

Task 1: Online Store

In this lab task, you will create an online store using object-oriented programming concepts. The system will allow users to view products, add them to a shopping cart, and checkout.

Requirements

(you can use built in list data structures -arraylist or linked list)

1. Create a Product class with the following properties:
 - a. id (int): a unique identifier for the product
 - b. name (String): the name of the product
 - c. description (String): a brief description of the product
 - d. price (double): the price of the product
 - e. quantity (int): the quantity of the product in stock
2. Create a Customer class with the following properties:
 - a. id (int): a unique identifier for the customer
 - b. name (String): the name of the customer
 - c. address (String): the address of the customer
3. Create an Order class with the following properties:
 - a. id (int): a unique identifier for the order
 - b. customer (Customer): the customer who placed the order
 - c. items (List<Product>): the list of products in the order
 - d. total (double): the total price of the order

Implement the following methods:

- a. addItem(item: Product, quantity: int): void
 - b. removeItem(item: Product, quantity: int): void
 - c. getTotal(): double -compute the total at checkout
 - d. getItems(): List<Product>
4. Create a Store class with the following properties:
 - a. A list to store all the products available at the storeImplement the following methods:
 - a. viewProducts(): displays a list of all available products
 - b. addProduct(product: Product): add a new product to the store
 - c. removeProduct(product: Product): remove an existing product from those available at the store
5. Create a main method to test your classes

Task 2: FeetInches Class

The class has two int fields: feet and inches. Together these fields hold a distance measured in feet and inches, such as 12 feet 7 inches. The feet field holds the feet part, and the inches field holds the inches part.

Method Description

Constructor #1 This constructor assigns 0 to both the feet and inches fields. (no-arg constructor)

Constructor #2 This constructor accepts two int arguments that are assigned to the feet and inches fields. The simplify method is also called.

Constructor #3 This constructor accepts a FeetInches object as its argument. The object that is being created will become a copy of the object passed as an argument. This type of constructor is sometimes referred to as a “copy constructor.”

simplify This method adjusts any set of values where the inches field is greater than 11. For example, 3 feet 14 inches would be adjusted to read 4 feet 2 inches.

setFeet This method accepts an int argument that is assigned to the feet field.

setInches This method accepts an int argument that is assigned to the inches field. The simplify method is then called.

getFeet This method returns the value in the feet field.

getInches This method returns the value in the inches field.

toString This method returns a string representing the distance held by the object. For example, if an object's feet field holds 3 and its inches field holds 7, the toString method would return the string “3 feet 7 inches”.

add This method accepts a FeetInches object as its argument. It returns a reference to a FeetInches object that is the sum of the calling object and the object that was passed as the argument.

equals This method accepts a FeetInches object as its argument. It returns the boolean value true if the calling object and the argument object hold the same data. Otherwise it returns false.

Also create a main class which has object of FeetInches class and call its method.