CSC 241 Assignment 3
Abstract Data Types and Programming Methodology
Due: Thursday, March 10

1 Introduction

In this semester, we are going to develop a Java program called "***GradeManager***", which manages students' grades. This program provides functions such as adding/deleting/editing grades of certain students. All the grades are stored in designated file(s) and updated as grading data change. It also supports to manage students, *e.g.*, adding a new student and his/her grading data or deleting a grading record of a particular student. This program will be built up through several assignments, in each of which you will be asked to apply what you will learn in lectures. At the end of semester, you will have a Java program which utilizes various OOP techniques and diverse data structures.

The assignments will be handed in an order for completing a final program. So, you MUST follow instructions and achieve requirements when you work on an assignment. Java code for each assignment should be errorless and submitted in the Blackboard course shell. Since a next assignment usually asks you to add more functions or edit what have been made in the prior assignment, you should keep the previous Java code(s). If the prior program is submitted with errors or runs unsuccessfully, it must be corrected before it goes to the next assignment.

2 Goal for This Assignment

The aim of the **third** assignment is to read students' grade in a json file. Json files facilitate management of data in a more efficient way. In the lecture 04, we learned how to read/write a json file by using methods in json library called JSON Processing. For this assignment, you will use the same class hierarchy built in Assignment 2. The only difference is the format of data. Thus, you may copy your assignment 2 and modify it as the assignment 3 requests.

3 Instructions

A. Template file

Each assignment should be built in a package. The names of package and class for this assignment are below.

Package: Assignment3

Class: GradeManager

```
package Assignment2;
        …
public class GradeManager {
        …
```

In this assignment, a template package (**Assignment3>GradeManager.java**) is provided. You first unzip the **Assignment3.zip** which is in the Blackboard course shell. DO NOT change the package name or class name!

B. Data File

Data files are integrated in this assignment, that is, course info and grading data are merged into a json file, named "course_code.json," for instance, `cs241.json`. Each datum is formatted as follows:

```
"key": ["String_value" | number_value]
```

Note that string value needs double quotation marks, whereas number value needs nothing. A key-value pair is separated by comma(,), so more key-value pairs may follow.

If all the pairs are wrapped by {braces}, they are defined in an object level. A brace-wrapped pairs may be grouped by [bracket], which stands for an array. In the `cs241.json` file, there are many key-value pairs for example, "`CRN`": "`14607`", "`Capacity`": `24`, and so on. There is an array of a set of pairs, named `students`. You could find that the array `students` consists of four students, David, John, Matthew, and Rachel. Also, you could locate an array named `course works` for each course work such as q1, q2, and so on within in a student. For more information, you may review lecture 4 and example codes.

Json file is in `Assignment3/data` folder, which was the folder for text grading data files. If your program is in `/Users/jwlee/CSC241/src`, then the full path for the json file `cs241.json` is below:

`/Users/jwlee/CSC241/src/Assignment3/data/cs241.json`

Remember that there is only one file for both course information and grading data. Also, if you are using a different operating system, you will have different separator and full path.

C. Properties for configuration

From the assignment 3, you will use a Java properties file to provide necessary information for configuration of your program, such as a file path. It must be located in top directory of your program. You may check it by using `System.getProperty` as shown in **Figure 1**.

```
import java.util.Properties;
            …
System.out.println(System.getProperty("user.dir""));
```

Figure 1. You can print the path for your program by using `System.getProperty(key)`. It is defined in `java.util.Proprties`.

`config.properties` file is provided with the assignment 3. You should put it in the folder you found from `System.getProperty("user.dir")`. Then you have to update "filepath" to the path. Currently, it contains "`/Users/myaccount /CSC241/src`" as shown in **Figure 2**.

```
filepath = /Users/myaccount/CSC241/src
```

Figure 2. filepath is declared for `/Users/myaccount /CSC241/src` in `config.properties` file.

The properties file is loaded by Properties class shown in **Figure 3**.

```
Properties prop = new Properties();
FileInputStream fis = new FileInputStream("config.properties");
prop.load(fis);

// Read a json file
String packageName = GradeManager.class.getPackageName();
String fileName = "cs241.json";
String filePath = prop.getProperty("filepath") + File.separator +
packageName + File.separator + "data" + File.separator +
fileName;
```

Figure 3. "filepath" described in `config.properties` is loaded by an object of Properties. `getProperty` method returns information you are looking for.

D. Json library

As we reviewed in lecture 4, there are multiple choices for json library. For this assignment, you will use the one named JSON Processing (shortly json-p or `javax.json`) by Oracle. Two ways to add the library have been introduced in class, so you should review lectures before you add the library to your program.

1.Add dependencies for the library in IntelliJ

In this assignment, you will download a library and add it to the list of dependencies. You could locate `javax.json.jar` file which is provided with the assignment package. Download it, place it in your desired folder, and add it to the list of dependencies for your program. The list of dependencies is located in **Project Structure** window which can be accessed from "**File>Project Structure**." In **Project Structure** window, click "**Modules**" under **Project settings**, and then choose "**Dependencies**" tab. Then you could add the jar file by clicking plus(+) sign.

After this, you may check whether it is successfully added by simply entering "`import javax.json.Json;`" in your code.

2.Set environment variable in system

If you want to program in java without IDEs such as IntelliJ, you need to configure environment variable for the Json library.

First, you should point to a fold where the `javax.json` library is stored on your machine. For this, you will set the JSON_HOME environment variable. For Windows, you set the environment variable GJSON_HOME to the location of `javax.json` library. For Mac, you make a directory `/Library/json` and place it there. Then you add it by entering the following in terminal:

```
export JSON_HOME=/Library/json
```

For Linux, you will do the same process as Mac, but the location of the library is `/usr/local/json`.

Next, you will specify the jar file to enable it export in your program. For this, you set the CLASSPATH environment variable. For Windows, you set the environment variable CLASSPATH to following:

```
%CLASSPATH%;%JSON_HOME%\javax.json.jar;.;
```

For Mac/Linux, you add it by entering the following in terminal:

```
export CLASSPATH=$CLASSPATH:$JSON_HOME/javax.json.jar:.
```

You may enter `echo $JSON_HOME` or `echo $CLASSPATH` for verification on Mac/Linux.

E. Developing Environment

Your program should be **implemented in Java only**. The program in another language will not be graded.

F. Submission

You will submit your Java package. Zip the package **Assignment3** again and upload it in the Blackboard course shell. DO NOT copy and paste your code into text files such as rtf, doc, or txt. You MUST submit java files, not text files! The assignment will give you **two weeks**, so it is **due on Thursday, February 24**. All submission **by 11:59 PM** on that day will be accepted without any penalty. On the due date, Blackboard may be suffering of too much network traffics and be unstable. There is no excuse about the issue, therefore you are strongly recommended to submit earlier than the due date.

4 Requirements

A. Completeness of the requests from the assignment 2

Regardless of the goal of this assignment, your program should run as shown in **Figure 4**. If you already completed all requirements in the assignment 2, you would need concentrate on managing a json file. Unless, you must complete it first.

```
Enter a course code: CS241
Name: Abstract Data Types and Programming Methodology | CRN: 14607
| Code: ccs241 | Capacity: 24 | Time: 13:50
Select menu [edit | quit]? edit
Enter a student: John
Name: John | ID: 123456789 | Q1: 10 | Q2: 8 | Q3: 8 | Midterm:87 |
Final: 91
Enter a coursework you want to edit (q1,q2,q3,mid,final): q1
Enter a new score: 9
Name: John | ID: 123456789 | Q1: 9 | Q2: 8 | Q3: 8 | Midterm:87 |
Final: 91
Select menu [edit | quit]? quit
```

Figure 4. Your program loads grading data and edits score as user requests.

B. Proper management of a json file

The purpose of this assignment is to manage data in json file. Thus, the primary requirement is to upgrade your program, which enable to read/write a json file. Do NOT forget that the program must update the original file accordingly when data changes.

5 Grading

A. Grading criteria

The lab is assigned **30** points, which is 10% of the final grade. It will be graded by evaluating the requirement. Any missing and unsatisfiable criteria will take off points. The tentative and brief criteria are below.

- Compilation: **5** points
- Execution: **5** points
- Proper output: **20** points

B. Late penalty

Late submission will take off **10% per day** after due date. **Thus, submission after 10 days will not be accepted in any circumstances**.

6 Academic Integrity

Any dishonest behaviors will not be tolerated in this class. Any form of plagiarism and cheating will be dealt with according to the guidelines on the Academic Integrity Policy on-line at http://www.oswego.edu/integrity. For more information about university policies, see the following online catalog at:

http://catalog.oswego.edu/content.php?catoid=2&navoid=47#stat_inte_inte

Student who is against the honor code will not have any credits in this project.