Advanced/Further Programming
COSC1295/COSC2391
Assignment 1 Sem 2 2023: Social Media Analyzer

Assessment Type	**Individual assignment;** no group work. Submit online via Canvas → Assignments → Assignment 1. Marks are awarded for meeting requirements as closely as possible according to assignment specifications and the supplied rubric. Clarifications or updates may be made via announcements.
Due Date	**Week 6, Friday August 25 11:59pm.** Late submission penalty will be applied unless special consideration has been granted. **There will be one milestone, demo check, during your prac class in Week 4.** You will describe and explain the key concepts adopted in your code and demonstrate code execution in the milestone.
Marks	16 marks out of 100 for assignment 1 submission. 4 marks (in addition to the 16 marks) out of 100 for milestone check. Rubrics for final submission and for milestone check will be posted to Canvas.

NOTE: Carefully read this document. In addition, please regularly follow the Canvas assignment discussion board for assignment related clarifications and discussions.

1. Overview

In this assignment, you are required to work individually to implement a basic Java console program that serves as a social media analyzer. The program should allow users to analyze posts from social media platforms and calculate statistics based on various criteria. You will practice your knowledge of Java basics, simple object-oriented (OO) concepts, Java Collections Framework, exceptions, and unit testing. You will use Java SE 8 or later.

2. Task specification

The program keeps a collection of anonymous posts that are collected from social media platforms. A list of collected social media posts is provided in `posts.csv` file (a csv file is a comma-separated values file). The first row in `posts.csv` file contains headers, and the rest of the rows contain the information of each post. Each row contains the following information of one post:

- ID: a unique ID (int) associated with each post
- content: the content of the social media post (String). For simplicity, you can assume that the content does not contain any "," symbol.
- author: an anonymous ID (String) of the post author
- likes: number of likes (non-negative integer) of the post
- shares: number of users (non-negative integer) that shared the post
- date-time: the date and the time that the post was first posted, in the format of (DD/MM/YYYY HH:MM)

The console-based program provides the following functionalities for users:

- Add a post to the collection
- Remove a post from the collection based on the post ID
- Retrieve a post from the collection based on the post ID
- Retrieve the top N posts with the most likes, and show retrieved posts in descending order of #likes
- Retrieve the top N posts with the most shares, and show retrieved posts in descending order of #shares

3. Assessment details

In this assignment, you will incrementally build a basic Java program. The assignment is structured into 1 milestone with video demo in week 4 and final submission in week 6.

Part A

You are required to implement the functions mentioned in the task specification in Section 1 - Overview. You can start this part from week 1 and work on it until the due date.

- You will incorporate basic object-oriented concepts (e.g., abstraction and encapsulation). In particular, you will define several classes to support the OO implementation. You will also need to choose appropriate data structures to store relevant information. You MUST use Java Collections because the JCF structure is growable.
- The console output should be well-formatted (HINT: you can use `System.out.printf` method for formatting).
- You can hard code the post information **in Part A**. You MUST use the information provided in `posts.csv`, for testing purposes; please do not create/add new posts information. After Week 5, you MUST change your program so that the collection of posts is read from the csv file automatically.
- You will need to document the code properly by following Code Conventions for the Java Programming Language by Oracle: https://www.oracle.com/java/technologies/javase/codeconventions-introduction.html

Following is a sample interaction with the social media analyzer. Text in **bold and green** would be input from the user:

```
Welcome to Social Media Analyzer!
--------------------------------------------------------------------------------
> Select from main menu
--------------------------------------------------------------------------------
   1) Add a social media post
   2) Delete an existing social media post
   3) Retrieve a social media post
   4) Retrieve the top N posts with most likes
   5) Retrieve the top N posts with most shares
   6) Exit
Please select: 2
Please provide the post ID: 1
Sorry the post does not exist in the collection!
--------------------------------------------------------------------------------
> Select from main menu
--------------------------------------------------------------------------------
   1) Add a social media post
   2) Delete an existing social media post
   3) Retrieve a social media post
   4) Retrieve the top N posts with most likes
   5) Retrieve the top N posts with most shares
   6) Exit
Please select: 1
Please provide the post ID: 100
Please provide the post content: I really enjoyed learning Java in this course!
Please provide the post author: XE1234
Please provide the number of likes of the post: 10000
Please provide the number of shares of the post: 100001
Please provide the date and time of the post in the format of DD/MM/YYYY HH:MM:
17/07/2023 10:00
The post has been added to the collection!
--------------------------------------------------------------------------------
> Select from main menu
--------------------------------------------------------------------------------
   1) Add a social media post
   2) Delete an existing social media post
   3) Retrieve a social media post
   4) Retrieve the top N posts with most likes
   5) Retrieve the top N posts with most shares
   6) Exit
Please select: 4
Please specify the number of posts to retrieve (N): 2
The top-2 posts with the most likes are:
   1) 100 | I really enjoyed learning Java in this course! | 10000
   2) 382 | What a miracle! | 2775
--------------------------------------------------------------------------------
> Select from main menu
--------------------------------------------------------------------------------
   1) Add a social media post
   2) Delete an existing social media post
   3) Retrieve a social media post
   4) Retrieve the top N posts with most likes
   5) Retrieve the top N posts with most shares
   6) Exit
Please select: 5
Please specify the number of posts to retrieve (N): 100
Only 6 posts exist in the collection. Showing all of them.
The top-6 posts with the most shares are:
   1) 100 | I really enjoyed learning Java in this course! | 100001
   2) 382 | What a miracle! | 13589
   3) 10 | Check out this epic film. | 1587
   4) 36778 | Fantastic day today. Congratulations to all winners. | 1214
   5) 37221 | Are we into Christmas month already?! | 25
   6) 20582 | Come and meet us at Building 14 of RMIT. | 24
--------------------------------------------------------------------------------
1) Add a social media post
   2) Delete an existing social media post
   3) Retrieve a social media post
   4) Retrieve the top N posts with most likes
   5) Retrieve the top N posts with most shares
   6) Exit
Please select: 6
Thanks for using Social Media Analyzer!
```

Part B

From Week 4, you will have the knowledge to work on this part.

You are required to write unit tests for all classes that include functions using JUnit framework. Your unit tests should have a good coverage of the typical use cases of your classes (e.g., calculating the total payment) and test all reasonable corner cases (e.g., handling invalid method arguments). You do not need to write unit tests for getters and setters.

Part C

From Week 4, you will have the knowledge to work on this part.

- You are required to catch and handle all reasonable and expected exceptions in your program. You need to consider the following cases:
    - When a user enters a menu option that is not valid.
    - When a user wants to add a new post with invalid information, e.g., invalid date and time, invalid value of #likes (such as entering “a” as #likes), invalid content (content containing “,”).
    - When the program cannot load the CSV file correctly.
- You will create customized exceptions that can be used in the program. For example, you can create an *InvalidMenuOption* exception. Then you can catch and handle the exception when a user enters an invalid menu

option. Avoid throwing an unspecific Exception; throw a specific exception (e.g., throw a *NumberFormatException* instead of an *IllegalArgumentException*).

4. Submission

You will submit all the relevant material as listed below via Canvas.

- You must have Main.java file which includes the main method. This file serves as the entry point of your program.
- You must include a README with instructions on how to compile and run your program from the command line. You will be given a sample console program and a document showing instructions on how to compile and run the sample program from the command line on Canvas. You must give similar instructions (as shown in step 2 in the given document) in your README.
- You must submit a zip file of your project. Make sure you zip the top-level project folder.
- You must not submit compiled files (*.class files) and any IDE related files (e.g., .settings folder generated by Eclipse). Please check your submission carefully, make sure all java files have been included.
- In the final submission, you must submit a video demonstrating the functionality of your code and unit tests. Note that this video demo is **different** from milestone check in week 4. You will receive **zero** mark if you do not include a video in the submission.
- You can submit your assignment as many times as you would like prior to the due date. The latest submission will be graded.

After the due date, you will have 5 business days to submit your assignment as a late submission. Late submissions will incur a penalty of 10% of the full assignment 1 marks per day. After these five days, Canvas will be closed, and you will lose ALL marks.

5. Marking Guidelines

- Functionality (5/20)
- Unit testing (4/20)
- Exception handling (2/20)
- Object oriented design and choice of data structures (2/20)
- Source code quality (3/20)
- Milestone check in week 4 (4/4)

Functionality: Menu selection and user input operations correctly implemented: 1 mark for each functionality in the program.

Unit testing: Unit tests with a good coverage of the typical use cases of classes and testing all reasonable corner cases.

Exception handling: Aim to catch and handle all possible exception; using specific exception classes.

Object oriented design and choice of data structures: Appropriate choice of data structures given the required functionalities of the program. Appropriate class designs to practice OO thinking.

Source code quality: Adequately documented and properly indented code; appropriate class/method/variable names.

Video demo in the final submission: Clearly perform the demo of all functions implemented and unit tests implemented. You must show your face while you demo. This is for the purpose of academic integrity. The demo must be within 8 minutes. You must use Microsoft Streams to create and share the demo. Instructions will be provided in a separate document.

Milestone check in week 4: Milestone check in week 4; clearly describe class design (i.e., classes and significant methods); clearly explain choice of data structures in JCF; demonstrating a working program that can allow a user to interact with the menu. The demo must be within 5 minutes.