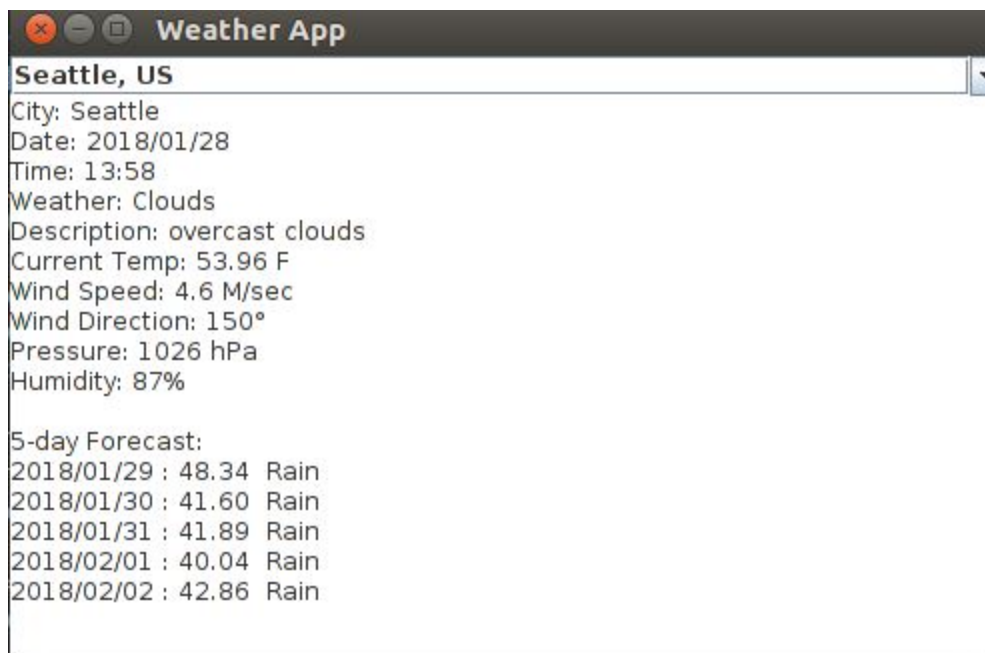


Application description:

The application for Program 2 consists of a weather application that is built using Java 8 and various 3rd party packages to simplify parsing Json objects from the OpenWeatherMap Rest API. The application takes a (city name, country) and returns information about the current weather such as weather description, temperature, wind speed, wind direction, atmospheric pressure, humidity, and the time and date of the report. The application also returns a 5-day forecast with the average temperature throughout the day.



Design:

Front end

The user interface consists of a JFrame with a JComboBox using autocomplete, and a text field. The text field displays the weather data that is retrieved for the city highlighted in the combobox field. The combobox will automatically update the combobox textfield with possible city choices available from the OWM API.

Backend

The weather API used to display data to the user is OpenWeatherMap and the data is returned in the form of a Json object that the java package Jsonlterator helps to parse out. Jsonlterator is a useful package the combines the Google gson and fasterXML.jackson Json parsing packages and handles the parsing of large Json objects. Usage of classes Weather.java and City.java are used to deserialize the Json objects of the weather API and the list of cities the API uses to create the autocomplete entries in the Jcombo box. The class WeatherGUI contains the applications main class and all the data related to the user interface. The APIrequest class is responsible for making the URL request calls to the weather API and returning the Json object of weather data to be parsed out. Java2sAutoTextField and Java2sAutoComboBox are 3rd party packages from Sun Micro that extend the Java.swing class combobox and get a list of the cities used by the weather API to autocomplete city names for the user. A Json file is provided by the OpenWeatherAPI with all the cities it contains data on. When the application is opened, the cities file is parsed out to populate the autocomplete combo box.

Usage:

In Linux:

Unzip the Program2 zip and extract the 'src' directory to your preferred location. Open a terminal window and change directory into the 'src' directory that has just been extracted.

Run the following commands to compile all the source code, and then execute the weather application:

```
./p2.sh
```

```
java WeatherGUI
```

If p2.sh is not executable run:

```
sudo chmod +x p2.sh
```

And type your password to make it an executable.

p2.sh contains the javac command that will compile all the source code and the needed jar files.