

**The University of Texas at El Paso**  
**Department of Computer Science**  
**CS 3331 – Advanced Object-Oriented Programming**  
**Instructor: Daniel Mejia**  
**Fall 2023**

**Programming Assignment 3**

**Academic Integrity Statement:**

This work is to be done in pairs (2-people). It is not permitted to share, reproduce, or alter any part of this assignment for any purpose. Students are not permitted from sharing code, uploading this assignment online in any form, viewing, receiving, or modifying code written from anyone else outside of the team. 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 produced individually by the student.

**Objective:**

Utilize Object-Oriented programming design and principles to create a system.

**Instructions:**

This assignment is to be done in pairs (2 people). Your code must be written in Java. You must submit your assignment through GitHub Classroom. In the comment heading of your source code, you should write your name, date, course, instructor, programming assignment 3, lab description, and honesty statement. The honesty statement must state that you completed this work entirely on your own (i.e., as a team) without any outside sources including non-team members, experts, online sources, or the like. Only assistance from the instructor, TA, or IA will be permitted.

**Scenario:**

You have recently been hired to work for *TicketMiner*, a company that sells tickets for sporting events, concerts, special events, etc. You have a few customers that are interested in creating their events using your system.

### Part A:

Read the requirements described in Part B to complete Part A. Part A should be completed before Part B

1. Write a refactored UML Class Diagram to structure your code using the classes, requirements, and concepts described in Part B
2. Write a level II UML Use Case Diagram based on Part B
  - a. 3 Use Cases
  - b. 2 Actors
  - c. 1 extend
  - d. 2 include

### Part B

1. Conduct a code review on your teammates code
  - a. Understand what they did
  - b. Understand their approach
  - c. Ask questions
  - d. Give constructive feedback
  - e. Discover and communicate the best way to incorporate their code
2. Combine code from all teammates into a single program
  - a. Do not simply use one person's entire code
  - b. Using Javadoc – write where each piece of code came from (i.e. Taken from Bob, Taken from Alice, Taken from Samantha)
3. Refactor the code
  - a. The code should handle all functionality from Programming Assignment 3 (PA3)
  - b. Fix anything that should be corrected
    - i. Appropriate data structures
    - ii. Appropriate use of objects
    - iii. Relationships between objects
    - iv. Algorithms and complexity
4. Handle receiving an input file that does not have the information listed in standard column orders as previous assignments
  - a. Consider creating a class that will handle reading a file
5. Use a design pattern as part of your refactoring process
  - a. You should use at least 2 design patterns in your system
  - b. Select one of the design patterns discussed in class or use one that you have researched on your own
  - c. Discuss its use in the lab report
6. Add new ticket purchasing functionality and data information maintenance
  - a. Include Texas Sales Tax into the charged price
    - i. Texas sales tax is 8.25% of the subtotal

1. If the subtotal price of tickets is \$100, then sales tax is \$8.25 and total price charged to the customer is \$108.25 (round down to the nearest cent, if appropriate)
  2. Tax is not profit and should be maintained and accounted for separately
- b. Add functionality to print the total tax charged per event
- c. If the customer is a “TicketMiner Member”, they should get a 10% discount on the subtotal
  - i. If the total price of tickets is \$100, then the new subtotal should be \$90. The subtotal is taxed 8.25% and then the customer should be charged \$97.42 (round down to the nearest cent, if appropriate)
  - ii. Keep track of the total amount each customer has saved (in this case \$10 plus any other concert purchases)
    1. It may be useful to have each customer keep track of their savings
  - iii. Keep track of the total amount the event has discounted (in this case \$10 plus other member purchases for this event)
7. Write a lab report describing your work (template provided)
  - a. Any assumptions made should be precisely commented in the source code and described in the lab report
  - b. Lab report should contain sample screenshots of the program being run in different circumstances including successful and failing changes
8. Complete an individual code review on your code (template provided)
9. Schedule a demo with the TA/IA
10. **\*\*If submission is past the deadline\*\*** Your report must have an additional section entitled “Why I submitted late”. In that section, explain the reason why your submission was late. (Note: you will still be penalized the typical late penalty)

### **Deadlines:**

October 17, 2023, by 11:59pm (Current Progress Commit) – GitHub classroom:

1. UML Class Diagram Progress (.pdf)
2. UML Use Case Diagram Progress (.pdf)
3. Current Progress Source Code (.java) – Commit current progress up to this point

For each item (1-3)

- a. Does not have to be complete
- b. Should be a significant amount of work done (as determined by instructional team)
- c. TA/IA will review for progress only

October 22, 2023, by 11:59pm - GitHub Classroom:

1. UML Class Diagram (.pdf)
2. UML Use Case Diagram (.pdf)
3. Lab report (.pdf file)
4. Source code (.java files)
5. Updated Event Sheet (.csv)
6. Updated Customer Sheet (.csv)
7. Log (.txt)