Georgian Recipe Management System (GRMS)

Description:

Create a Recipe Management System (RMS) specifically tailored to Georgian cuisine. The GRMS will allow users to store, add, and remove traditional Georgian recipes, as well as print recipe information.

Class Recipe:

- Fields: **title**, **ingredients**, **instructions**
- Methods: **getTitle()**, **setTitle()**, **getIngredients()**, **setIngredients()**, **getInstructions()**, **setInstructions()**, **toString()**

```
package midterm.tekla_jikhvashvili_2.task3;

4 usages
public class Recipe {
    3 usages
    private String title;
    3 usages
    private String ingredients;
    3 usages
    private String instructions;

    no usages
    public String getTitle() {
        return title;
    }

    no usages
    public void setTitle(String title) {
        this.title = title;
    }

    no usages
    public String getIngredients() {
        return ingredients;
    }

    no usages
    public void setIngredients(String ingredients) {
        this.ingredients = ingredients;
    }
```

```java
    no usages
    public void setIngredients(String ingredients) {
        this.ingredients = ingredients;
    }

    no usages
    public String getInstructions() {
        return instructions;
    }

    no usages
    public void setInstructions(String instructions) {
        this.instructions = instructions;
    }

    @Override
    public String toString() {
        return "Recipe{" +
                "title='" + title + '\'' +
                ", ingredients='" + ingredients + '\'' +
                ", instructions='" + instructions + '\'' +
                '}';
    }
}
```

Class GRMS:

- Inner structure: Storage for recipes.
- Methods: **addRecipe(Recipe recipe)**, **removeRecipe(Recipe recipe)**, **printRecipes()**

```java
package midterm.tekla_jikhvashvili_2.task3;

import java.util.ArrayList;
import java.util.List;

no usages
public class GRMS {
    4 usages
    private List<Recipe> storage = new ArrayList<>();

    no usages
    public void addRecipe(Recipe recipe) {
        storage.add(recipe);
    }

    no usages
    public boolean removeRecipe(Recipe recipe) {
        return storage.remove(recipe);
    }
```

```java
    no usages
    public void printRecipes() {
        if (storage.isEmpty()) {
            System.out.println("The storage is empty");
        } else {
            for (Recipe recipe : storage) {
                System.out.println(recipe);
            }
        }
    }
}
```

Class GRMSTester:

- Main method to test the GRMS functionality.

- Creates sample recipes and tests adding, removing, and printing recipes.

```java
package midterm.tekla_jikhvashvili_2.task3;

public class GRMSTester {
    public static void main(String[] args) {
        Recipe r1 = new Recipe();
        r1.setTitle("Khachapuri");
        r1.setIngredients("Flour, cheese, eggs, butter, milk");
        r1.setInstructions("Mix flour, cheese, eggs, and butter. Bake in the oven.");

        Recipe r2 = new Recipe();
        r2.setTitle("Khinkali");
        r2.setIngredients("Flour, meat, onion, water");
        r2.setInstructions("Make dough with flour and water. Fill with meat and onion. Boil in water.");

        GRMS grms = new GRMS();
        grms.addRecipe(r1);
        grms.addRecipe(r2);

        grms.removeRecipe(r1);

        grms.printRecipes();
    }
}
```

Future Improvements:

1. **Recipe Availability:** Add the ability to track how many servings of each recipe are available.
2. **User Accounts:** Introduce user accounts for chefs or cooking enthusiasts, allowing them to save and manage their favorite recipes.
3.
4. **Advanced Search:** Implement advanced search functionalities, such as searching recipes by ingredients or difficulty level.
5. **Recipe Reviews:** Enable users to leave reviews and ratings for recipes they've tried.
6. **Integration with Georgian Ingredients Database:** Integrate the GRMS with a database of Georgian ingredients, providing users with information about traditional ingredients used in Georgian cuisine.