

## Stock APIs

Type	Method Name	Return Type	Functions
Constructor	<b>Stock</b> newstock = new <b>Stock</b> ( )	void	调用 stock
Add	<b>addItem</b> ( <b>String</b> name, <b>int</b> quantity, <b>Date</b> expiration, <b>FoodItem.FoodType</b> type, <b>FoodItem.PlaceLocation</b> location)	void	添加新的 item 需要传入 item 相关的所有参数
Edit	<b>updateItem</b> ( <b>FoodItem</b> item, <b>String</b> name, <b>int</b> quantity, <b>Date</b> expiration, <b>FoodItem.FoodType</b> type, <b>FoodItem.PlaceLocation</b> location)	void	修改原有的 item 需要传入原有 item, 以及所有新参数
Remove	<b>removeItem</b> ( <b>FoodItem</b> item)	void	删除指定 item 需要传入想删除的 item
Get Item List	<b>getItems</b> ( )	ArrayList<FoodItem>	获取全部的 stock items, 返回 ArrayList
	<b>getExpiredItems</b> ( <b>Stock.StockType</b> type)  【 <b>Stock.StockType</b> type 包括 : { <b>ALL, FROZEN, REFRIGERATED, VEGETABLE, MEAT, FRUIT, DRINK, OTHER</b> } 】	ArrayList<FoodItem>	获取已过期的 stock items, 需要传入 type 返回 ArrayList  例如 : getExpiredItems (Stock.StockType.ALL) 获取全部物品中已过期部分
	<b>getAlmostExpiredItems</b> ( <b>Stock.StockType</b> type)	ArrayList<FoodItem>	获取在三天内快过期的 stock items, 需传入 type 返回 ArrayList
	<b>getLowStockItems</b> ( <b>Stock.StockType</b> type)	ArrayList<FoodItem>	获取低库存的 stock items, 需要传入 type 返回 ArrayList
	<b>getItemsByType</b> ( <b>FoodItem.FoodType</b> type)	ArrayList<FoodItem>	获取指定 type 的全部 stock items, 需要传入 food type 返回 ArrayList
	<b>getItemsByLocation</b> ( <b>FoodItem.PlaceLocation</b> location)	ArrayList<FoodItem>	获取指定 location 的全部 stock items, 需要传入 location 返回 ArrayList

## FoodItem APIs

Type	Method Name	Return Type	Functions
Constructor	<code>FoodItem newItem = new FoodItem (String name, int quantity, Date expiration, FoodItem. FoodType type, FoodItem.PlaceLocation location)</code>	void	调用 FoodItem 需要输入全部相关参数 (使用时, 必须检查 name 非空, quantity>0,
Get information of Items	<code>getName ()</code>	<code>String</code> name	获取 item 名字
	<code>getQuantity ()</code>	<code>int</code> quantity	获取 item 数量
	<code>getExpiration()</code>	<code>Date</code> expiration	获取 item 过期日
	<code>getType ()</code>	<code>FoodItem. FoodType</code> type	获取 item 类型
	<code>getLocation()</code>	<code>FoodItem.PlaceLocation</code> location	获取 item 存放位置
Get URL	<code>getWFURL ()</code>	<code>String</code> URL	获取 Whole Foods URL
	<code>getAFURL ()</code>	<code>String</code> URL	获取 Amazon Fresh URL
String	<code>toString ()</code>	<code>String</code> info	输出 item 的全部信息 格式为 : "Name, quantity, date, type, location"

## RecipeDatabase APIs

Type	Method Name	Return Type	Functions
Constructor	<code>RecipeDatabase</code> recipeDatabase = new RecipeDatabase();	void	调用 recipeDatabase
Get List	<code>getAllRecipes</code> ( )	ArrayList<Recipe>	获取数据库内的所有 recipe list
	<code>getSpecificRecipeList</code> (String[ ] input Ingredients, int range)	ArrayList<Recipe>	输入冰箱里食材的 String[ ], 和需要菜谱的 多少数量, 获取指定数量的可用 recipe list

## RecipeItem APIs

Type	Method Name	Return Type	Functions
Get information of Recipes	<code>getName</code> ( )	String	获取 recipe 名字
	<code>ingredientsToString</code> ( )	String	获取 recipe 的食材表 之间用逗号隔开
	<code>detailsOfIngredientToString</code> ( )	String	获取 recipe 的食材详 情, 内部有换行
	<code>getMethod</code> ( )	String	获取 recipe 烹饪方法
	<code>getServing</code> ( )	String	获取 recipe 可食用份数
	<code>getCookTime</code> ( )	String	获取 recipe 烹饪时间
	<code>getCalories</code> ( )	String	获取 recipe 卡路里