

# CPSC 304 Project Cover Page

Milestone #: 4

Date: Apr 4, 2024

Group Number: 66

Name	Student Number	CS Alias (Userid)	Preferred Email Address
Brendan Yuen	16212987	u2r4d	<a href="mailto:brendanyuen@hotmail.com">brendanyuen@hotmail.com</a>
Zehao Guo	13448915	r2k8i	<a href="mailto:zahaog@student.ubc.ca">zahaog@student.ubc.ca</a>
Roberto Mulliadi	45216298	d4n0y	<a href="mailto:robertomulliadi@gmail.com">robertomulliadi@gmail.com</a>

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

## 1. Repo Link

[https://github.students.cs.ubc.ca/brenyuen/CPSC304\\_Java\\_Project-master/tree/master](https://github.students.cs.ubc.ca/brenyuen/CPSC304_Java_Project-master/tree/master)

Note: for some reason the course repo can't compile on our local ends, so we created this repo instead.

## 2. SQL Script

[https://github.students.cs.ubc.ca/brenyuen/CPSC304\\_Java\\_Project-master/tree/master/src/ca/ubc/cs304/sql/scripts](https://github.students.cs.ubc.ca/brenyuen/CPSC304_Java_Project-master/tree/master/src/ca/ubc/cs304/sql/scripts)

## 3. Project Description

The domain of the application is in the culinary area. It covers entities like recipes, ingredients, and ratings. This database, for recipes for restaurants and individual use provides a platform for managing and sharing recipes among professional and amateur chefs. Users can search, browse, filter and upload recipes based on dietary restrictions, cuisine type, or difficulty level and so on, with several types of queries implemented.

## 4. Schema

### 4.1 Updates

We merged the many to one tables instead of having a table for a single relation and a single entity.

Changed schema:

recipe\_created\_by merges recipe and created\_by  
recipe\_isof  
recipe\_has  
review\_has  
review\_leaves

### 4.2 Lists of Schemas

*MEALTYPE*

University of British Columbia, Vancouver  
Department of Computer Science

	TYPE
1	Beverage
2	Breakfast
3	Dessert
4	Dinner
5	Lunch

RECIPEUSERS

	USERID	USERNAME	PASSWORD	PREFERENCES	EMAIL
1	1	John Smith	supersecretpassword123	Italian cuisine	smith_john@gmail.com
2	2	Ben Simmons	iamtall4019	Healthy eating	bensimmons@gmail.com
3	3	Alexander Bell	imakeTelephonesringring3303	Desserts	bellringer_alex@yahoo.com
4	4	Lionel Messi	iamthegoat	Asian dishes	greatestofall_time@gmail.com
5	5	l	jkjllkjlmk	jlkljnl	gkj@outlook.com



RECIPE\_IS\_OF

	RECIPEID	TITLE	INSTRUCTIONS	COOKINGTIME	SERVINGSIZE	TYPE
1	1	Best Buttermilk Biscuits	insert instruction	50	4	Breakfast
2	2	Oatmeal	boil oats	30	1	Breakfast
3	3	Good soup	boil things	30	5	Lunch
4	4	Smoothie	blend fruits	10	4	Beverage
5	5	Apple Juice	juice apples	10	2	Beverage


RECIPE\_CREATEDBY

	RECIPEID	TITLE	INSTRUCTIONS	COOKINGTIME	SERVINGSIZE	USERID
1	1	Best Buttermilk B...	insert instruction	50	4	1
2	2	Oatmeal	boil oats	30	1	2
3	3	Good soup	boil things	30	5	3
4	4	Smoothie	blend fruits	10	4	4
5	5	Apple Juice	juice apples	10	2	5
6	111	w	s	1	1	1
7	123	dcwce	xwxwd	10	1	1

ALLERGNINFORMATION

	 ALLERGENID ↕	 ALLERGENTYPE ↕
1	1	Gluten
2	2	Dairy
3	3	Nuts
4	4	Eggs
5	5	Soy




*DIFFICULTYLEVEL*

	 LEVEL ↕
1	1
2	2
3	3
4	4
5	5



*EQUIPMENT*

## University of British Columbia, Vancouver

Department of Computer Science

	 EQUIPMENTID ▾	 CATEGORY ▾	 NAME ▾
1	1	Cutting	Paring Knife
2	2	Cooking	Microwave
3	3	Cooking	Oven
4	4	Pots and Pans	Frying Pan
5	5	Measuring	Liquid Measure




### INCLUDES

	 RECIPEID ▾	 INGREDIENTID ▾
1	1	1
2	1	10
3	4	8
4	4	9
5	5	8



### INGREDIENT

## University of British Columbia, Vancouver



Department of Computer Science

	 INGREDIENTID ▾	 NAME ▾	 MEASUREMENT... ▾
1	1	Eggs	unit
2	2	Rice	gram
3	3	Chicken Breast	gram
4	4	Potato	unit
5	5	Cow Milk	mL
6	6	Steak	gram
7	7	Salmon	gram
8	8	Apple	unit
9	9	Orange	unit
10	10	Flour	mL
11	11	Peanut	unit
12	12	Almond	unit



### INGREDIENT\_HAS\_ALLERGENINFORMATION

	 INGREDIENTID ▾	 ALLERGENID ▾
1	1	4
2	5	2
3	10	1
4	11	3
5	12	3

### INGREDIENQUANTITY




	 QUANTITY ▾	 INGREDIENTID ▾
1	1	1
2	1	2
3	1	3
4	2	300
5	2	400

MAYCONTAIN





	 RECIPEID ▾	 ALLERGENID ▾
1	1	1
2	1	2
3	1	3
4	1	4
5	2	3

University of British Columbia, Vancouver  
Department of Computer Science







NON\_VEGAN

	 INGREDIENTID ▾	 NAME ▾	 MEASUREMENTUNIT ▾
1	1	Eggs	unit
2	3	Chicken Breast	gram
3	5	Cow Milk	mL
4	6	Steak	gram
5	7	Salmon	gram

NUTRITIONALINFORMATION



	 NUTRITIONID ▾	 CALORIES ▾	 MACRONUTRIENTS ▾	 VITAMINSMINERALS ▾
1	1	212	4.2 g Protein, 9.8 ...	Cholesterol 1.8 mg,...
2	2	95	5 g Protein, 3 g fa...	8.1 g fiber per 81 g
3	3	80	5.3 g Protein, 2.2 ...	Vitamin A 206.4 µg,...
4	4	513	21 g Protein , 25 g...	vitamin C 12.3 mg p...
5	5	114	0.2 g Protein, 0.3 ...	vitamin C 13.3 mg ,...

RECIPE\_HAS\_DIFFICULTYLEVEL



	 RECIPEID ▾	 TITLE ▾	 INSTRUCTIONS ▾	 COOKINGTIME ▾	 SERVINGSIZE ▾	 LEVEL ▾
1	1	Best Buttermilk Biscui...	insert instruction	50	4	4
2	2	Oatmeal	boil oats	30	1	2
3	3	Good soup	boil things	30	5	3
4	4	Smoothie	blend fruits	10	4	2
5	5	Apple Juice	juice apples	10	2	1

RECIPE\_HAS\_NUTRITIONALINFORMATION





	 RECIPEID ↕	 NUTRITIONID ↕
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

#### REQUIRES

	 EQUIPMENTID ↕	 RECIPEID ↕
1	1	3
2	2	2
3	3	1
4	5	4
5	5	5

#### REVIEW\_HAS

	 REVIEWID ↕	REVIEWTEXT	DATE ↕	 RECIPEID ↕
1	1	Tastes great!	2024-03-01	3
2	2	I like the process of doing it.	2024-03-02	4
3	3	It would be better if it can taste sweeter	2024-03-05	1
4	4	Took too long to make it.	2024-04-02	1
5	5	Nice!	2024-03-01	5

## REVIEW\_LEAVES

	🔍 REVIEWID ↕	📄 REVIEWTEXT ↕	📅 DATE ↕	👤 USERID ↕
1	1	Tastes great!	2024-03-01	2
2	2	I like the process of doing it.	2024-03-02	3
3	3	It would be better if it can taste sweeter	2024-03-05	1
4	4	Took too long to make it.	2024-04-02	4
5	5	Nice!	2024-03-01	5

## VEGAN

	🔍 INGREDIENTID ↕	📄 NAME ↕	📄 MEASUREMENTUNIT ↕
1	2	Rice	gram
2	4	Potato	unit
3	8	apple	unit
4	9	Orange	unit
5	10	Flour	ml
6	11	Peanut	unit
7	12	Almond	unit

## 5. Queries

Note: numbers refer to line numbers in file DatabaseConnectionHandler.java

*INSERT (302-322)*

*The 'insert recipe' button allows user to add a recipe and their creator info.*

**-before**

RecipeID:

Title:

Instructions:

Cooking Time:

Serving Size:

User Id

Recipe ID	Title	Instructions	Cooking Time	Serving Size	User ID
1	Best Buttermilk Biscuits	insert instruction	50	4	1
2	Oatmeal	boil oats	30	1	2
3	Good soup	boil things	30	5	3
4	Smoothie	blend fruits	10	4	4
5	Apple Juice	juice apples	10	2	5
111	w	s	1	1	1
123	dcwce	xxxxwd	10	1	1
999	demorec	donth	5	1	1

Back to Menu

Add Recipe

-doing

RecipeID:

Title:

Instructions:

Cooking Time:

Serving Size:

User Id

888

lovely lemon pie

bake lemon pie

5

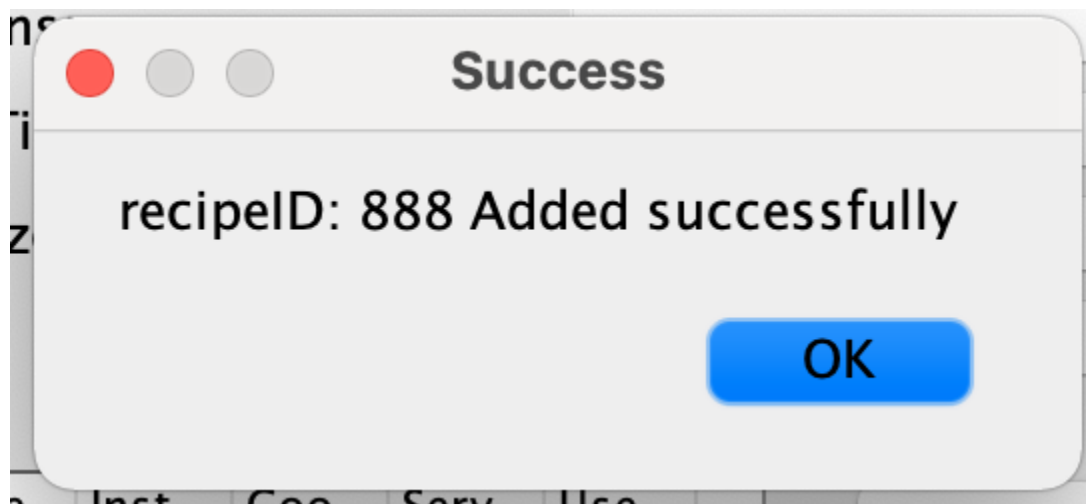
4

1

Recipe ID	Title	Instructions	Cooking Time	Serving Size	User ID
1	Best Buttermilk Biscuits	insert instruction	50	4	1
2	Oatmeal	boil oats	30	1	2
3	Good soup	boil things	30	5	3
4	Smoothie	blend fruits	10	4	4
5	Apple Juice	juice apples	10	2	5
111	w	s	1	1	1
123	dcwce	xxxxwd	10	1	1
999	demorec	donth	5	1	1

Back to Menu

Add Recipe



-after

RecipeID:	
Title:	
Instructions:	
Cooking Time:	
Serving Size:	
User id	

Recipe ID	Title	Instructions	Cooking Time	Serving Size	User ID
1	Best Buttermilk Biscuits	insert instruction	50	4	1
2	Oatmeal	boil oats	30	1	2
3	Good soup	boil things	30	5	3
4	Smoothie	blend fruits	10	4	4
5	Apple Juice	juice apples	10	2	5
111	w	s	1	1	1
123	dcwce	xxwxd	10	1	1
999	demorec	donth	5	1	1
888	lovely lemon pie	bake lemon pie	5	4	1

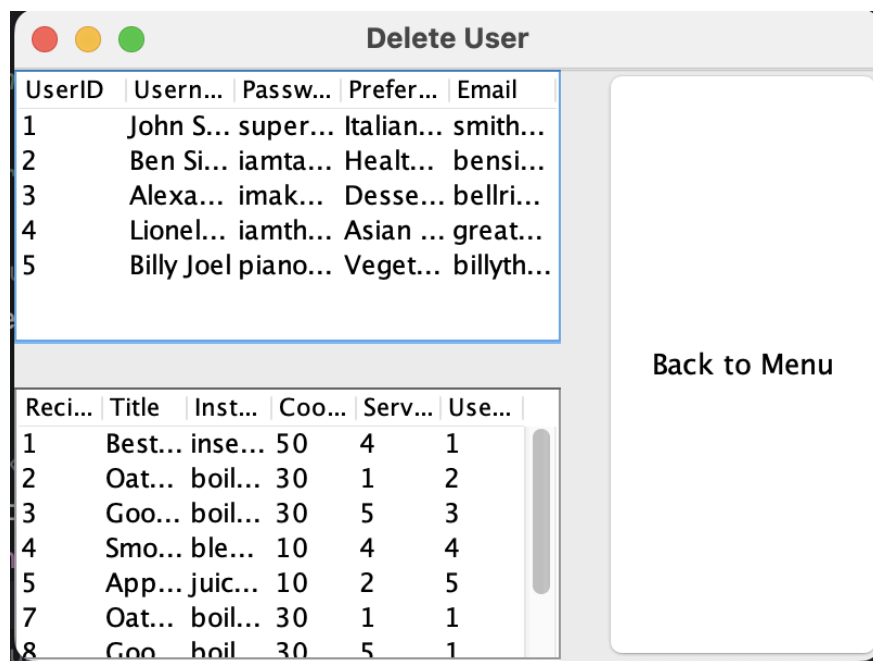
Back to Menu

Add Recipe

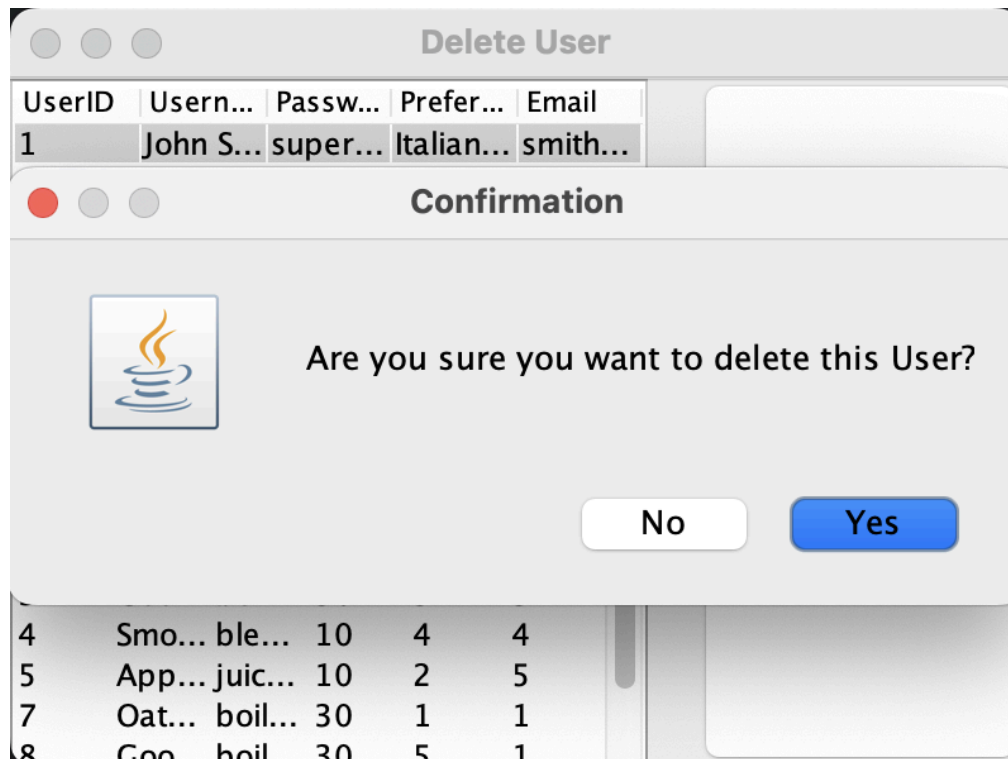
## DELETE(280-300)

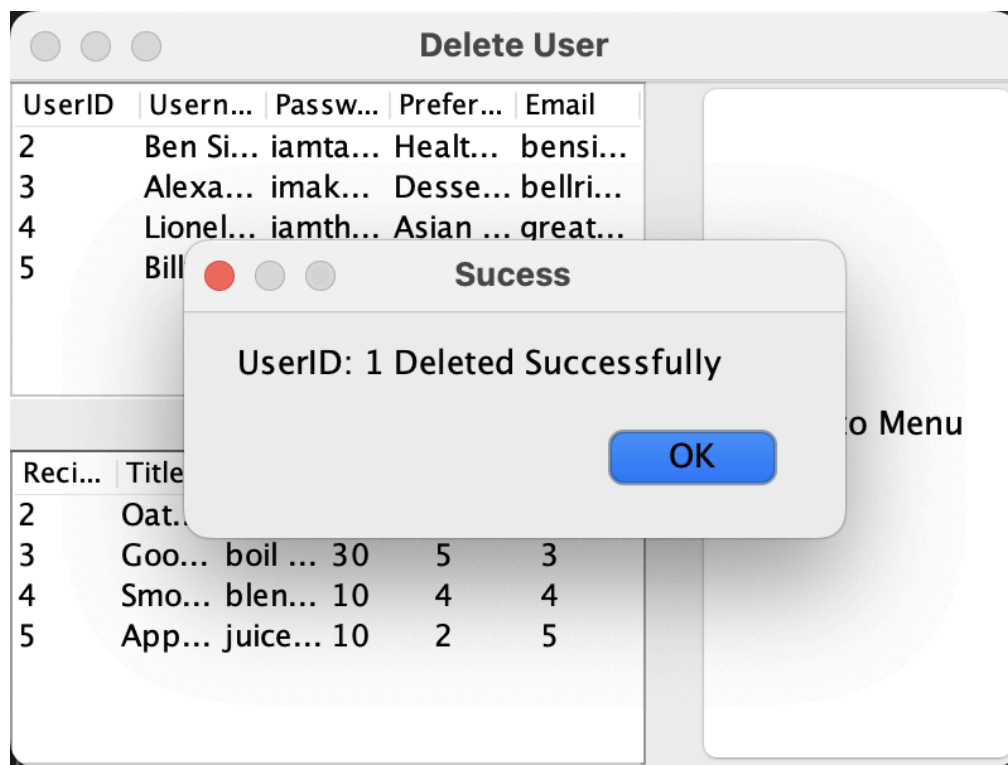
The 'delete user' button allows the deletion of a user from the database.

-before

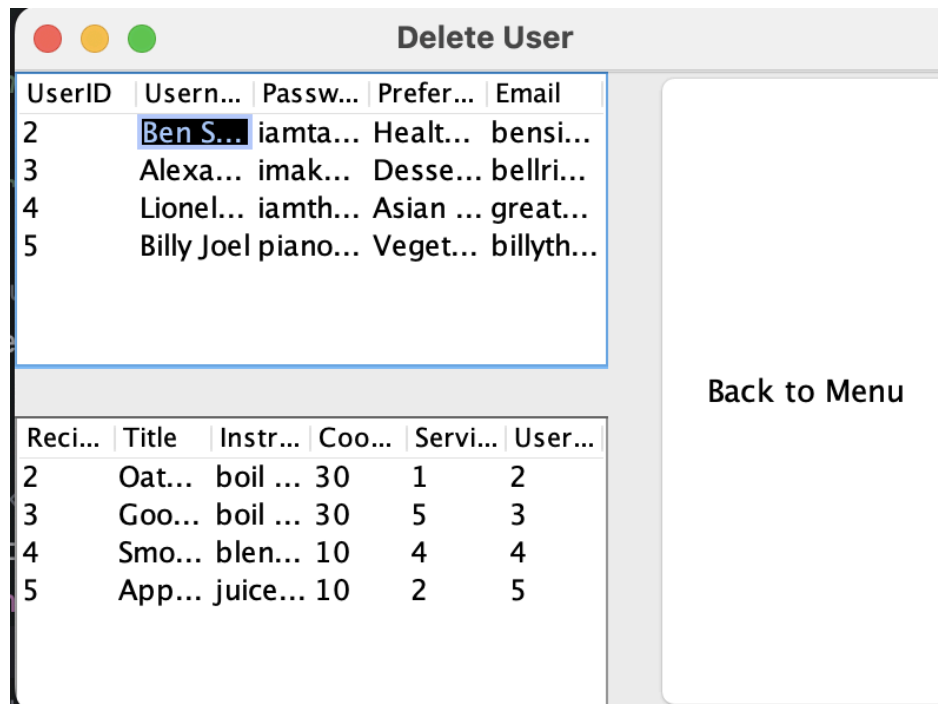


-doing





-after

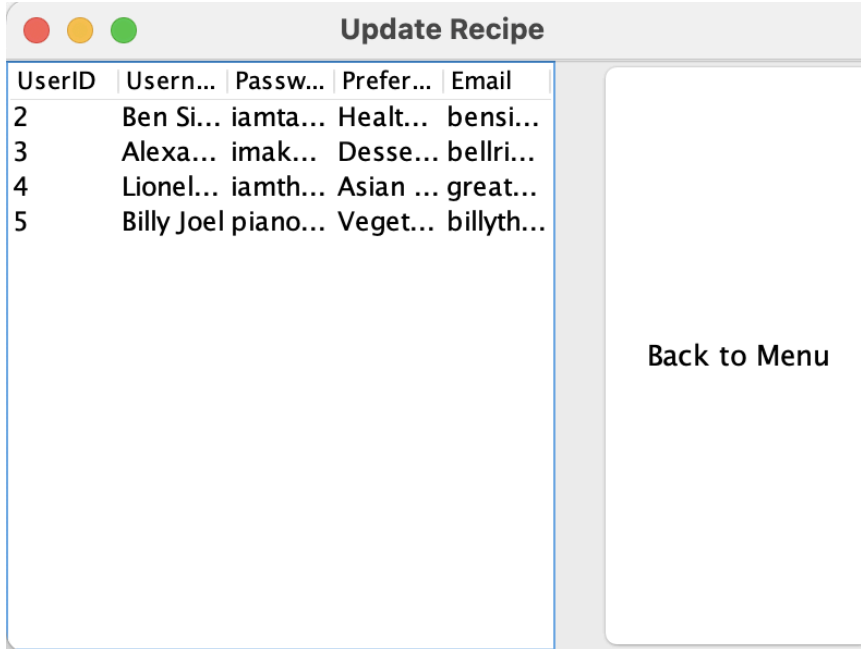


---

UPDATE(254-277)

The 'update user' button allows updating info for a user in the database.

-before

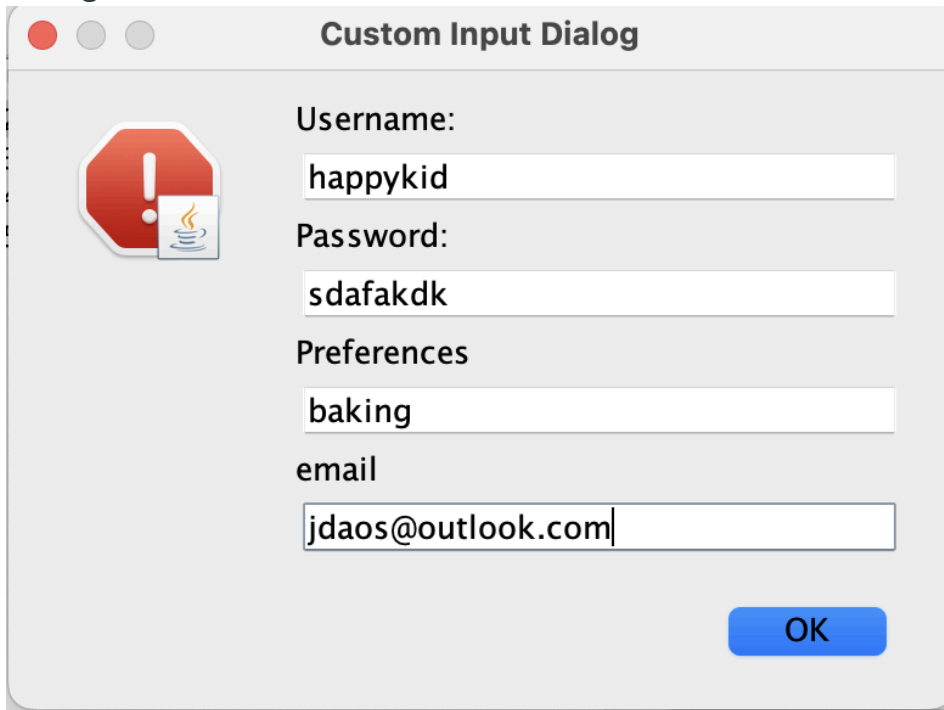


The 'Update Recipe' window displays a table with the following data:

UserID	Usern...	Passw...	Prefer...	Email
2	Ben Si...	iamta...	Healt...	bensi...
3	Alexa...	imak...	Desse...	bellri...
4	Lionel...	iamth...	Asian ...	great...
5	Billy Joel	piano...	Veget...	billyth...

Below the table is a button labeled 'Back to Menu'.

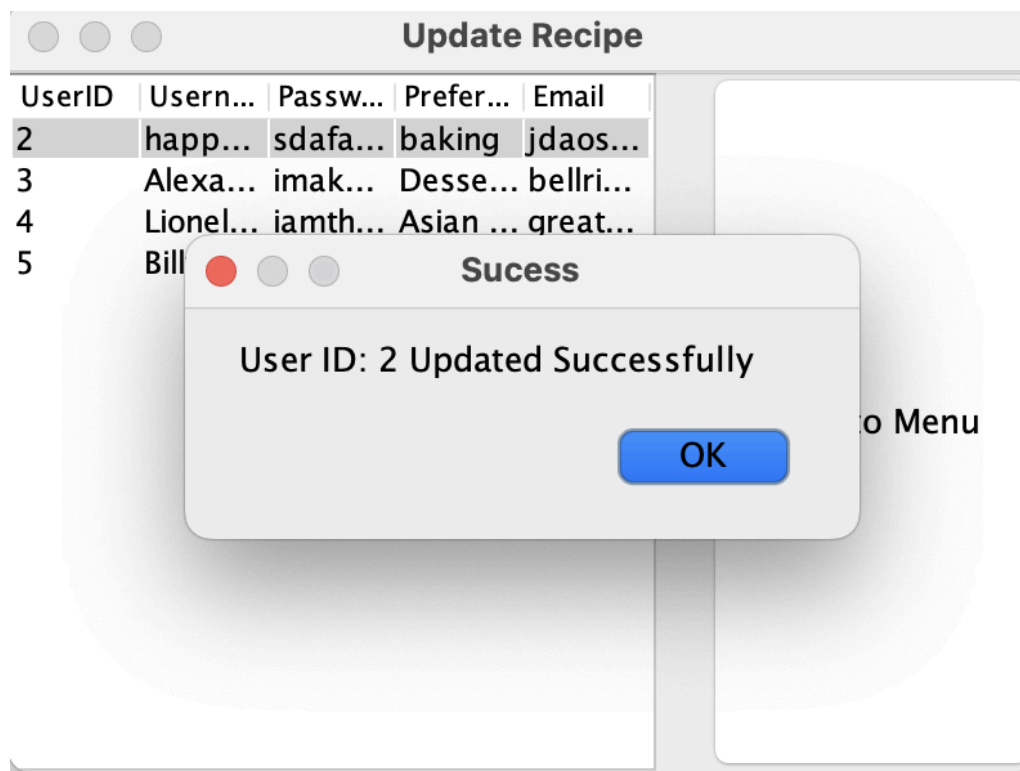
-doing



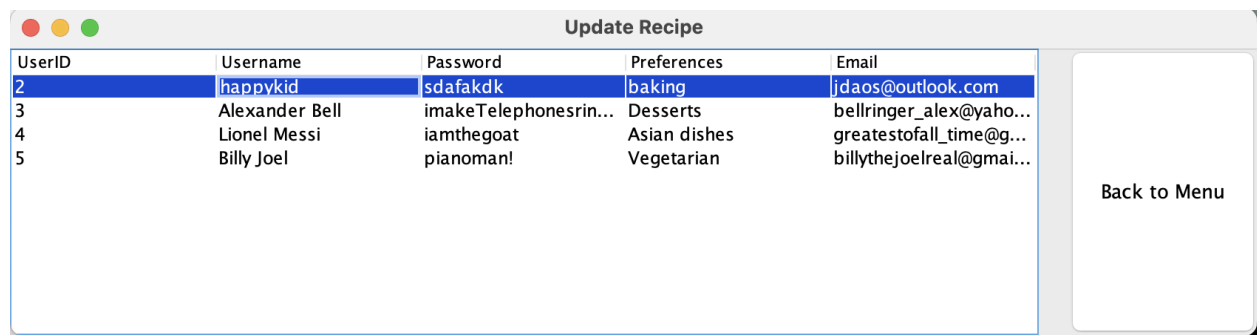
The 'Custom Input Dialog' window contains the following input fields:

- Username: happykid
- Password: sdafakdk
- Preferences: baking
- email: jdaos@outlook.com

An 'OK' button is located at the bottom right.



-after

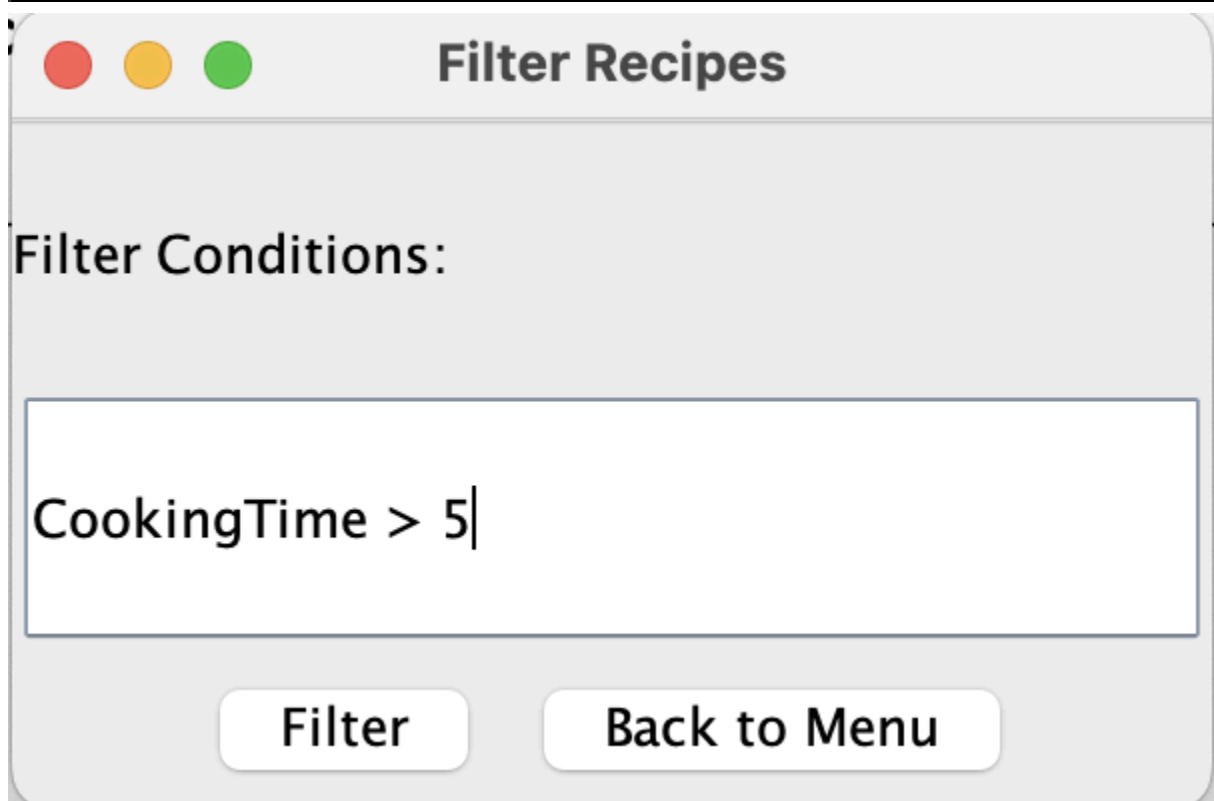


*SELECTION(75-103)*

*The 'filter recipes' button allows users to filter the recipe with their input 'where' part of the SQL statements.*

-before





A window titled "Filter Recipes" with a light gray background and rounded corners. At the top left are three colored circles (red, yellow, green). Below the title bar, the text "Filter Conditions:" is displayed. A large white text input field contains the text "CookingTime > 5". At the bottom of the window are two rounded buttons: "Filter" and "Back to Menu".

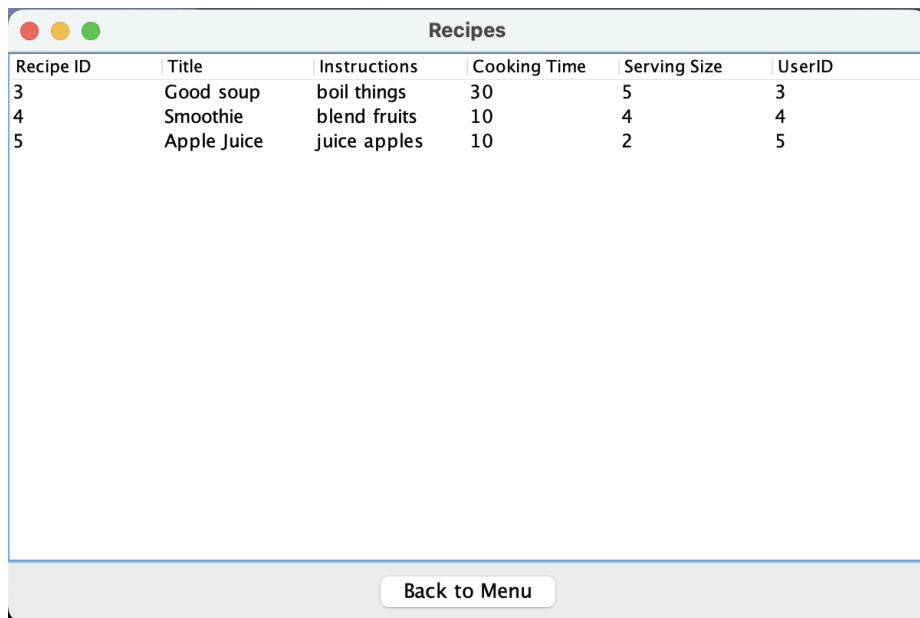
Filter Recipes

Filter Conditions:

CookingTime > 5

Filter Back to Menu

-after



A window titled "Recipes" with a light gray background and rounded corners. At the top left are three colored circles (red, yellow, green). The window contains a table with 6 columns: Recipe ID, Title, Instructions, Cooking Time, Serving Size, and UserID. The table has 3 rows of data. At the bottom of the window is a rounded button labeled "Back to Menu".

Recipe ID	Title	Instructions	Cooking Time	Serving Size	UserID
3	Good soup	boil things	30	5	3
4	Smoothie	blend fruits	10	4	4
5	Apple Juice	juice apples	10	2	5

Back to Menu

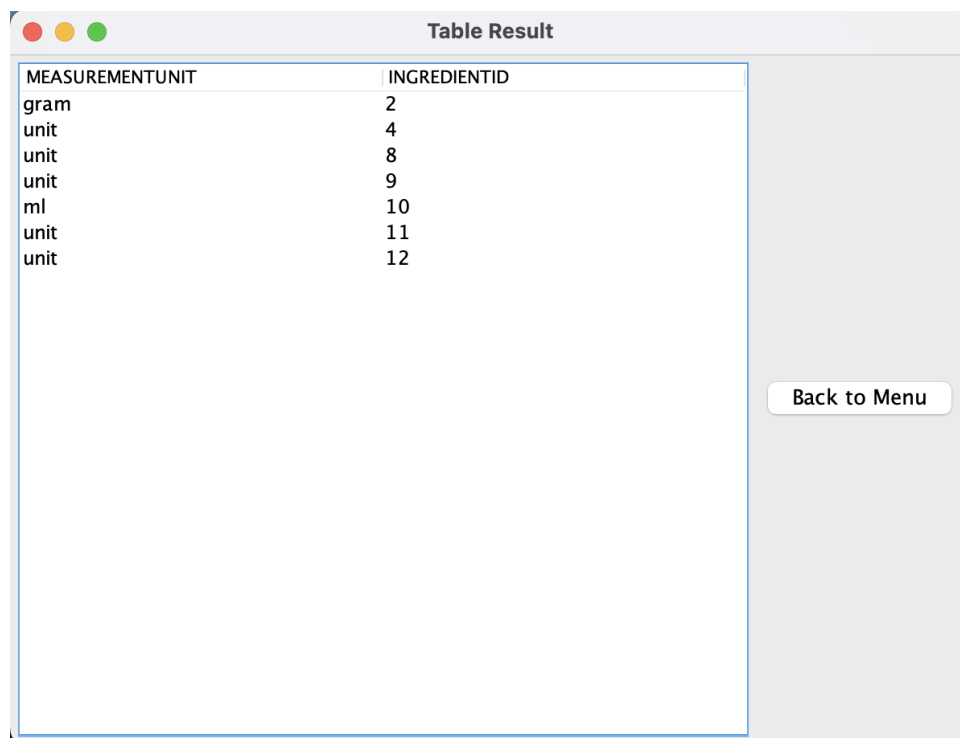
*PROJECTION(133-165)*

*The 'projection' button allows projecting various attributes of all schemas.*

**-doing**

The screenshot shows a window titled "Projection" with a light gray background. At the top, there is a text input field containing "VEGAN" and a blue button with a double arrow icon. To the right of the input field are two buttons: "Project" and "Back to Menu". Below the input field, there is a list of attributes with checkboxes. The first attribute is "NAME" with an unchecked checkbox. The second attribute is "MEASUREMENTUNIT" with a checked checkbox. The third attribute is "INGREDIENTID" with a checked checkbox.

**-after**



A window titled "Table Result" with a table containing two columns: "MEASUREMENTUNIT" and "INGREDIENTID". The table lists measurement units and their corresponding ingredient IDs. A "Back to Menu" button is located on the right side of the window.

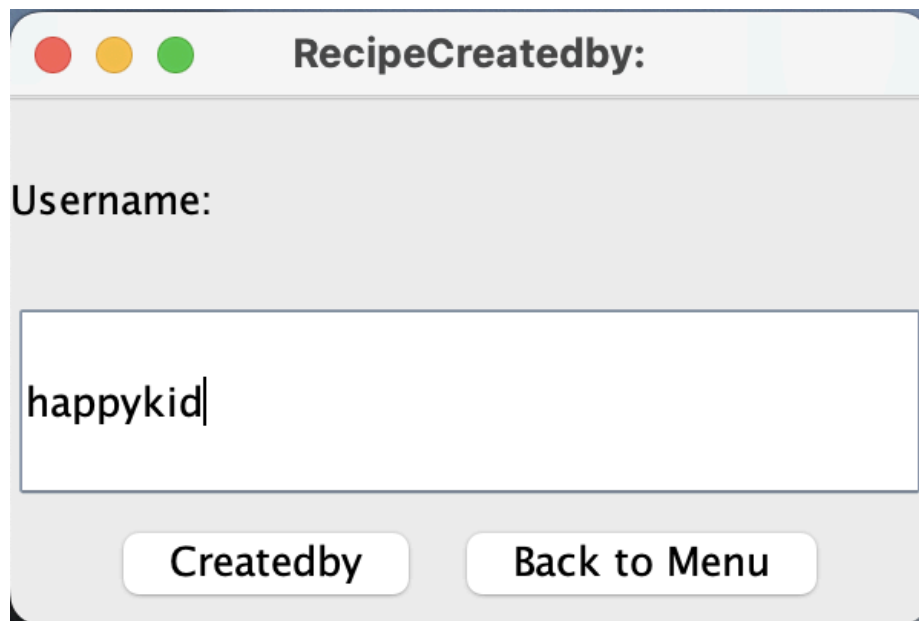
MEASUREMENTUNIT	INGREDIENTID
gram	2
unit	4
unit	8
unit	9
ml	10
unit	11
unit	12

Back to Menu

*JOIN (221-249)*

*The 'view recipe created by' button allows users to input a username, and will search recipes created by the user using a join of 'RecipeCreatedBy' and 'RecipeUsers'.*

**-before**



A window titled "RecipeCreatedBy:" with a "Username:" label and a text input field containing the text "happykid". Below the input field are two buttons: "Createdby" and "Back to Menu".

RecipeCreatedBy:

Username:

happykid

Createdby Back to Menu

- after

Recipes					
Recipe ID	Title	Instructions	Cooking Time	Serving Size	UserID
2	Oatmeal	boil oats	30	1	2

Back to Menu

#### AGGREGATION WITH GROUP BY (428-446)

The 'view total recipes per user' button displays the sum of all recipes grouped by user ids.

Total Recipes Per User	
UserID	Total Recipes
2	1
4	1
5	1
3	1

Back to Menu

*AGGREGATION WITH HAVING (408-426)*

*The 'view users with > 2 recipes' displays active users who have uploaded more than 2 recipes. It'll show the user id and recipe count.*

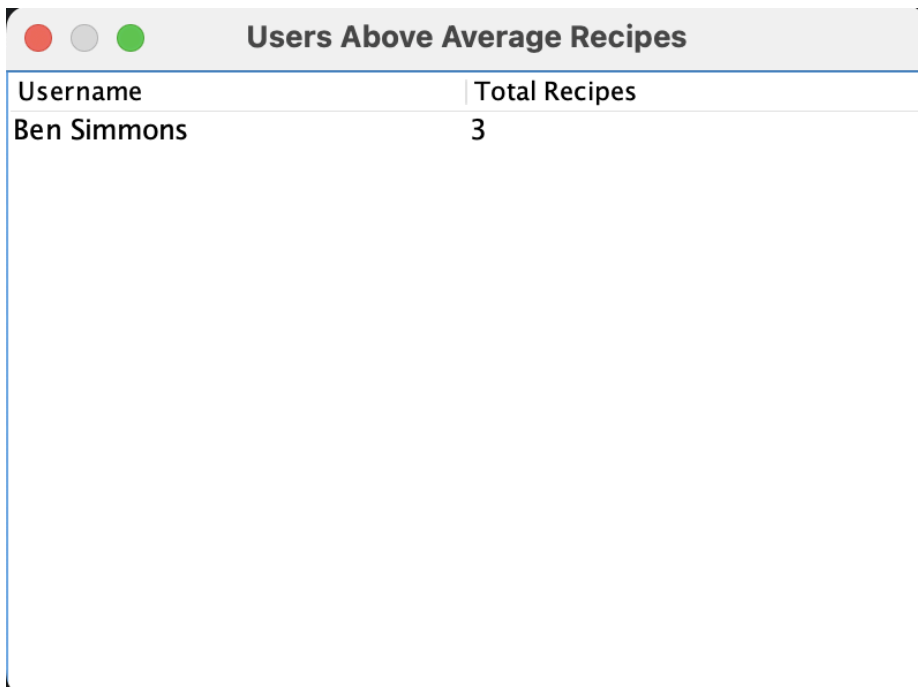


A screenshot of a web application window titled "Users with More Than Two Recipes". The window contains a table with two columns: "UserID" and "Recipe Count". The table has one row with the values "2" and "3" respectively.

UserID	Recipe Count
2	3

*NESTED AGGREGATION WITH GROUP BY (448-474)*

*The 'show user above average recipes' button displays users whose total recipe number is above average.*

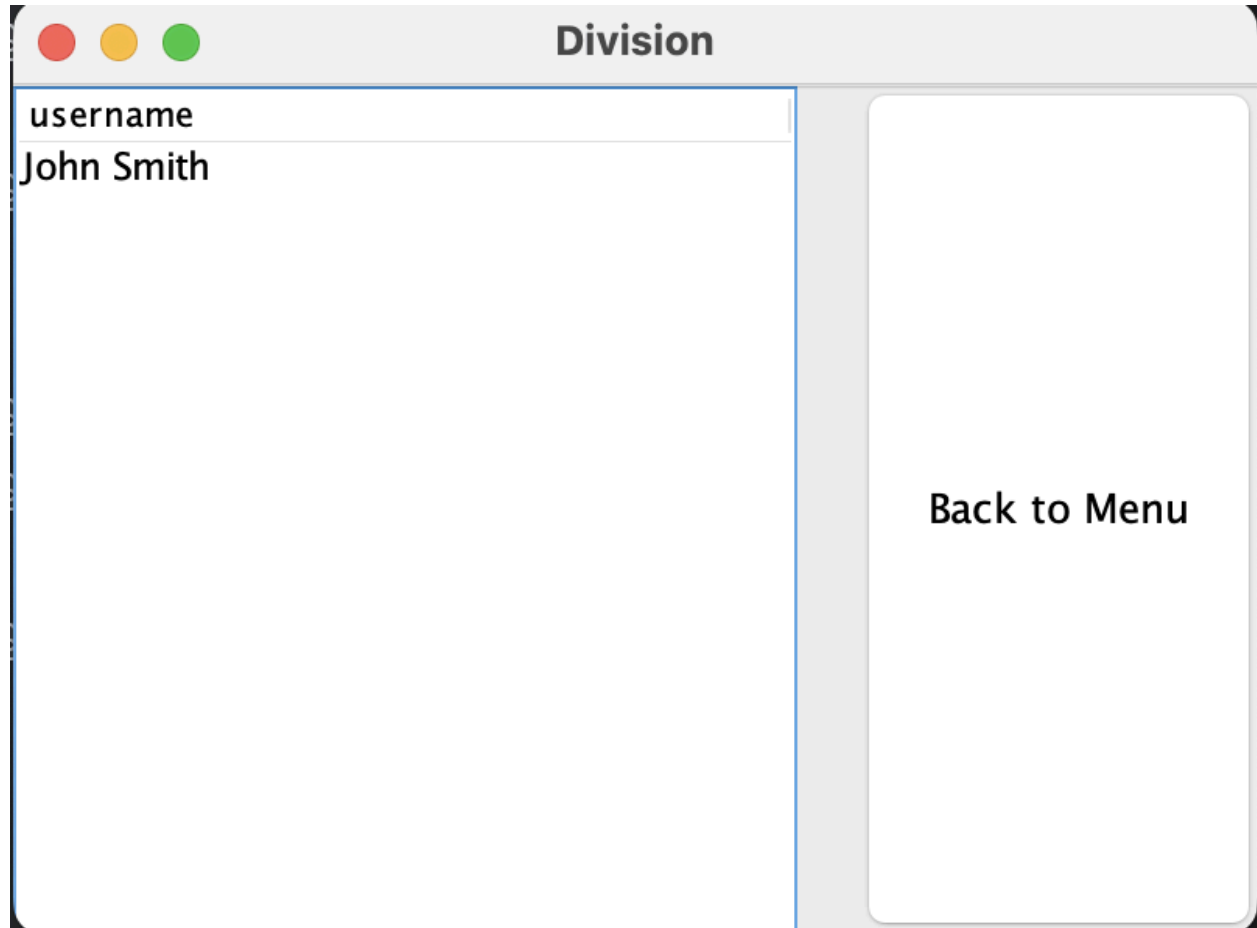


A screenshot of a web application window titled "Users Above Average Recipes". The window contains a table with two columns: "Username" and "Total Recipes". The table has one row with the values "Ben Simmons" and "3" respectively.

Username	Total Recipes
Ben Simmons	3

*DIVISION (47-70)*

*The 'division' button shows usernames who have all meal types in the database.*



*Additional Function: Random Projection*

*This button allows users to input 'from' and 'where' of SQL statements directly to get output.*

**List of all SQL queries used and where it can be found:**

**1. Division (Line 51 in DatabaseConnectionHandler.java)**

```
SELECT DISTINCT u.username
FROM RecipeUsers u
WHERE NOT EXISTS (SELECT mt.Type FROM recipe_isof mt
                  WHERE NOT EXISTS (SELECT * FROM Recipe_isOf r
                                    JOIN Recipe_createdby rc ON r.RecipeID = rc.RecipeID
                                    WHERE r.Type = mt.Type AND rc.userid = u.userid))
```

**2. FilterRecipe (Line 75 in DatabaseConnectionHandler.java)**

```
SELECT *
FROM recipe_CreatedBy
WHERE UserInput
```

**3. Projection (Line 109 in DatabaseConnectionHandler.java)**

```
SELECT Selection
FROM From
```

[Selection and From are selected by user](#)

**4. getTableNames (Line 174 in DatabaseConnectionHandler.java)**

```
SELECT table_name
FROM user_tables
```

**5. getColNames (Line 200 in DatabaseConnectionHandler.java)**

```
SELECT DISTINCT column_name
FROM all_tab_columns
WHERE table_name = ?
```

**6. RecipeCreatedBy (Line 225 in DatabaseConnectionHandler.java)**

```
SELECT *
FROM recipe_CreatedBy, RecipeUsers
WHERE recipe_CreatedBy.UserID = RecipeUsers.UserID AND
      RecipeUsers.Username = ?
```

**7. updateUser (Line 256 in DatabaseConnectionHandler.java)**

```
UPDATE RecipeUSERS SET USERNAME = ?, PASSWORD = ?, PREFERENCES = ?,  
EMAIL = ? WHERE USERID = ?
```

**8. deleteRecipeUser (Line 282 in DatabaseConnectionHandler.java)**

```
DELETE FROM RecipeUSERS  
WHERE USERID = ?
```

**9. insertRecipeUser (Line 304 in DatabaseConnectionHandler.java)**

```
INSERT INTO RECIPE_CREATEDBY VALUES (?, ?, ?, ?, ?, ?)
```

**10. getRecipeUserInfo (Line 328 in DatabaseConnectionHandler.java)**

```
SELECT * FROM RECIPE_CREATEDBY
```

**11. getUserInfo (Line 357 in DatabaseConnectionHandler.java)**

```
SELECT * FROM RecipeUSERS
```

**12. getUsersWithMoreThanTwoRecipes (Line 411 in DatabaseConnectionHandler.java)**

```
SELECT UserID, COUNT(*) AS RecipeCount  
FROM Recipe_CreatedBy  
GROUP BY UserID  
HAVING COUNT(*) > 2
```

**13. getTotalRecipesPerUser (Line 431 in DatabaseConnectionHandler.java)**

```
SELECT UserID, COUNT(*) AS RecipeCount  
FROM Recipe_CreatedBy  
GROUP BY UserID
```

**14. getUsersAboveAverageRecipes (Line 256 in DatabaseConnectionHandler.java)**

```
SELECT u.username, COUNT(*) AS total_recipes  
FROM Users u JOIN Recipe_createdby rc ON u.userid = rc.userid  
GROUP BY u.username  
HAVING COUNT(*) > ( SELECT AVG(recipe_count)  
                     FROM (SELECT userid, COUNT(*) AS recipe_count  
                           FROM Recipe_createdby  
                           GROUP BY userid) tempTable)
```



## University of British Columbia, Vancouver

Department of Computer Science

---

### List of relational schemas with PK and FK underlined and bolded respectively

Equipment (equipmentID: INTEGER, Category: CHAR(20), Name: VARCHAR)

- Not Null: Name

requires(equipmentID: INTEGER, **RecipeID**: INTEGER)

Recipe(RecipeID: INTEGER, Title: VARCHAR, Instructions: VARCHAR, CookingTime: INTEGER, Serving Size INTEGER)

- Not Null: Instructions

isOf(**RecipeID**: INTEGER, **Type**: VARCHAR)

MealType(Type: VARCHAR)

Recipe\_has\_DifficultyLevel(**RecipeID**: INTEGER, **Level**: INTEGER)

DifficultyLevel(Level: INTEGER)

includes(**RecipeID**: INTEGER, **IngredientID**: INTEGER)

Ingredient(IngredientID: INTEGER, Name: VARCHAR, MeasurementUnit: VARCHAR)

Non-Vegan(**IngredientID**: INTEGER, Name: VARCHAR, MeasurementUnit: VARCHAR)

Vegan(**IngredientID**: INTEGER, Name: VARCHAR, MeasurementUnit: VARCHAR)

- Not Null: Name, Measurement Unit

IngredientQuantity(Quantity: INTEGER, **IngredientID**: INTEGER)

Ingredient\_has\_AllergenInformation(**IngredientID**: INTEGER, **AllergenID**: INTEGER)

mayContain(**RecipeID**: INTEGER, **AllergenID**: INTEGER)

AllergenInformation(AllergenID: INTEGER, AllergenType: VARCHAR)

- Not Null: AllergenType

createdBy(**RecipeID**: INTEGER, **UserID**: INTEGER)

User(**UserID**: INTEGER, Username: VARCHAR, Password: VARCHAR, Preferences: VARCHAR, email: VARCHAR)

- Unique: Username, email
- Not Null: Username, Password, email
- Candidate Key: Username, email

leaves(**UserID**: INTEGER, **reviewID**: INTEGER)

Review(reviewID: INTEGER, Review Text: VARCHAR, Date: DATE)

Review\_has\_Recipe(reviewID: INTEGER, RecipeID: INTEGER)

Recipe\_has\_NutritionalInformation(RecipeID: INTEGER, NutritionID: INTEGER)

NutritionalInformation(NutritionalID: INTEGER, Calories: INTEGER, Macronutrients: VARCHAR, Vitamins & Minerals: VARCHAR)