Package recipemanager

Class RecipeBox

java.lang.Object
    recipemanager.RecipeBox

```
public class RecipeBox
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`RecipeBox()`	Constructors RecipeBox() takes no arguments and will initialize the list of recipes
`RecipeBox(ArrayList<recipemanager.Recipe> listOfRecipes)`	RecipeBox(ArrayList) takes the existing list of recipes as a parameter and initializes the list of recipes with it

Method Summary

All Methods | Static Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
void	`addRecipe(recipemanager.Recipe recipe)`	Adds recipe to list of recipe
void	`createRecipe(recipemanager.MyMethods myMethods, Scanner scanner)`	Prompts user for information to create new recipe
`List<recipemanager.Recipe>`	`getListOfRecipes()`	gets List of Recipes
static void	`main(String[] args)`	Main method is the program's entry point
static int	`printMenu(recipemanager.MyMethods myMethods)`	Prints menu options to the user and returns their selection
void	`printRecipeDetails(RecipeBox recipeBox)`	Prints recipe information
void	`printRecipeNames(RecipeBox recipeBox)`	Prints recipe names
void	`setListOfRecipes(List<recipemanager.Recipe> listOfRecipes)`	Sets recipe list to array list

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

RecipeBox

```
public RecipeBox()
```

Constructors RecipeBox() takes no arguments and will initialize the list of recipes

RecipeBox

```
public RecipeBox(ArrayList<recipemanager.Recipe> listOfRecipes)
```

RecipeBox(ArrayList) takes the existing list of recipes as a parameter and initializes the list of recipes with it

Parameters:

`listOfRecipes` -

Method Details

main

```
public static void main(String[] args)
```

Main method is the program's entry point

Parameters:

`args` -

createRecipe

```
public void createRecipe(recipemanager.MyMethods myMethods,
                         Scanner scanner)
```

Prompts user for information to create new recipe

Parameters:

`myMethods` -

`scanner` -

addRecipe

```
public void addRecipe(recipemanager.Recipe recipe)
```

Adds recipe to list of recipe

Parameters:

`recipe` -

getListOfRecipes

```
public List<recipemanager.Recipe> getListOfRecipes()
```

gets List of Recipes

Returns:

List

setListOfRecipes

```
public void setListOfRecipes(List<recipemanager.Recipe> listOfRecipes)
```

Sets recipe list to array list

Parameters:

`listOfRecipes` -

printMenu

```
public static int printMenu(recipemanager.MyMethods myMethods)
```

Prints menu options to the user and returns their selection

Parameters:

`myMethods` -

Returns:

int

printRecipeNames

```
public void printRecipeNames(RecipeBox recipeBox)
```

Prints recipe names

Parameters:

`recipeBox` -

printRecipeDetails

```
public void printRecipeDetails(RecipeBox recipeBox)
```

Prints recipe information

Parameters:

`recipeBox` -

Package recipemanager

Class MyMethods

java.lang.Object
    recipemanager.MyMethods

```
public class MyMethods
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`MyMethods()`	

Method Summary

All Methods | **Instance Methods** | **Concrete Methods**

Modifier and Type	Method	Description
double	`readDouble(String prompt)`	Reads an double from the user.
double	`readDouble(String prompt, double low, double high)`	Reads double from user, validates the double is within range
float	`readFloat(String prompt)`	Reads a input from user and validates that it is a float
float	`readFloat(String prompt, float low, float high)`	Reads user input and validates that its a float within certain range
int	`readInteger(String prompt)`	Validates user input as string
int	`readInteger(String prompt, int low, int high)`	Reads an integer from the user, validate integer within range

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

MyMethods

```
public MyMethods()
```

Method Details

readInteger

```
public int readInteger(String prompt)
```

Validates user input as string

Parameters:

`prompt` - string prompt

Returns:

int

readInteger

```
public int readInteger(String prompt,
                       int low,
                       int high)
```

Reads an integer from the user, validate integer within range

Parameters:

`prompt` - string prompt

`low` - int low

`high` - int high

Returns:

int

readDouble

```
public double readDouble(String prompt)
```

Reads an double from the user. Validate double

Parameters:

`prompt` - string prompt

Returns:

double

readDouble

```
public double readDouble(String prompt,
                         double low,
                         double high)
```

Reads double from user, validates the double is within range

Parameters:

`prompt` - string prompt

`low` - int low

`high` - int high

Returns:

double

readFloat

```
public float readFloat(String prompt)
```

Reads a input from user and validates that it is a float

Parameters:

`prompt` - string prompt

Returns:

float

readFloat

```
public float readFloat(String prompt,
                       float low,
                       float high)
```

Reads user input and validates that its a float within certain range

Parameters:

`prompt` - string prompt

`low` - float low

`high` - float high

Returns:

float

Package recipemanager

Class Recipe

java.lang.Object
    recipemanager.Recipe

```
public class Recipe
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`Recipe(String name, int serving, List<Ingredient> ingredients, double calories)`	Constructor to initialize a recipe with its details

Method Summary

All Methods | **Instance Methods** | **Concrete Methods**

Modifier and Type	Method	Description
`Ingredient`	`addNewIngredient()`	Calls createFromUserInput from Ingredient class
`Ingredient`	`createNewIngredient(String name, float numberOfCups, int caloriesPerCup, double calories)`	Creates a new ingredient using given parameters
`List<Ingredient>`	`getRecipeIngredients()`	
`String`	`getRecipeName()`	Gets recipe name
`int`	`getServings()`	Gets number of servings
`double`	`getTotalRecipeCalories()`	Gets total calories in recipe
`void`	`printRecipe()`	Prints recipe information to console
`void`	`setRecipeIngredients(List<Ingredient> recipeIngredients)`	Sets recipe ingredients to the given parameter
`void`	`setRecipeName(String name)`	Sets recipe name to given parameter
`void`	`setServings(int servings)`	Sets serving to given parameter
`void`	`setTotalRecipeCalories (double totalRecipeCalories)`	Sets the total recipe calories to the given parameter

Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Constructor Details

Recipe

```
public Recipe(String name,
              int serving,
              List<Ingredient> ingredients,
              double calories)
```

Constructor to initialize a recipe with its details

Parameters:

`name` - string name

`serving` - int serving

`ingredients` - list ingredient

`calories` - double calories

Method Details

getRecipeName

`public String getRecipeName()`

Gets recipe name

Returns:

string

setRecipeName

`public void setRecipeName(String name)`

Sets recipe name to given parameter

Parameters:

`name` - string name

getServings

`public int getServings()`

Gets number of servings

Returns:

int

setServings

`public void setServings(int servings)`

Sets serving to given parameter

Parameters:

`servings` - int servings

getRecipeIngredients

`public List<Ingredient> getRecipeIngredients()`

Returns:

ArrayList of type Ingredient

setRecipeIngredients

`public void setRecipeIngredients(List<Ingredient> recipeIngredients)`

Sets recipe ingredients to the given parameter

Parameters:

`recipeIngredients` - list ingredient

getTotalRecipeCalories

```
public double getTotalRecipeCalories()
```

Gets total calories in recipe

Returns:

double

setTotalRecipeCalories

```
public void setTotalRecipeCalories(double totalRecipeCalories)
```

Sets the total recipe calories to the given parameter

Parameters:

`totalRecipeCalories` - double total calories

createNewIngredient

```
public Ingredient createNewIngredient(String name,
                                      float numberOfCups,
                                      int caloriesPerCup,
                                      double calories)
```

Creates a new ingredient using given parameters

Parameters:

`name` - string name

`numberOfCups` - int number of cups

`caloriesPerCup` - int calories per cup

`calories` - double calories

Returns:

Ingredient

addNewIngredient

```
public Ingredient addNewIngredient()
```

Calls createFromUserInput from Ingredient class

Returns:

Ingredient

printRecipe

```
public void printRecipe()
```

Prints recipe information to console

Package recipemanager

Class Ingredient

java.lang.Object
    recipemanager.Ingredient

```
public class Ingredient
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`Ingredient(String nameOfIngredient, float numberOfCups, int numberOfCaloriesPerCup)`	Constructor to initialize the ingredient

Method Summary

All Methods | Static Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
static Ingredient	createFromUserInput()	Creates an Ingredient object based on user input.
String	getNameOfIngredient()	Gets the name of the ingredient
int	getNumberOfCaloriesPerCup()	Gets the number of calories per cup
float	getNumberOfCups()	Gets the number of cups
double	getTotalCalories()	Gets the total calories
void	setNameOfIngredient(String name)	Sets the name of ingredient
void	setNumberOfCaloriesPerCup(int caloriesPerCup)	Sets the number calories per cup
void	setNumberOfCups(float number)	Sets the number of cups for ingredient
void	setTotalCalories(double totalCalories)	Sets the total calories
String	toString()	Returns a string representation of the ingredient.

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Details

Ingredient

```
public Ingredient(String nameOfIngredient,
                  float numberOfCups,
                  int numberOfCaloriesPerCup)
```

Constructor to initialize the ingredient

Parameters:

`nameOfIngredient` - the name of the ingredient

`numberOfCups` - the number of cups of the ingredient

`numberOfCaloriesPerCup` - the number of calories per cup of the ingredient

Method Details

getNameOfIngredient

`public String getNameOfIngredient()`

Gets the name of the ingredient

Returns:

the name of the ingredient

setNameOfIngredient

`public void setNameOfIngredient(String name)`

Sets the name of ingredient

Parameters:

`name` - of ingredient

getNumberOfCups

`public float getNumberOfCups()`

Gets the number of cups

Returns:

the number of cups

setNumberOfCups

`public void setNumberOfCups(float number)`

Sets the number of cups for ingredient

Parameters:

`number` - number of cups

getNumberOfCaloriesPerCup

`public int getNumberOfCaloriesPerCup()`

Gets the number of calories per cup

Returns:

the number of calories per cup

setNumberOfCaloriesPerCup

`public void setNumberOfCaloriesPerCup(int caloriesPerCup)`

Sets the number calories per cup

Parameters:

`caloriesPerCup` - calories in one cup

getTotalCalories

```
public double getTotalCalories()
```

Gets the total calories

Returns:

the total calories

setTotalCalories

```
public void setTotalCalories(double totalCalories)
```

Sets the total calories

Parameters:

`totalCalories` - total calories in ingredient

createFromUserInput

```
public static Ingredient createFromUserInput()
```

Creates an Ingredient object based on user input.

Returns:

Ingredient

toString

```
public String toString()
```

Returns a string representation of the ingredient.

Overrides:

`toString` in class `Object`

Returns:

a string describing the ingredient.