# Lab3A: Shopping Cart

**Objects and Classes : Problem Solving**

- This lab exercise is divided into three stages – Stage 1, Stage 2 and Stage 3.
- You need to complete Stage 1 without errors before you can proceed to Stage 2, and complete Stage 2 without errors before you can proceed to Stage 3.

**Problem Description:**

Maju Jaya Sdn. Bhd. is a small online shop that sells a number of clothing items to its customers. When a customer visits the shop via the internet, the customer will be given a shopping cart to hold the items that the customer would like to buy. However, the shopping cart can hold only three different items at a time. This is because the shop has an unusual policy that allows its customer to buy only three different items at a time.

## Stage 1:

1. Create a Java project named `Lab3Stage1`.
2. Copy class `Tester1.java` into your project.

Define class `Item`, `ShoppingCart` and `Customer` based on the following descriptions:

| Class: Item | Description |
|---|---|
| - id: String<br>- name: String<br>- price: double<br>- unit: int | Represents item's id.<br>Represents item's name.<br>Represents item's price.<br>Represents item's unit. |
| + Item()<br>+ Item(String, String, double)<br><br><br>+ toString() : String | Constructor with no parameter and empty body.<br>Constructor that assigns parameter 1 to item's id, parameter 2 to item's name, parameter 3 to item's price and 0 to item's unit.<br>Returns String "<Classname>[<item id>]". |

| Class: ShoppingCart | Description |
|---|---|
| - item1: Item<br>- item2: Item<br>- item3: Item | Represents shopping cart's item 1.<br>Represents shopping cart's item 2.<br>Represents shopping cart's item 3. |
| + toString() : String | Returns String "<Classname>[]". |

| Class: Customer | Description |
|---|---|
| - name: String<br>- cart: ShoppingCart | Represents customer's name.<br>Represents customer's shopping cart. |
| + Customer(String)<br><br><br><br><br>+ toString() : String | Constructor that<br>• Assigns the parameter to customer's name, and<br>• Creates a ShoppingCart object and assigns the object to the variable that represents customer's shopping cart.<br>Returns String "<Classname>[<customer name>]". |

Check your answer by invoking the `main` method in class `Tester1`, and your output should be as follows:

```
Creating Item object 1: Item[P001]

Creating Item object 2: Item[P002]

Creating Item object 3: Item[null]

Creating Item object 4: Item[null]

Creating Customer object 1: Customer[Doremi]

Creating Customer object 2: Customer[Fasola]
```

*Proceed to Stage 2 only after you have completed Stage 1 without errors.

## Stage 2:

1. Create a Java project named `Lab3Stage2`.
2. Copy class `Tester2.java` into your project.
3. Copy class `Item`, `ShoppingCart` and `Customer` in Stage 1 into your Stage 2 Java project.

Add into class `Item`, `ShoppingCart` and `Customer`:

| Class: Item | Description |
|---|---|
| - id: String | Represents item's id. |
| - name: String | Represents item's name. |
| - price: double | Represents item's price. |
| - unit: int | Represents item's unit. |
| + Item() | Constructor with no parameter and empty body. |
| + Item(String, String, double) | Constructor that assigns parameter 1 to item's id, parameter 2 to item's name, parameter 3 to item's price and 0 to item's unit. |
| + toString() : String | Returns String "<Classname>[<item id>]". |
| + setId(String): void | Assigns the parameter to item's id. |
| + getId(): String | Returns item's id. |
| + setName(String): void | Assigns the parameter to item's name. |
| + getName(): String | Returns item's name. |
| + setPrice(double): void | Assigns the parameter to item's price. |
| + getPrice(): double | Returns item's price. |
| + setUnit(int): void | Assigns the parameter to item's unit. |
| + getUnit(): int | Returns item's unit. |

| Class: ShoppingCart | Description |
|---|---|
| - item1: Item | Represents shopping cart's item 1. |
| - item2: Item | Represents shopping cart's item 2. |
| - item3: Item | Represents shopping cart's item 3. |
| + toString() : String | Returns String "<Classname>[]". |
| + getItem1(): Item | Returns shopping cart's item 1. |
| + getItem2(): Item | Returns shopping cart's item 2. |
| + getItem3(): Item | Returns shopping cart's item 3. |

| Class: Customer | Description |
|---|---|
| - name: String<br>- cart: ShoppingCart | Represents customer's name.<br>Represents customer's shopping cart. |
| + Customer(String)<br><br><br><br><br>+ toString() : String<br>+ getName(): String<br>+ getShoppingCart(): ShoppingCart | Constructor that<br>• Assigns the parameter to customer's name, and<br>• Creates a ShoppingCart object and assigns the object to the variable that represents customer's shopping cart.<br>Returns String "<Classname>[<customer name>]".<br>Returns customer's name.<br>Returns customer's shopping cart. |

Check your answer by invoking the `main` method in class `Tester2`, and your output should be as follows:

```
Creating Item object: Item[P001] >> id-P001 >> name-Pants >> price-39.99 >> unit-0

Creating Item object: Item[P002] >> id-P002 >> name-Long skirt >> price-59.99 >> unit-0

Creating Item object: Item[null] >> id-null >> name-null >> price-0.0 >> unit-0

Creating Item object: Item[null] >> id-null >> name-null >> price-0.0 >> unit-0

Creating Customer object: Customer[Doremi] >> name-Doremi >> cart-ShoppingCart[]
cart's item 1-null >> cart's item 2-null >> cart's item 3-null

Creating Customer object: Customer[Fasola] >> name-Fasola >> cart-ShoppingCart[]
cart's item 1-null >> cart's item 2-null >> cart's item 3-null
```

Points to ponder:

1. Why the 3rd and 4th `Item` objects have id-null, name-null, price-0.0 and unit-0, unlike the 1st and 2nd `Item` objects?
2. Why cart's item 1, cart's item 2 and cart's item 3 of both `Customer` objects are null?

*Proceed to Stage 3 only after you have completed Stage 2 without errors.

## Stage 3:

1. Create a Java project named `Lab3Stage3`.
2. Copy class `Tester3.java` into your project.
4. Copy class `Item`, `ShoppingCart` and `Customer` in Stage 2 into your Stage 3 Java project.

Add into class `Item`, `ShoppingCart` and `Customer`:

| Class: Item | Description |
|---|---|
| - id: String | Represents item's id. |
| - name: String | Represents item's name. |
| - price: double | Represents item's price. |
| - unit: int | Represents item's unit. |
| + Item() | Constructor with no parameter and empty body. |
| + Item(String, String, double) | Constructor that assigns parameter 1 to item's id, parameter 2 to item's name, parameter 3 to item's price and 0 to item's unit. |
| + toString() : String | Returns String "<Classname>[<item id>]". |
| + setId(String): void | Assigns the parameter to item's id. |
| + getId(): String | Returns item's id. |
| + setName(String): void | Assigns the parameter to item's name. |
| + getName(): String | Returns item's name. |
| + setPrice(double): void | Assigns the parameter to item's price. |
| + getPrice(): double | Returns item's price. |
| + setUnit(int): void | Assigns the parameter to item's unit. |
| + getUnit(): int | Returns item's unit. |
| + getItemTotal(): double | Returns the item's total price. |

| Class: ShoppingCart | Description |
|---|---|
| - item1: Item | Represents shopping cart's item 1. |
| - item2: Item | Represents shopping cart's item 2. |
| - item3: Item | Represents shopping cart's item 3. |
| + getItem1(): Item | Returns shopping cart's item 1. |
| + getItem2(): Item | Returns shopping cart's item 2. |
| + getItem3(): Item | Returns shopping cart's item 3. |
| + toString() : String | Returns String "<Classname>[]". |
| + addItem(Item, int): boolean | Checks whether cart's item 1, item 2 and item 3 are null.<br>1) If all of these items are not null that means the shopping cart is already full. So, the method will return false, to indicate that the adding of another item to cart has failed.<br>2) Otherwise, checks whether item 1 is null. If item 1 is null:<br>• Create an Item object (using the parameterless constructor in class Item) and assigns the object to the variable that represents item 1.<br>• Assigns id of Item object that has been received by the method as 1st parameter to item 1's id.<br>• Assigns name of Item object that has been received by the method as 1st parameter to item 1's name.<br>• Assigns price of Item object that has been received by the method as 1st parameter to item 1's price.<br>• Assigns integer value that has been received by the method as 2nd parameter to item 1's unit.<br>• If item 1 is not null but item 2 is null, do the above to item 2, or if item 1 and item 2 are not null but item 3 is null, do the above to item 3.<br>• Returns true to indicate that the adding of another item to cart is successful. |
| + getCartTotal(): double | Returns the total price of cart's items that are not null. |

| Class: Customer | Description |
|---|---|
| - name: String<br>- cart: ShoppingCart | Represents customer's name.<br>Represents customer's shopping cart. |
| + Customer(String)<br><br><br><br><br>+ toString() : String<br>+ getName(): String<br>+ getShoppingCart(): ShoppingCart<br>+ addItemToCart(Item, int): boolean<br>+ getTotalPurchase(): double | Constructor that<br>• Assigns the parameter to customer's name, and<br>• Creates an object of ShoppingCart and assigns the object to the variable that represents customer's shopping cart.<br>Returns String "<Classname>[<customer name>]".<br>Returns customer's name.<br>Returns customer's shopping cart.<br>Asks customer's cart to add an item.<br>Asks customer's cart to return total price of cart's items. |

Check your answer by invoking the `main` method in class `Tester3`, and your output should be as follows:

```
Doremi adds 2 Pants into the shopping cart.
Doremi adds 2 Shirt into the shopping cart.
Doremi adds 3 T-shirt into the shopping cart.
Doremi adds 1 Long skirt into the shopping cart. Error - the shopping cart is full
Fasola adds 2 T-shirt into the shopping cart.
Fasola adds 1 Pants into the shopping cart.

Charging Doremi for:

Item ID: P001
Item name: Pants
Price: RM 39.99
Unit: 2
Sub total: RM 79.98

Item ID: P003
Item name: Shirt
Price: RM 69.99
Unit: 2
Sub total: RM 139.98

Item ID: P004
Item name: T-shirt
Price: RM 19.99
Unit: 3
Sub total: RM 59.97

Total: RM 279.92999999999995


Charging Fasola for:
```

```
Item ID: P004
Item name: T-shirt
Price: RM 19.99
Unit: 2
Sub total: RM 39.98

Item ID: P001
Item name: Pants
Price: RM 39.99
Unit: 1
Sub total: RM 39.99

Total: RM 79.97
```

Points to ponder:

1. Why does the total price for Doremi printed as `RM 279.92999999999995`?
2. How do you output the amount in 2 decimal places?