Post-Assessment Reflection

Java Console App Inventory Manager Project

Name: Marquita Scotton

Date: 07/12/2025

Instructor: Eric Wise

Purpose

This reflection helps you analyze your learning experience, identify strengths, and set goals for improvement in Java programming. Be honest and detailed in your responses—this will help you grow as a developer.

Section 1: Self-Assessment of Project Performance

1. How would you rate your overall performance on this project?

(Select one)

⬜ Excellent – I met or exceeded all requirements with confidence.

⬜ Good – I completed most requirements successfully with some challenges.

☑ Satisfactory – I met the basic requirements but struggled in some areas.

⬜ Needs Improvement – I had difficulty completing key components.

2. Which parts of the project were the most challenging for you? Why?

(Explain any difficulties you faced, such as coding logic, debugging, understanding OOP concepts, or managing time.)

The most challenging part was understanding how to use inheritance and method overriding in Java. It took time to figure out how to make PerishableProduct work properly with the Product class. I also struggled with file handling—especially saving and loading data with different product types. Debugging those issues was tricky. Lastly, managing my time between coding, testing, and documentation was a bit overwhelming at first.

3. Which parts of the project were the easiest or most enjoyable for you? Why?

The most enjoyable part was building the main menu and getting the user input to work. It felt good to see the program respond to different choices and actions. Adding products and displaying them ~~was also pretty easy for me because it was straightforward and helped me see my pro~~gress quickly. I liked testing the features and watching it all come together.

Section 2: Learning Outcomes Reflection

4. How well do you understand the following Java concepts after completing this project?

(Rate yourself on a scale of 1 to 5, with 1 being "Not confident" and 5 being "Very confident.")

Concept	Rating (1-5)	Comments (What helped you learn? What still confuses you?)
Object-Oriented Programming (OOP)	4	
Inheritance & Polymorphism	4	
Exception Handling & Validation	4	
File I/O & Data Persistence	3	
Console-Based User Interfaces	3	

5. If you had to redo this project, what would you do differently? Why?

If I had to redo this project, I would plan my time better and start testing earlier. I would also try to organize my code more from the beginning by setting up packages and classes right away.

Section 3: Growth & Future Goals

6. What skills or concepts from this project do you think will be useful in future coding projects?

I learned how to use object-oriented programming, especially classes, inheritance, and method overriding, which will help in any future Java project. I also gained experience with file handling and user input. Writing unit tests with JUnit helped me understand how to test my code properly.

7. What is one specific goal you have for improving your programming skills after this project?

(Example: "I want to improve my debugging skills by practicing error handling in smaller programs.")

One specific goal I have is to get better at writing cleaner, more organized code by using design patterns and following best practices. I want to improve how I structure projects from the start so they're easier to maintain and expand later.

8. What additional support or resources would help you become a stronger Java programmer?

(Example: More practice problems, additional tutorials, one-on-one mentoring, etc.)

Watching video tutorials that break down Java concepts with real coding examples would really help. Step-by-step walkthroughs and hands-on tutorials, especially for advanced topics like inheritance or file handling, would make it easier to understand and apply what I'm learning

Final Thoughts

9. What advice would you give to future students working on this project?

(What strategies worked well for you? What would you warn them about?)

Start early and take it one step at a time. Test your code often so you catch mistakes early. Keep your files organized and comment your code to stay on track. Don't forget to save your work regularly! Also, be ready to spend extra time on file saving and loading.

10. Any additional comments or feedback about this project?

n/a

Submission Instructions

- **Due Date: _____**
- **Submission Format:** Submit a **typed or handwritten** copy to the instructor.
- **Grading:** This reflection is **not graded** but is required for **course completion** and personal growth.

Instructor Use Only (Optional Feedback)

(Instructor may provide additional feedback based on reflection responses.)

This reflection helps you **process what you've learned** and prepares you for **future programming challenges**. Great job on completing the project!