Project 3

Let's proceed with the next steps for enhancing the inventory project.

Step 1: Modify `ProductTester` Class to Include User Input and Exception Handling

1. **Ask the user to enter the number of products they wish to add**:
   - We will use a `do-while` loop to ensure the user inputs a valid positive integer or zero.
   - We'll handle exceptions for incorrect data type entries.

2. **Handle multiple products using an array**:
   - If the user enters zero, display a message and end the program.
   - Otherwise, create an array of `Product` objects and populate it using user inputs.

Modified `ProductTester` Class

```java
import java.util.Scanner;

public class ProductTester {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int maxSize = -1;  // Initial value to fail the loop initially

        // Prompt user for number of products
        do {
            try {
                System.out.print("Enter the number of products you would like to add (Enter 0 if you do not wish to add products): ");
                maxSize = in.nextInt();
                if (maxSize < 0) {
```

```
            System.out.println("Incorrect Value entered");
        }
    } catch (Exception e) {
        System.out.println("Incorrect data type entered! Please enter a valid integer.");
        in.nextLine(); // Clear the input buffer
    }
} while (maxSize < 0);

if (maxSize == 0) {
    System.out.println("No products required!");
} else {
    Product[] products = new Product[maxSize];
    int tempNumber;
    String tempName;
    int tempQty;
    double tempPrice;

    for (int i = 0; i < maxSize; i++) {
        in.nextLine(); // Clear the input buffer
        System.out.println("Enter the details for Product " + (i + 1) + ":");
        System.out.print("Item Number: ");
        tempNumber = in.nextInt();
        in.nextLine(); // Consume newline left-over
        System.out.print("Name: ");
        tempName = in.nextLine();
        System.out.print("Quantity: ");
        tempQty = in.nextInt();
        System.out.print("Price: ");
        tempPrice = in.nextDouble();

        // Create a new product and add it to the array
```

```
                products[i] = new Product(tempNumber, tempName, tempQty, tempPrice);
            }

            // Display the details of each product using a for-each loop
            System.out.println("\nProduct details:");
            for (Product product : products) {
                System.out.println(product.toString());
            }
        }

        // Closing the Scanner
        in.close();
    }
}
```

Updates to `Product` Class

No changes are needed to the `Product` class as it already meets the requirements for this part of the project.

Step 8: Save Your Project

Ensure all changes are saved in your IDE or text editor.

This completes the required updates for Section 6 of the inventory project. Run and test the program to ensure it works as expected. If you have any issues or need further assistance, feel free to ask!