**# recipspro**

Program to suggest a recipe based on available ingredients and use-by dates

GITHub: <https://github.com/naveenshiraguppi/recipspro>

**Approach:**

1. Decide on algorithm to support the core logic of picking the best suitable receips based on following criterias:

1.a. An ingredient that is past its use-by date cannot be used for cooking.

1.b. If more than one recipe is found, then preference should be given to the recipe with the closest use-by item

1.c. If more than one recipe is found, then preference should be given to the number of ingredients with closest use-by item. - THIS IS NOT PART OF ORIGINAL REQUIREMENT

1.d. If more than one recipe is found, then preference should be given to the total amount of ingredients in each of the matching ingredients. - THIS IS NOT PART OF ORIGINAL REQUIREMENT

2.INPUT: two files. One .csv file for ingredients. Second .json file for recipes.

3.OUTPUT: If receip is not found then return "Order Takeout". Else return the Salad Sandwitch.

3.a Note that the input JSON file contains name of the recipe as 'salad sandwitch' but program needs to return 'Salad Sandwitch'.

**Program design:**

1. CSVReader.java reads from the file provided as first argument and returns collection of instances Ingredient class.

2. JSONReader.java reads from the file provided as second argument and returns a collection of instances of Recipe class.

3. Ingredient.java class contains item (String name), amount (int quantity), unit (Enum of measures), use\_by (date).

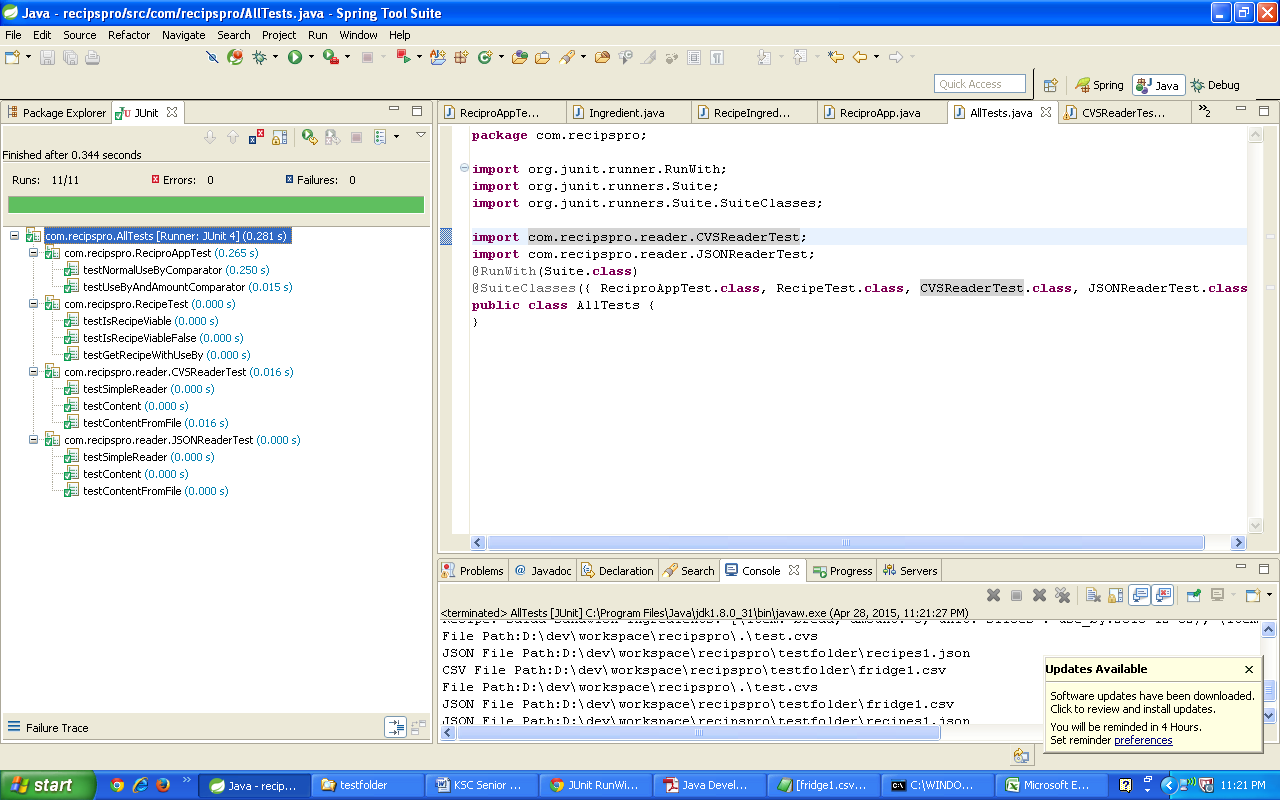
4. Recipe.java class contains name (String recipe name), collection of Ingredient objects without use\_by value.

5. ReciproApp.java. This class has the main method. Usage: class has the main method. Usage: class has the main method. Usage: **java com.recipspro.ReciproApp [fridge.csv recipe.json [options]]**

**Main features:**

* **Robust**
  + **Using Lambda expressions of Java 8.**
  + **Developed using Test Driven Development.**
* **Extensible**
  + **Using strategy of comparators to give flexible algorithm.**
* **Quality** 
  + **Unit tests for two reader classes**
  + **Unit tests for business methods implemented in Recipe.**
  + **Functional tests for two scenarios using different comparators.**
  + **A test suite class**
* **Usability through scripting**
  + **Run Script: “run-normal-input.bat” to use the comparator based on Use By date only.**
  + **Run Script for variant: “run-normal-input-amount-option.bat” to use the comparator based on Use By and Max Amount used.**
  + **Run Script: “run-help.bat” to get usage help.**

**Unit tests are successfully run:**

****