## Isabella Gaytan

## October 2, 2022

CS 3331 – Advanced Object-Oriented Programming – Fall 2022

## Dr. Daniel Mejia

## PA2

This work was done individually and completely on my own. I did not share, reproduce, or alter any part of this assignment for any purpose. I did not share code, upload this assignment online in any form, or view/received/modified code written from anyone else. All deliverables were produced entirely on my own. This assignment is part of an academic course at The University of Texas at El Paso and a grade will be assigned for the work I produced.

NOTE: Please write in complete sentences (paragraph form). Check grammar, punctuation, and ensure your writing is clear. Write enough to make sure you clearly explain each part of the question.

# **Program Explanation**

In this section, explain the overview of the assignment.

What did you do?

* For this assignment, I built on my previous work from PA0 and PA1 to include a design pattern, this being the VenueFactory class and the EventFactory class. Additionally, I made sure to verify that all my functionality from PA1 was optimized. For instance, instead of having all my functions in the main method, I separated them into actual methods also stored in the RunTicket class. Moreover, for PA2, I updated user interaction functionality by implementing the system administrator inquiries. Before, my program was targeted towards customers only, now, my program asks the user whether they are a customer or a system administrator. Each input prompts different functionalities.

PA3:

* For this assignment, I built on previous work from PA2 to handle receiving an input file that does not have the information listed in standard column orders as previous assignments. I also added a Singleton Design Pattern titled “Logger” to handle adding each action taken by the user to the TicketMinerLog.txt. Additionally, I also added a new ticket purchasing functionality and data information maintenance: I included the Texas Sales Tax into the charged price with the salesTaxPrint() method in which I also check if the user is a ticketminer member and if so, give them a 10% discount. The salesTaxPrint method also keeps track of the total amount the user has saved with their discount and the total amount of discounts that have been applied to a specific event.

PA4:

* For this assignment, I built on previous work from PA3 to include new event admin functionality, automatic purchasing functionality, creating an electronic ticket summary, and more user interaction. These functionalities required me to review my previous work and feedback for PA3, and also consider how to handle new exceptions. I created three new classes: Date, Autopurchasing, and UserInterface. The Date class handles the user input for the newly created event date, the autopurchasing class takes care of performing the actions stated in the csv files, and the user interface class handles working with the user as a customer.

How did you tackle the problem?

* I tackled the problem by first reading the feedback I got on the previous assignment which mainly focused on optimizing my program by separating my functionalities into methods. I also included header in my classes since they weren’t present before. Additionally, it was helpful to attend Ali’s office hours regarding any questions that came up regarding the assignment.

PA3:

* I tackled the problem by first reading the feedback I got on the previous programming assignment. Then, I realized that the functionalities I was supposed to include in this assignment could simply be added through a method.

PA4:

* I tackled the problem by setting the time to think about how to implement the new functionalities and how they would work with my existing code. It was helpful to ask questions during class and office hours. Additionally, I also took notes as to facilitate my implementations since it is getting hard to keep track of my code as it keeps growing larger.

What techniques did you use to solve the problem?

* To solve the problem, I used my Factory Design Pattern notes from class in order to implement the factories into my program. I also focused on creating the menu for the system administrator in a similar way as that of the menu for the customer. I used methods, for loops, design patterns, and functions within the Java program to aid me in solving the problem and applying the required functionalities.

PA3:

* To solve the problem, I attended office hours to clear up any concerns or questions I had regarding the instructions which saved me time on this assignment. I focused on implementing what was asked of me in a way that is maintainable and therefore I found it useful to simple create new methods and call them from the main. I had to add new attributes to my code such as moneySaved which represents the amount of money the customer saved if they are a ticketminer member and amountDiscounted which represents the amount of money that has been discounted for a specific event.

PA4:

* To solve the problem, I attended office hours to clear up any concerns or questions I had. For instance, I was not sure whether we were allowed to hardcode the venue list displayed to the admin or whether we were expected to read the information from the file and from there display it to the admin. Ali was able to clear it up for me. I also created more classes to facilitate my code maintenance.

Did you break the problem into smaller problems? Explain.

* I broke my code into smaller problems for PA2 by separating my functionalities into methods and starting from the smallest problem and then building on from there. For instance, I went to the smallest section of my code in the main method which was basically the part where I construct the ticket object once the user has successfully processed the transaction. Then, I asked myself what needs to happen in my code before this is implemented and created another method off that and so on.

PA3:

* I broke my code into smaller problems for PA3 by separating the functionalities into methods and seeing what I could handle in each method. For instance, although I have a method named “salesTaxPrint” it handles more than the sales tax as it also handles applying discount from the price if the user is a ticketminer member. I also found it simple to handle the event file being randomized since my code was easy to maintain. I simply had to create a variable data structure to hold the title of the event info and use that to construct my event objects.

PA4:

* I broke my code into smaller problems for PA4 by creating methods or classes for almost all new functionalities implemented. This made it easier to access the functionalities whenever they were needed. I also separated the autopurchasing functionality from the user interface functionality that was I could differentiate the two. I believe that will be helpful for the next assignments.

# **What did I learn?**

What did you learn as a result of this assignment?

* As a result of this assignment, I learned how much cleaner my code looks once I have applied the factory design pattern. In PA1, I had a lot of repetitive work being done over and over again and I was able to minimize my lines of code and optimize my code’s readability by applying separate methods to handle certain functionalities. I also keep learning how important it is to start early since I catch myself turning in assignments late.

PA3:

* As a result of this assignment, I learned how to handle a file that doesn’t act the way we expect it to, in this case, the file having the information in random order. I also learned how to apply a Singleton Design Pattern to my advantage.

PA4:

* As a result of this assignment, I learned how useful it was to start early, and to ask questions. I also learned that it is simpler to separate functionalities into different classes. I also learned the importance of maintenance and adding comments to my code.

How can my solution be improved?

* My solution can definitely be improved by adding more functions to handle some exceptions such as the user entering a number that is out of range whether that be for the menu options or for the event IDs. I also know that I have some repetitive work still especially in my event classes. I need to improve on how I apply values to the objects I am using.

PA3:

* My solution could probably be improved if I took the time to analyze all aspects of my code to see if anything can be written with more space or with more than one line in order to improve readability since I feel like that might be the thing most wrong with my code at the moment. I also think that I may still have some repetitive code that could be optimized.

PA4:

* My solution could be improved by reducing the repetitiveness in my code, although I believe that my code works efficiently, I believe that there may be a way to reduce the amount of repetitiveness throughout it. I will more than likely ask during office hours.

What ideas do I have about another way to solve the problem?

* I believe that maybe using a different data structure can help me improve my knowledge in the different ways this problem can be solved. As mentioned, I believe that creating a separate method to handle exceptions will clean up my code since a lot of the methods end up using the try and catch to catch the same exception.

PA3:

* I believe that using a different data structure can help me improve my knowledge in the different ways this problem can be solved. As mentioned, I believe that creating a separate method to handle exceptions will clean up my code since a lot of the methods end up using the try and catch to catch the same exception.

PA4:

* I believe that I could’ve implemented the autopurchasing functionality with the user interface since they handle almost all the same things, however, in my head it made more sense to separate them in case PA5 required functionalities that are to be applied only in one of the two. However, separating them into two classes makes the code look a bit repetitive.

How long did it take me to complete this lab assignment?

* Completing this assignment took me a long time, not necessarily because I struggled to find a solution, but simply because I have other responsibilities that take up time out of my week such as 4 other courses to consider coursework for, being an instructional assistant for data structures, being an advocate for CAHSI, and preparing for the GMiS conference. Therefore, this assignment took me three extra days to complete.

PA3:

* Completing this assignment did not take me long at all. I simply was prioritizing other work above it which is why I am submitting it on the due date. However, if I had to be specific about the amount of time it took me to improve my code and implement the functionalities specified in the instructions, then I would say it took me about 4 hours. I also had an extension for this programing assignment since I attended the GMiS conference.

PA4:

* Completing this assignment did not take me as long as I expected it to. I worked on the assignment about 3 days total and only for a few hours each day. If anything, I feel like it took me longer than it should have now that I finished it since I would easily get distracted in between working on my code.

# **Solution Design**

What did I do in this program?

* In this program, I implemented more user interaction by adding the system administrator menu. I also created an updated event list csv file with the updated values for event information such as tickets sold and revenue for each ticket type. Additionally, I also created an updated customer list csv file adding the amount of tickets they have purchased and the number of transactions they applied during their login session.

PA3:

* In this program, I implemented more calculation functionalities since I had to consider the Texas Sales Tax and membership discount when calculating the total price of the transaction. I also implemented the Singleton Design Pattern to take care of logging the actions taken by the user into the created TicketMinerLog txt file. Moreover, I also handled a file with randomized order of information.

PA4:

* In this program, I implemented more user interaction and new functionalities for the admin. I also created new classes and methods. Moreover, I also edited attributes in existing classes such as the Customer class where I added extra attributes to aid me in performing certain functionalities such as creating the electronic ticket summary.

What was my approach to solving this problem?

* My approach was to first implement the factory design pattern in order to clean up my ReadFile class since it has a lot of repetitive work that could’ve been handled in fewer lines. Then, I focused on cleaning up my main method by creating separate methods to handle specific functionalities. Then I implemented the admin menu and inquiries which required me to make changes and updates to values such as tickets sold and such. During these processes, I made sure to run my code to check that everything was working properly. Finally, I made sure my code handled all possible exceptions that could be prompted by the user interaction.

PA3:

* My approach was to first deal with handling the event file with the randomized order in order to get the hardest part out the way and then follow by implementing everything else. After doing this, I implemented the rest of the functionalities. During these processes, I made sure to run my code to check that everything was working properly. Finally, I made sure my code handled all possible exceptions that could be prompted by the user interaction.

PA4:

* My approach to solving the overall assignment problems was to begin with the user interaction regarding the create new event functionality. Then, I asked Ali for recommendations on how to implement the autopurchasing functionality. Then, after I implemented the autopurchasing, I went ahead and created the methods necessary to create an elecetronic ticket summary text file. I also made sure to run my code in between in order to catch any errors before moving on to the next step.

What data structures did I use? Why?

* I focused on using hashmaps since I believed it was the most efficient way to access an event and its information. After applying it, I got comfortable using it and therefore improved my applications of it. Having keys and values was extremely helpful for this assignment considering that the user chooses a key and I am expected to return information which is stored in the value. Therefore, using the “get” function for hashmaps was helpful to get information about certain objects.

PA3:

* For PA3, I focused on using hashmaps since I believed it was the most efficient way to access an event and its information. After applying it, I got comfortable using it and therefore improved my applications of it. Having keys and values was extremely helpful for this assignment considering that the user chooses a key and I am expected to return information which is stored in the value. Therefore, using the “get” function for hashmaps was helpful to get information about certain objects. Additionally, I implemented an ArrayList data structure to store all user actions and once the program executes, I traverse through that list to add to the text file.

PA4:

* For PA4, I focused on using hashmaps like I have been doing with the past programming assignments. I believe that it has been the most helpful data structure to implement the required functionalities. For instance, I used it to create a list of events purchased where I store the ticket object as the key and the event object as the value. This helped me to create the electronic ticket summary.

What assumptions, if any, did I make?

* I made the assumption that this assignment wouldn’t take me long based off the pdf with the instructions. This is because it did not seem that different from the previous assignment yet it took me longer that I expected it to. Moreover, I thought that the tickets sold could be derived from the input I originally sent to the constructor which was true but I did not think that I would have to convert the percentage of tickets sold back to whole numbers in order to have that in the updated event file. I also assumed that creating a new csv file would be harder than it actually was.

PA3:

* For PA3, I made the assumption that the assignment would not take me long and I think for once I was actually right. Yet I am still submitting it on the deadline date.

PA4:

* I assumed that this assignment would take me longer than expected judging by the size of the instruction sheet and how overwhelming it felt to hear that we had to implement autopurchasing for over 100k transactions. However, as I broke down my code I realized that it was actually simpler than I originally thought.

# **Testing**

How did I test my program?

* I tested my program as I was creating and changing my methods in order to ensure that I would catch a problem right away if any came up. Additionally, I made sure to test unexpected inputs such as a string when an integer was expected.

Did I use black-box, white-box testing, or both? Why?

* I used white box testing the most since VS code allows me to look at the terminal output at the same time as my code. For instance, whenever I needed to catch an exception, I made sure to print that exception to see exactly why I got it.

Did I test my solution enough? How can my testing practices be improved?

* I believe I tested my solution a fair amount. I do believe that I could’ve applied more test cases and I could’ve tried to break my code more but considering the time it took me to simply implement the solution and how late my assignment is, as long as it worked, I decided it was time to submit it. My testing practices can be improved by applying test cases each time I add or change something. I did test my code a couple of times, but I feel like there are exceptions I might’ve missed.

What are the test cases I used?

* Test cases I used included inputting a string when an integer is expected, inputting an integer that is out of the range expected, or not following the prompt that is expected from the user to begin with. These test cases helped me improve my code by noticing where I needed to catch such exceptions and implement a similar design throughout the sections that required so.

Did I break my program and use that to improve it?

* I tried to break my program as much as possible within the time I had left to submit. This definitely helped me improve my code to avoid the program to terminate if anything wasn’t handled properly. As mentioned, I do believe that I could’ve tried breaking my code further to improve my implementation and solution however I now feel rushed to submit this as soon as possible.

# **Test results**

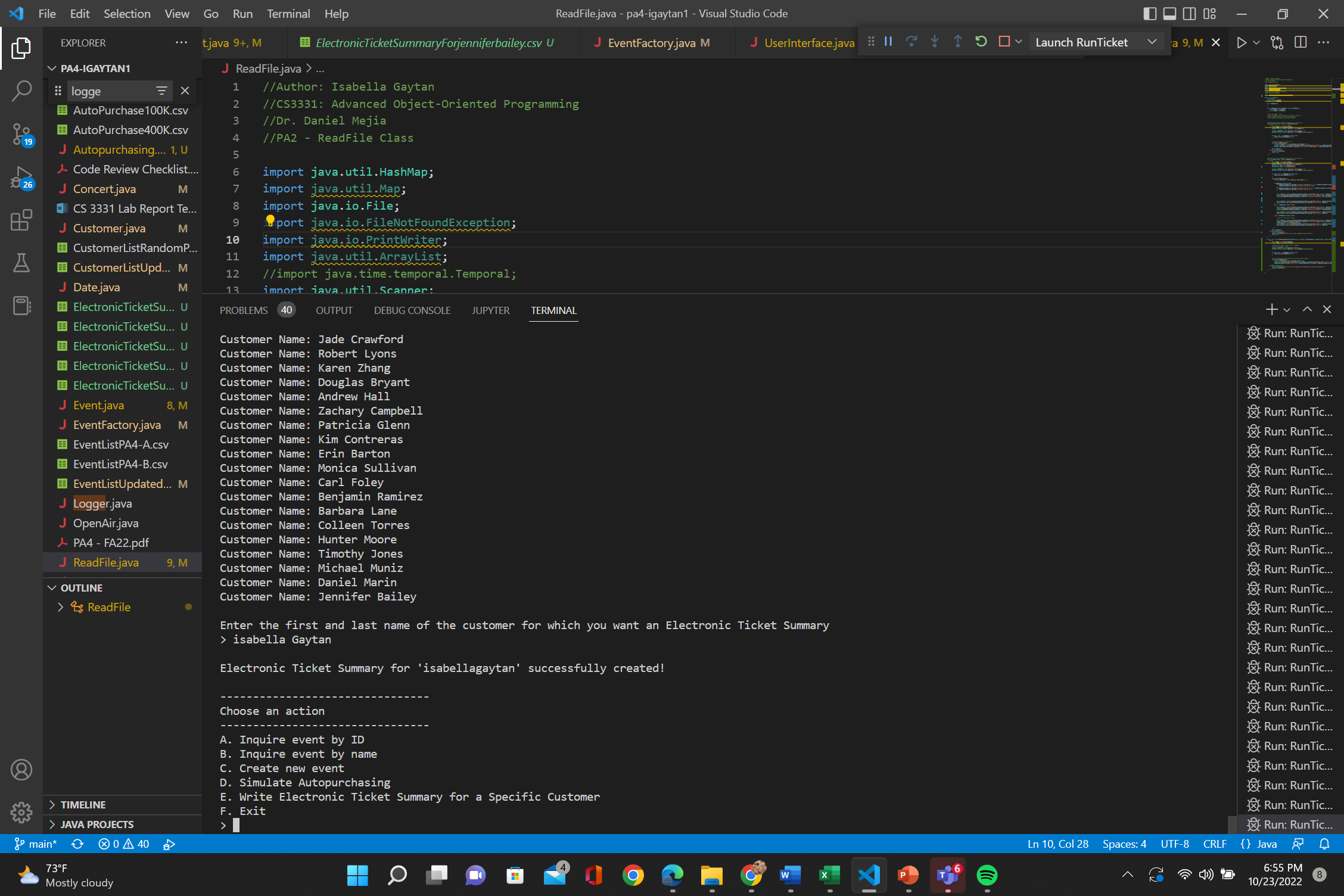
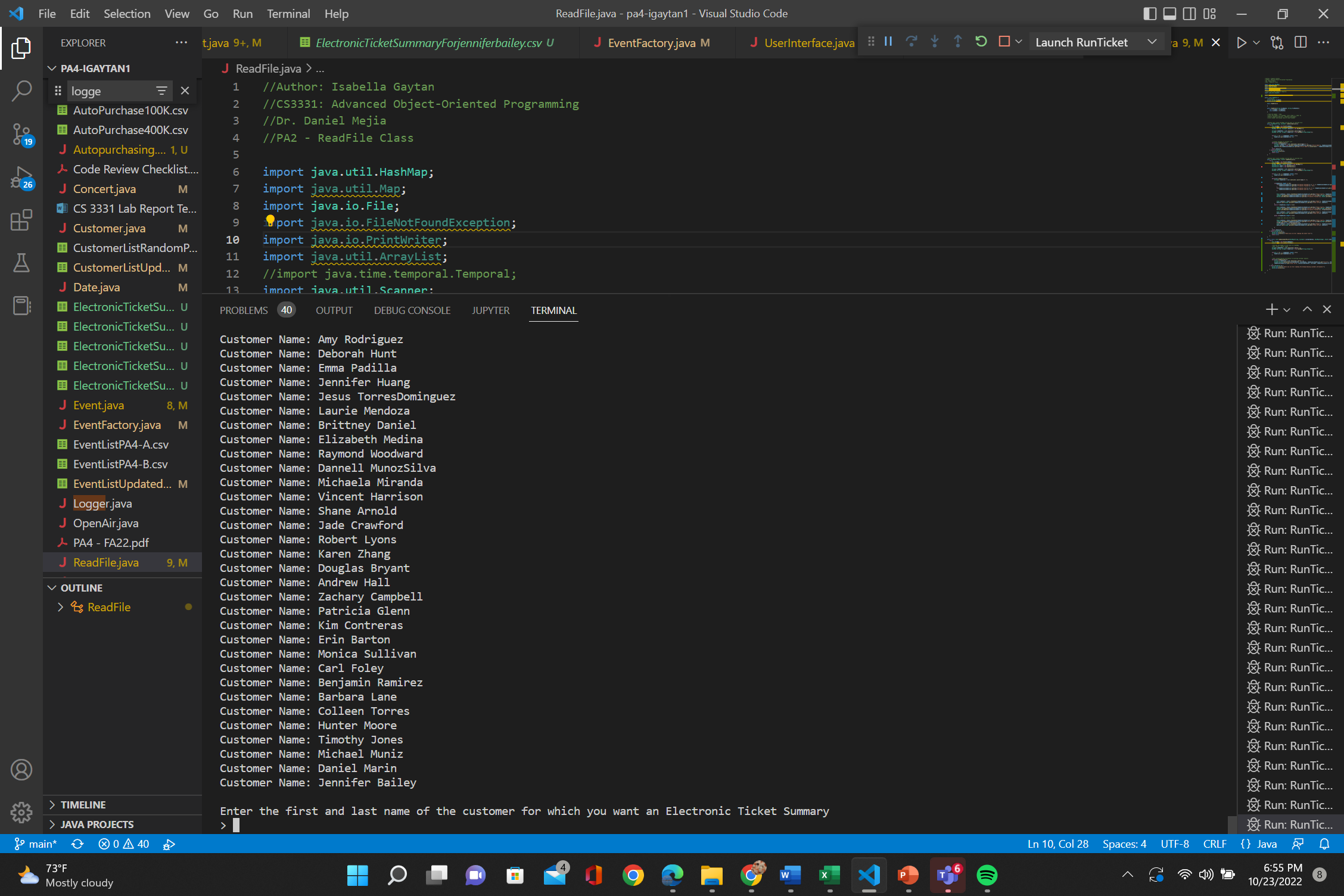
Describe the results of your tests.

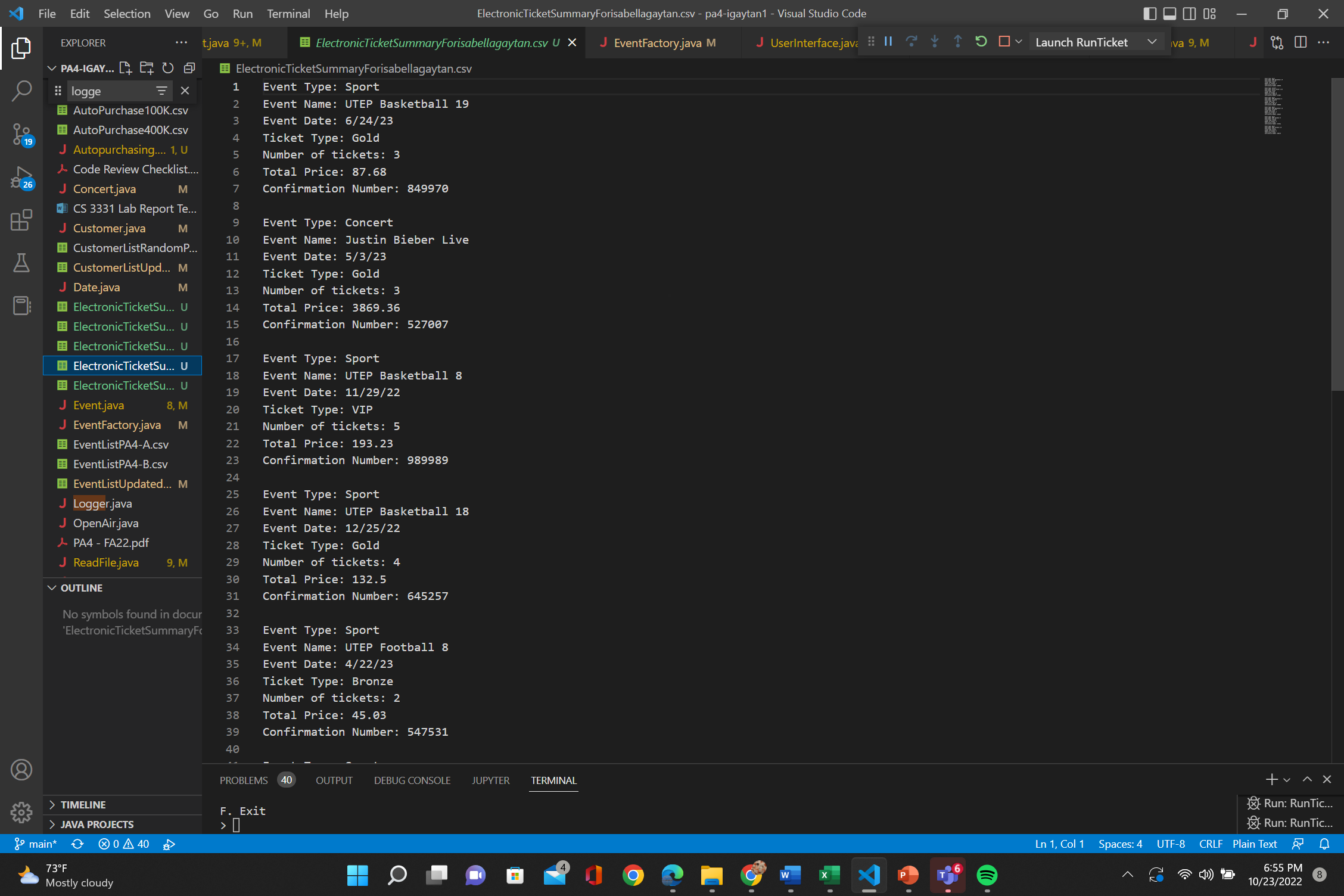
* Some of my test cases where handled appropriately with the implementations I already had, however, I ran into some problems with certain test cases. For instance, when creating the electronic ticket summary, I got a null pointer exception error for the hashmap I created storing the list of tickets purchased for the customer, however, I realized it was actually because when I did not initialize the hashmap correctly.

Include any console outputs showing your results.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated



Include any text document results of your tests.

# **Code Review**

Explain how you conducted a review of your code. Describe how you checked each part of the code review checklist.

Implementation

• Does this code do what it is supposed to?

* Yes my code implements all the new functionality expected from the instructions provided.

• Can it be simplified?

* Most definitely. My code can be simplified if I were to add functions to handle errors that are repetitive such as input mismatch.

• Is the code dynamic or hardcoded?

* I believe my code is not hardcoded as I have implemented solutions that are dynamic in nature.

• Is the code maintainable?

* After separating my long main method into separate methods, I have made my code more maintainable.

Logic

• Cases where code does not behave as expected/intended?

* I think all the test cases I applied led to the code behaving as it is expected to.

• Test cases where it may fail?

* All the test cases I applied worked, however, I do not feel confident that I broke my code to its full extent, it is still not perfect.

Readability/Style

• Easy to read/understand?

* After cleaning up my code, I believe it has become more readable.

• What parts can be modified or adjusted?

* I believe I might’ve applied the use of objects and their classes in an inefficient way since I have methods withing those classes that have their own information as parameters.

• Is the structure appropriate?

* I believe the structure is appropriate

• Does it follow the appropriate language style?

* My code follows the appropriate java language style

• Is the code well documented?

* My code includes comments and headers all throughout.

Performance

• What is the code complexity?

* The complexity of the code heavily depends on the size(n) of the csv files provided since I am creating a event/customer object for each line in file O(n).

• How does the complexity change with various inputs?

* The inputs for this programming assignment are the csv files and they are a great factor in determining the time complexity of the program. The larger the csv file, the larger complexity of the program.