|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | [**Package**](http://docs.google.com/rain/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ShoppingCart.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | | |  |
| [**PREV CLASS**](http://docs.google.com/rain/Recording.html)   [**NEXT CLASS**](http://docs.google.com/rain/ShoppingCartItem.html) | [**FRAMES**](http://docs.google.com/index.html?rain/ShoppingCart.html)    [**NO FRAMES**](http://docs.google.com/ShoppingCart.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **rain**

Class ShoppingCart

[java.lang.Object](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html)  
 **rain.ShoppingCart**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/docs/jdk_docs/api/java/io/Serializable.html)

public class **ShoppingCart**extends [Object](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html)implements [Serializable](http://docs.google.com/docs/jdk_docs/api/java/io/Serializable.html)

This class describes a shopping cart.

A shopping cart is a collection of shopping cart items. You can add items to the cart and remove them. Methods are available to get a list of the cart items.

Usage Example:

//  
 // Create the shopping cart  
 //  
 ShoppingCart userCart = new ShoppingCart();  
  
  
 //  
 // Add to the cart  
 Product theProduct = ...  
 userCart.add(theProduct);  
  
  
 //  
 // Remove items from the cart  
 //  
 userCart.remove(theProductId);  
  
  
 //  
 // Get a list of shopping cart items  
 //  
 Collection<ShoppingCartItem> items = userCart.getItems();  
  
  
 //  
 // Show the contents of the shopping cart  
 //  
  
 for (ShoppingCartItem tempItem : items) {  
 Product tempProduct = tempItem.getProduct();  
  
 System.out.println(tempProduct);  
 System.out.println("Quantity: " + tempItem.getQuantity());  
 }

**See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#rain.ShoppingCart)

|  |  |
| --- | --- |
| **Constructor Summary** | |
| [**ShoppingCart**](http://docs.google.com/rain/ShoppingCart.html#ShoppingCart())()            Constructs an empty shopping cart |

|  |  |
| --- | --- |
| **Method Summary** | |
| void | [**add**](http://docs.google.com/rain/ShoppingCart.html#add(rain.Product))([Product](http://docs.google.com/rain/Product.html) theProduct)            Add to the shopping cart. |
| void | [**add**](http://docs.google.com/rain/ShoppingCart.html#add(rain.ShoppingCartItem))([ShoppingCartItem](http://docs.google.com/rain/ShoppingCartItem.html) item)            Add to the shopping cart. |
| void | [**clear**](http://docs.google.com/rain/ShoppingCart.html#clear())()            Clears out the cart |
| [Collection](http://docs.google.com/docs/jdk_docs/api/java/util/Collection.html)<[ShoppingCartItem](http://docs.google.com/rain/ShoppingCartItem.html)> | [**getItems**](http://docs.google.com/rain/ShoppingCart.html#getItems())()            Returns an collection of ShoppingCartItems |
| int | [**getNumberOfProducts**](http://docs.google.com/rain/ShoppingCart.html#getNumberOfProducts())()            Returns the number of products in the shopping cart |
| double | [**getTotalPrice**](http://docs.google.com/rain/ShoppingCart.html#getTotalPrice())()            Returns the total price of all products in the shopping cart |
| boolean | [**isEmpty**](http://docs.google.com/rain/ShoppingCart.html#isEmpty())()            Returns true if number of products == 0, else false |
| void | [**remove**](http://docs.google.com/rain/ShoppingCart.html#remove(java.lang.String))([String](http://docs.google.com/docs/jdk_docs/api/java/lang/String.html) theProductId)            Removes a product from the shopping cart. |

|  |
| --- |
| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html) |
| [clone](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#clone()), [equals](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#toString()), [wait](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#wait()), [wait](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/docs/jdk_docs/api/java/lang/Object.html#wait(long,%20int)) |

|  |
| --- |
| **Constructor Detail** |

### ShoppingCart

public **ShoppingCart**()

Constructs an empty shopping cart

|  |
| --- |
| **Method Detail** |

### add

public void **add**([Product](http://docs.google.com/rain/Product.html) theProduct)

Add to the shopping cart. This is accomplished with the following steps:

1. Check to see if the product id exists in the cart
2. If so then grab that item and increment its quantity
3. Else create the cart item and add it to the cart

### add

public void **add**([ShoppingCartItem](http://docs.google.com/rain/ShoppingCartItem.html) item)

Add to the shopping cart.

**See Also:**[add(Product theProduct)](http://docs.google.com/rain/ShoppingCart.html#add(rain.Product))

### remove

public void **remove**([String](http://docs.google.com/docs/jdk_docs/api/java/lang/String.html) theProductId)

Removes a product from the shopping cart.

This is accomplished with the following steps:

1. Check to see if the product id exists in the cart
2. If so then grab that item and decrement its quantity
3. If this is the last cartItem then remove it from the shopping cart

### getItems

public [Collection](http://docs.google.com/docs/jdk_docs/api/java/util/Collection.html)<[ShoppingCartItem](http://docs.google.com/rain/ShoppingCartItem.html)> **getItems**()

Returns an collection of ShoppingCartItems

### getNumberOfProducts

public int **getNumberOfProducts**()

Returns the number of products in the shopping cart

### getTotalPrice

public double **getTotalPrice**()

Returns the total price of all products in the shopping cart

### clear

public void **clear**()

Clears out the cart

### isEmpty

public boolean **isEmpty**()

Returns true if number of products == 0, else false

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | [**Package**](http://docs.google.com/rain/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ShoppingCart.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | | |  |
| [**PREV CLASS**](http://docs.google.com/rain/Recording.html)   [**NEXT CLASS**](http://docs.google.com/rain/ShoppingCartItem.html) | [**FRAMES**](http://docs.google.com/index.html?rain/ShoppingCart.html)    [**NO FRAMES**](http://docs.google.com/ShoppingCart.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |