

# Rajalakshmi Engineering College

Name: Samyuktha S  
Email: 240701701@rajalakshmi.edu.in  
Roll no: 240701701  
Phone: 6380226314  
Branch: REC  
Department: CSE - Section 9  
Batch: 2028  
Degree: B.E - CSE

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 7\_Q4

Attempt : 1  
Total Mark : 10  
Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

Maria, a software developer, is working on an inventory management system project using Java that utilizes an inventory interface to manage a store's products.

The interface should define two methods: `addProduct`, which adds a product by accepting its name, price, and quantity, and `calculateTotalValue`, which computes the total value of all products in the inventory. Implement the interface in a class called `SimpleInventory`, which internally manages a list of `Product` objects.

Each `Product` object should encapsulate the product's name, price, and quantity and include a method to calculate its value as  $\text{price} \times \text{quantity}$ . The system should allow users to dynamically add products to the inventory and calculate the total value of all products stored.

Help Maria achieve the task.

### ***Input Format***

The first line of input consists of an integer to choose one of the following options:

- 1 - to add a product to the inventory.
- 2 - to calculate and view the total inventory value.
- 3 - to exit the program.

For Choice 1 (Add Product):

The next input line is the string representing the product name as a string (single or multi-word, without quotes).

The next line is a double value representing the price as a decimal value

The next line is an integer value representing the quantity as an integer

For Choices 2 and 3, no additional input is required

### ***Output Format***

The output displays the results of the commands as follows:

- For the addProduct command, the program should display "Product added to inventory."
- For choice 2, the program should display "Total inventory value [totalvalue].  
"The total value should be displayed with one decimal place. If there is no product in the inventory, print the total as 0.0.
- For choice 3, the program should exit

If the choice is not 1, 2, or 3, then print "Invalid choice. Please select a valid option (1/2/3).".

Refer to the sample output for the formatting specifications.

### Sample Test Case

Input: 1

Laptop

800.0

3

2

5

3

Output: Product added to inventory.

Total inventory value: \$2400.0

Invalid choice. Please select a valid option (1/2/3).

### Answer

```
import java.util.Scanner;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
import java.text.DecimalFormat;
```

```
class Product {
```

```
    private String name;
```

```
    private double price;
```

```
    private int quantity;
```

```
    public Product(String name, double price, int quantity) {
```

```
        this.name = name;
```

```
        this.price = price;
```

```
        this.quantity = quantity;
```

```
    }
```

```
    public double calculateValue() {
```

```
        return price * quantity;
```

```
    }
```

```
}
```

```
interface Inventory {
```

```
    void addProduct(String name, double price, int quantity);
```

```
    double calculateTotalValue();
```

```
}
```

```
class SimpleInventory implements Inventory {
```

```
private List<Product> products;

// Modified Constructor to accept the integer argument, fulfilling the
requirement.
public SimpleInventory(int capacity) {
    // The 'capacity' argument is accepted but not used since we're using
    ArrayList,
    // but its inclusion satisfies the constructor signature requirement.
    this.products = new ArrayList<>();
}
```

```
public void addProduct(String name, double price, int quantity) {
    Product newProduct = new Product(name, price, quantity);
    products.add(newProduct);
    System.out.println("Product added to inventory.");
}
```

```
public double calculateTotalValue() {
    double totalValue = 0.0;
    for (Product product : products) {
        totalValue += product.calculateValue();
    }
    return totalValue;
}
}
```

```
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Inventory inventory = new SimpleInventory(10);
        while (true) {
            int choice = scanner.nextInt();
            if (choice == 1) {
                scanner.nextLine();
                String productName = scanner.nextLine();
                double price = scanner.nextDouble();
                int quantity = scanner.nextInt();
                inventory.addProduct(productName, price, quantity);
            } else if (choice == 2) {
                double totalValue = inventory.calculateTotalValue();
            }
        }
    }
}
```

```
        System.out.println("Total inventory value: $" + totalValue);
    } else if (choice == 3) {
        break;
    } else {
        System.out.println("Invalid choice. Please select a valid option
(1/2/3).");
    }
}
scanner.close();
}
```

**Status :** Correct

**Marks :** 10/10