Setriakor Addom

ICP Java Project Reflective Summary

For our Java project we were to create four java classes to represent an earthquake, and observatory, a monitor for the observatory and a monitoringIO class with a main method to run the program.

In the Earthquake class we were supposed to have instance variables for the magnitude, position in latitude and longitude, and the year of the event and in addition have the appropriate constructors and accessor and mutator methods for these variables. I made the magnitude variable a double, the year a string, and the position, latitude, and longitude variables strings.

In the observatory class we where to have instance variable s for the name of the observatory, the name of the country it is located in, the year in which the earthquake observations started, the area covered by the earthquakes and a list of earthquake events that it has recorded. I made the name, country, and year variables string objects. The area was an int object and I made an ArayList object to store the earthquake events it has recorded. We were also to add methods to return the largest magnitude earthquake recorded by the observatory, the average earthquake magnitude recorded at the observatory, a list of all earthquakes recorded at the observatory with a magnitude greater than a given number. I implemented the first method simply by keeping track of earthquake magnitudes as they were entered into the observatory class and every time an earthquake with a magnitude higher than the previous one was entered into the observatory stored its value has the highest magnitude and then when the method is called is simply return that value. For the average earthquake magnitude for the observatory I did something similar.

Every time an earthquake object is entered into the observatory I take its magnitude and add it to a int variable so that I get the sum of earthquake values. Then when the method is called I divide the sum by the size of the list and return that value as the average. For the method that return a list of earthquakes

In the Monitoring class, which is supposed to hold information about all the observatories, we are supposed to write methods to return the observatory with the largest average earthquake magnitude, the largest magnitude earthquake ever recorded, and a list of all earthquakes recorded with magnitude greater than a given number. This class had methods to return the observatory with the largest average earthquake magnitude, the largest magnitude earthquake ever recorded, a list of all earthquakes recorded with magnitude greater than a given number.

The MonitoringIO class, is the class that essentially runs the whole program and gives a user a list of commands for them to run such as add an earthquake, add an observatory, check for the observatory with the highest average magnitude, and return a list of earthquakes with a magnitude higher than a given number.

Our project also had to include a version 2 of the program which was to be implemented using a GUI and connected to a database to store the information about the earthquakes and observatories and retrieve information in the same manner as the version 1 of the program.

The GUI part of the project was the more difficult part of the project for me to do because it was more complicated and I hadn't done w program that involved connecting to a database before. Something that also made the project slightly more difficult for me was working with someone I didn't know well. I generally prefer to work alone or with people I know especially when programming so it was a bit of a difficult experience for me working with Ohemaa.

The main way my programming could have been improved is in the catching of errors. I didn’t really program in a way that was good at catching errors and so the code only really works perfectly if inputs are entered the way they are expected to and therefore the code can crash if unexpected inputs are given. In conclusion I enjoyed working on this project for the most part and it was an educational experience that I learned a lot from.