Brainstorming

1. Email and password
2. Profile
3. Public Recipes
4. Private Recipes
5. Users
6. Grocery lists
7. Occasion
8. Followers
9. Following
10. Comments

Relationships

One to Many users->private recipes, users-> grocery lists, users-> Following, users->comments

Many To Many:

One to One: User-Profile

Columns:

Users:

1. User Id-Indicates the User
2. Email- sign in method
3. Password- secure way to get in
4. First name- easy identifier
5. Last name- goes hand in hand with first name
6. Profile pic- visual identifier
7. Birthday- ensures user is old enough

Recipes:

1. Recipe ID- specifies recipe
2. Private_recipe_name- gives name to private recipe
3. Private_recipe_time: Specify time to cook
4. Public_recipe_name- gives name to public recipe
5. public_recipe_time: Specify time to cook

Grocery List:

1. Grocery Id-specifies grocery list
2. Ingredients: Shows ingredients for a recipe_id
3. list_name: Identifies individual list

Occasions:

1. Occasion Id: Specifies the occasion
2. Occasions_name: Identifies the occasion by name
3. Occasions_date: specifies the date of the occasion

Following:

1. Following_Id: specifies the follower
2. Following: Integer that shows how many thing users is following
3. Followers: Integer that shows how many followers user has.

Comments:

1. Comment_id: Specifies the comment
2. User_id: Specifies user who made comment
3. Posts_id: Identifies post
4. Comment_body: the comment text

Posts:

1. Post_id:Identifies the post
2. User_id: identifies the user who posted
3. Body: Post text
4. Pics: Post pictures of the recipe

Step 2 Table Ideas

- User Table
- Recipes
- Occasions
- Following
- Comments
- Posts

Postgres SQL code:

```
-- create table users (
--   user_id serial primary key,
--   email VARCHAR,
--   password varchar(500),
--   first_name varchar,
--   last_name varchar,
--   proflie_pic text,
--   birthday date
--   );
-- drop table recipe;

-- create table recipe (
-- recipe_id serial primary key,
-- user_id int not null references users(user_id),
-- recipe_time varchar,
-- is_public boolean default true
-- );

-- create table grocery_list (
--   grocery_id serial primary key,
--   user_id int not null references users(user_id),
--   ingredients varchar,
--   list_name varchar
--   );

-- drop table occasions;

-- create table occasions (
```

```
--   occasions_id serial primary key,
--   user_id int not null references users(user_id),
--   occasion_name int not null references recipe(recipe_id),
--   occasion_date date
--   );

-- create table following (
-- following_id serial primary key,
-- followers int not null references users(user_id),
-- following int not null references users(user_id)
-- );

--   create table posts (
--   post_id serial primary key,
--   user_id int not null references users(user_id),
--   body text,
--   pics varchar
--    );

--   create table comments (
--    comment_id serial primary key,
--    user_id int not null references users(user_id),
--    post_id int not null references posts(post_id),
--    comment_body text
--    );
--   insert into users (
--    email,
--    password,
--    first_name,
```

```
-- last_name,
-- proflie_pic,
-- birthday
-- )
-- values (
--   'j@gmail.com',
--   'fff',
--   'Jake',
--   'Albiston',
--   'https://www.hi.com/photo',
--   '1990-12-13'
-- );

-- select * from users
```