARY KEY,
  "author_id" INT NOT NULL,
  "item" INT NOT NULL,
  "quantity" INT NOT NULL
);

CREATE TABLE "occasion" (
  "occasion_id" SERIAL PRIMARY KEY,
  "occasion_name" VARCHAR(50),
  "author_id" INT NOT NULL,
  "recipe_id" INT NOT NULL
);

ALTER TABLE "recipe" ADD FOREIGN KEY ("author_id") REFERENCES "users" ("user_id");

ALTER TABLE "recipe" ADD FOREIGN KEY ("ingredients_list") REFERENCES
"recipe_ingredients" ("recipe_ingredients_id");

ALTER TABLE "recipe_ingredients" ADD FOREIGN KEY ("recipe_id") REFERENCES "recipe"
("recipe_id");

ALTER TABLE "recipe_ingredients" ADD FOREIGN KEY ("ingredients") REFERENCES
"ingredients" ("ingredient_id");

ALTER TABLE "grocery_list" ADD FOREIGN KEY ("author_id") REFERENCES "users"
("user_id");

ALTER TABLE "grocery_list" ADD FOREIGN KEY ("item") REFERENCES "ingredients"
("ingredient_id");

ALTER TABLE "occasion" ADD FOREIGN KEY ("author_id") REFERENCES "users" ("user_id");

ALTER TABLE "occasion" ADD FOREIGN KEY ("recipe_id") REFERENCES "recipe"
("recipe_id");
```