Programming Exercise: Searching Earthquake Data

For the following assignments, you will start with the files provided, using most of the classes, and modifying only a few of them, and create a new class. The classes provided are:

- The class **Location**, from the Android platform and revised for this course, a data class representing a geographic location. One of the constructors has parameters latitude and longitude, and one of the public methods is **distanceTo**.

- The class **QuakeEntry**, from the lesson, which has a constructor that requires latitude, longitude, magnitude, title, and depth. It has several get methods and a **toString** method.

- The class **EarthQuakeParser**, from the lesson, which has a **read** method with one String parameter that represents an XML earthquake data file and returns an ArrayList of QuakeEntry objects.

- The class **EarthQuakeClient**, which has been started for you and creates an EarthQuakeParser to read in an earthquake data file, creating an ArrayList of QuakeEntrys. You can test the program with the method **createCSV** to store an ArrayList of the earthquake data and print a CSV file. You will complete the methods that filter magnitude and distance in this class and add additional methods to it.

- The class **ClosestQuakes**, which has been started for you to find the ten closest quakes to a particular location. You will complete this method.

You can run your programs on the URL and see live data, but in order to make sure your program is correct we recommend running your program on a small file we have created that has only 25 lines of earthquake data. That file is **nov20quakedatasmall.atom**, and its contents printed as QuakeEntrys are:

```
# quakes read: 25
(33.54, -116.66), mag = 0.30, depth = -10410.00, title = 2km SE of Anza, California
(5.86, 126.18), mag = 4.50, depth = -68910.00, title = 55km S of Pondaguitan,
Philippines
(38.70, 20.51), mag = 4.80, depth = -10000.00, title = 17km W of Nidri, Greece
(26.38, 142.71), mag = 5.50, depth = -12890.00, title = 91km SSE of Chichi-shima,
Japan
(63.44, -147.62), mag = 1.60, depth = -7400.00, title = 66km E of Cantwell, Alaska
(40.37, 73.20), mag = 4.90, depth = -40790.00, title = 21km WNW of Gul'cha, Kyrgyzstan
```

(19.29, -155.23), mag = 2.10, depth = -2700.00, title = 16km S of Volcano, Hawaii
(0.91, 127.31), mag = 4.70, depth = -10000.00, title = 15km NNW of Kota Ternate, Indonesia
(49.41, 155.28), mag = 4.60, depth = -69810.00, title = 153km SSW of Severo-Kuril'sk, Russia
(36.27, -121.66), mag = 2.00, depth = -7630.00, title = 28km SSE of Carmel Valley Village, California
(63.25, -150.43), mag = 1.70, depth = -99900.00, title = 75km WSW of Cantwell, Alaska
(36.65, -117.13), mag = -0.20, depth = -9300.00, title = 44km SW of Beatty, Nevada
(-11.63, 165.52), mag = 5.10, depth = -20700.00, title = 106km SSW of Lata, Solomon Islands
(35.00, -118.21), mag = 1.30, depth = 1010.00, title = Quarry Blast - 7km SSW of Mojave, California
(49.39, -120.44), mag = 2.40, depth = -20.00, title = Explosion - 8km SSE of Princeton, Canada
(47.89, -122.71), mag = 1.90, depth = -18.00, title = Explosion - 4km SSW of Port Ludlow, Washington
(39.69, -119.41), mag = 0.00, depth = -7600.00, title = 16km WNW of Fernley, Nevada
(8.53, -71.34), mag = 5.00, depth = -25160.00, title = 5km ENE of Lagunillas, Venezuela
(37.00, -116.94), mag = -0.10, depth = -6700.00, title = 18km WNW of Beatty, Nevada
(-24.20, -66.84), mag = 4.60, depth = -185930.00, title = 53km W of San Antonio de los Cobres, Argentina
(45.12, -108.59), mag = 2.20, depth = 2000.00, title = Quarry Blast - 35km NNW of Lovell, Wyoming
(-36.16, -71.12), mag = 4.00, depth = -121990.00, title = 55km SE of Linares, Chile
(38.27, 142.53), mag = 4.60, depth = -30500.00, title = 109km E of Ishinomaki, Japan
(-24.67, -175.93), mag = 5.10, depth = -10000.00, title = South of Tonga
(34.05, -117.36), mag = 1.20, depth = 1040.00, title = Quarry Blast - 4km WNW of Grand Terrace, California

*This output has been modified to reflect the updated EarthQuakeParser class, 1/12/16. Note especially the possibility of negative magnitudes.

Assignment 1: Filtering by Magnitude and Distance

In this assignment you will complete the program to filter earthquake data by magnitude and distance, which was described in this lesson in the videos "Coding a Magnitude Filter" and "Coding a Distance Filter."

Specifically, for this assignment, you will only modify one class, the **EarthQuakeClient** class:

- Write the method **filterByMagnitude** that has already been started for you. This method has two parameters, an ArrayList of type QuakeEntry named **quakeData**, and a double named **magMin**. This method should return an ArrayList of type QuakeEntry of all the earthquakes from **quakeData** that have a magnitude larger than **magMin**. Notice that we have already created an ArrayList named **answer** for you to store those earthquakes that satisfy this requirement.

- Modify the method **bigQuakes** that has no parameters to use **filterByMagnitude** and print earthquakes above a certain magnitude, and also print the number of such earthquakes. Currently this method reads data on earthquakes from a file, stores a QuakeEntry for each earthquake read in the ArrayList named **list**, and prints out the number of earthquakes read in. After making modifications, when you run your program on the file **nov20quakedatasmall.atom** for quakes larger than 5.0, you should get the output:

```
read data for 25 quakes
(26.38, 142.71), mag = 5.50, depth = -12890.00, title = 91km SSE of Chichi-shima,
Japan
(-11.63, 165.52), mag = 5.10, depth = -20700.00, title = 106km SSW of Lata, Solomon
Islands
(-24.67, -175.93), mag = 5.10, depth = -10000.00, title = South of Tonga
Found 3 quakes that match that criteria
```

- Write the method **filterByDistanceFrom** that has already been started for you. This method has three parameters, an ArrayList of type QuakeEntry named **quakeData**, a double named **distMax**, and a Location named **from**. This method should return an ArrayList of type QuakeEntry of all the earthquakes from **quakeData** that are less than **distMax** from the location **from**. Notice that we have already created an ArrayList named **answer** for you to store those earthquakes that satisfy this requirement.

- Modify the method **closeToMe** that has no parameters to call **filterByDistance** to print

  out the earthquakes within 1000 Kilometers to a specified city (such as Durham, NC).

  For each earthquake found, print the distance from the earthquake to the specified city,

  followed by the information about the city (use **getInfo**). Currently this method reads

  data on earthquakes from a URL, stores a QuakeEntry for each earthquake read in the

  ArrayList named list, and prints out the number of earthquakes read in. It also gives the

  location for two cities, Durham, NC (35.988, -78.907) and Bridgeport, CA (38.17,

  -118.82). After making modifications, when you run your program on the file

  nov20quakedatasmall.atom for the city location Durham, NC, no earthquakes are

  found. But if you then run the program for the city location Bridgeport, CA, seven

  earthquakes are found, and you should get the output:

```
read data for 25 quakes
549.2146875 2km SE of Anza, California
329.17615625 28km SSE of Carmel Valley Village, California
225.440046875 44km SW of Beatty, Nevada
356.428375 Quarry Blast - 7km SSW of Mojave, California
176.09775 16km WNW of Fernley, Nevada
210.82540625 18km WNW of Beatty, Nevada
475.94878125 Quarry Blast - 4km WNW of Grand Terrace, California
Found 7 quakes that match that criteria
```

Assignment 2: Filtering by Depth

In this assignment you will filter earthquakes by their depth, finding those earthquakes whose depth is between a minimum and maximum value. For more information on what the "depth" of an earthquake means, see the information here:

http://earthquake.usgs.gov/learn/topics/seismology/determining_depth.php

Specifically, for this assignment, you will add new methods to one class, the **EarthQuakeClient** class:

- Write the method **filterByDepth** that has three parameters, an ArrayList of type QuakeEntry named **quakeData**, a double named **minDepth** and a double named **maxDepth**. This method should return an ArrayList of type QuakeEntry of all the earthquakes from **quakeData** whose depth is between **minDepth** and **maxDepth**, exclusive. (Do not include quakes with depth exactly **minDepth** or **maxDepth**.)

- Write the void method **quakesOfDepth** that has no parameters to use **filterByDepth** and print all the earthquakes from a data source whose depth is between a given minimum and maximum value. You should also print out the number of earthquakes found. After writing this method, when you run your program on the file **nov20quakedatasmall.atom** for quakes with depth between -10000.0 and -5000.0 you should find five such quakes and get the output:

```
read data for 25 quakes
Find quakes with depth between -10000.0 and -5000.0
(63.44, -147.62), mag = 1.60, depth = -7400.00, title = 66km E of Cantwell, Alaska
(36.27, -121.66), mag = 2.00, depth = -7630.00, title = 28km SSE of Carmel Valley
Village, California
(36.65, -117.13), mag = -0.20, depth = -9300.00, title = 44km SW of Beatty, Nevada
(39.69, -119.41), mag = 0.00, depth = -7600.00, title = 16km WNW of Fernley, Nevada
(37.00, -116.94), mag = -0.10, depth = -6700.00, title = 18km WNW of Beatty, Nevada
Found 5 quakes that match that criteria
```

Assignment 3: Filtering by Phrase in Title

In this assignment you will filter earthquakes by a phrase in the title given for the earthquake in three ways, finding those earthquakes whose title starts with a phrase, ends with a phrase, or just has a phrase somewhere in the title.

Specifically, for this assignment, you will add new methods to one class, the **EarthQuakeClient** class:

- Write the method **filterByPhrase** that has three parameters, an ArrayList of type QuakeEntry named **quakeData**, a String named **where** that indicates where to search in the title and has one of three values: ("start", "end", or "any"), and a String named **phrase**, indicating the phrase to search for in the title of the earthquake. The title of the earthquake can be obtained through the **getInfo** method. The **filterByPhrase** method should return an ArrayList of type QuakeEntry of all the earthquakes from **quakeData** whose titles have the given **phrase** found at location **where** ("start" means the phrase must start the title, "end" means the phrase must end the title and "any" means the phrase is a substring anywhere in the title.)

- Write the void method **quakesByPhrase** to use **filterByPhrase** and print all the earthquakes from a data source that have phrase in their title in a given position in the title. You should also print out the number of earthquakes found. After writing this method, when you run your program on the file **nov20quakedatasmall.atom** for quakes with **phrase** "California" and **where** set to "end" you should find four such quakes and get the output:

```
read data for 25 quakes
(33.54, -116.66), mag = 0.30, depth = -10410.00, title = 2km SE of Anza, California
(36.27, -121.66), mag = 2.00, depth = -7630.00, title = 28km SSE of Carmel Valley
Village, California
(35.00, -118.21), mag = 1.30, depth = 1010.00, title = Quarry Blast - 7km SSW of
Mojave, California
(34.05, -117.36), mag = 1.20, depth = 1040.00, title = Quarry Blast - 4km WNW of Grand
Terrace, California
Found 4 quakes that match California at end
```

For another example, when you run your program on the file **nov20quakedatasmall.atom** for quakes with **phrase** "Can" and **where** set to "any" you should find three such quakes and get the output:

```
read data for 25 quakes
(63.44, -147.62), mag = 1.60, depth = -7400.00, title = 66km E of Cantwell, Alaska
(63.25, -150.43), mag = 1.70, depth = -99900.00, title = 75km WSW of Cantwell, Alaska
(49.39, -120.44), mag = 2.40, depth = -20.00, title = Explosion - 8km SSE of
Princeton, Canada
Found 3 quakes that match Can at any
```

For another example, when you run your program on the file **nov20quakedatasmall.atom** for quakes with **phrase** "Explosion" and **where** set to "start" you should find two such quakes and get the output:

```
read data for 25 quakes
(49.39, -120.44), mag = 2.40, depth = -20.00, title = Explosion - 8km SSE of
Princeton, Canada
(47.89, -122.71), mag = 1.90, depth = -18.00, title = Explosion - 4km SSW of Port
Ludlow, Washington
Found 2 quakes that match Explosion at start
```

Assignment 4: Finding the Closest Earthquakes to a Location

In this assignment you will complete the program to determine the N closests earthquakes to a specified location that was described in this lesson in the video.

Specifically, for this assignment, you will only modify one class, the **ClosestQuakes** class:

- The first method you need has already been written for you. The method **findClosestQuakes** reads in data on earthquakes storing them in the ArrayList list and prints how many quakes there are. It sets a location variable named **jakarta** to the location of the city Jakarta. It then calls the method **getClosest** to determine the ten closest earthquakes in list and prints information about those quakes and how close they are to Jakarta. This method has already been written for you, but doesn't work yet since the method **getClosest** is not complete.

- Complete the method **getClosest** that has already been started for you. This method has three parameters, an ArrayList of type QuakeEntry named **quakeData**, a Location named **current**, and an int named **howMany**. This method should find the closest number of **howMany** earthquakes to the **current** Location and return them in an ArrayList of type QuakeEntry. The earthquakes should be in the ArrayList in order with the closest earthquake in index position 0. If there are fewer then **howMany** earthquakes in **quakeData**, then the ArrayList returned would be the same size as **quakeData**.

- Now run the method **findClosestQuakes** by calling **getClosest** with the location **current** set to Jakarta (-6.211,106.845) and **howMany** set to 3. When you run your program on the file **nov20quakedatasmall.atom** you should find the three closest quakes and get the output:

```
read data for 25
2406.51     (0.91, 127.31), mag = 4.70, depth = -10000.00, title = 15km NNW of Kota
Ternate, Indonesia
2529.61     (5.86, 126.18), mag = 4.50, depth = -68910.00, title = 55km S of
Pondaguitan, Philippines
5289.10     (26.38, 142.71), mag = 5.50, depth = -12890.00, title = 91km SSE of
Chichi-shima, Japan
number found: 3
```

Assignment 5: Finding the Largest Magnitude Earthquakes

In this assignment you will write a new class and methods to determine the N biggest earthquakes, those with largest magnitude.

Specifically, for this assignment, you will:

- Write a new class named **LargestQuakes**. Be sure to import java.util.*;
- Write a void method named **findLargestQuakes** that reads in earthquake data from a source and storing them into an ArrayList of type QuakeData. Then it prints all the earthquakes and how many earthquakes that were from the source. You should read in earthquakes from the small file **nov20quakedatasmall.atom**, print all the earthquakes and also print how many there are. After this works you should comment out the printing of all the earthquakes, but continue to print out the total number of earthquakes read in.
- Write a method named **indexOfLargest** that has one parameter, an ArrayList of type QuakeEntry named **data**. This method returns an integer representing the index location in **data** of the earthquake with the largest magnitude. You should test out this method by adding code to the method **findLargestQuakes** to print the index location of the largest magnitude earthquake in the file **nov20quakedatasmall.atom** and the earthquake at that location. You will see that the largest such earthquake is at location 3 and has magnitude 5.50. Once this works you may want to comment this out.
- Write a method named **getLargest** that has two parameters, an ArrayList of type QuakeEntry named **quakeData** and an integer named **howMany**. This method returns an ArrayList of type QuakeEntry of the top **howMany** largest magnitude earthquakes from quakeData. The quakes returned should be in the ArrayList in order by their magnitude, with the largest magnitude earthquake in index position 0. If **quakeData** has fewer than **howMany** earthquakes, then the number of earthquakes returned in the ArrayList is equal to the number of earthquakes in **quakeData**. This method should call the method **indexOfLargest**.

- Modify the method **findLargestQuakes** to call **getLargest** to print the five earthquakes of largest magnitude from the file **nov20quakedatasmall.atom**. Those five earthquakes are:

```
(26.38, 142.71), mag = 5.50, depth = -12890.00, title = 91km SSE of Chichi-shima,
Japan
(-11.63, 165.52), mag = 5.10, depth = -20700.00, title = 106km SSW of Lata, Solomon
Islands
(-24.67, -175.93), mag = 5.10, depth = -10000.00, title = South of Tonga
(8.53, -71.34), mag = 5.00, depth = -25160.00, title = 5km ENE of Lagunillas,
Venezuela
(40.37, 73.20), mag = 4.90, depth = -40790.00, title = 21km WNW of Gul'cha, Kyrgyzstan
```