CS 272 In-Class Exercise 15

Part 1 – XML and XPath

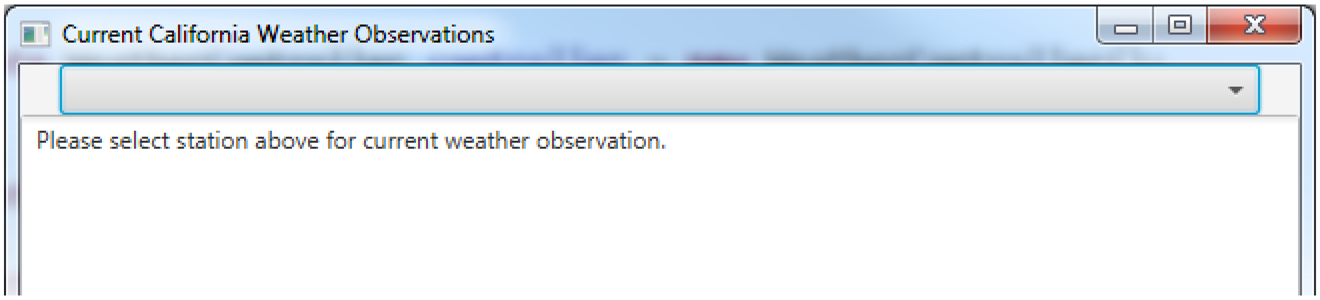
CA Weather Observations in Real-Time: In this exercise, you'll learn how to use the DocumentBuilder, Document and XPath classes in the javax.xml.* packages to write source code that imports live, real-time data into your applications.

Specifically, we will be using live XML data from the National Weather Service (NOAA), which has thousands of weather reporting stations throughout the United States and Canada. In this assignment, we will be processing a master list (weather_stations.xml) and filtering the stations, such that only those from California will appear. In this file, each station is a child element and has the following format (e.g. KSNA – Weather Station for John Wayne Airport in Santa Ana):

```
<station>
    <station_id>KSNA</station_id>
    <state>CA</state>
            <station_name>Santa Ana, John Wayne Airport-Orange County Airport</station_name>
    <latitude>33.68</latitude>
    <longitude>-117.86639</longitude>
            <html_url>http://weather.noaa.gov/weather/current/KSNA.html</html_url>
            <rss_url>https://w1.weather.gov/xml/current_obs/KSNA.rss</rss_url>
            <xml_url>https://w1.weather.gov/xml/current_obs/KSNA.xml</xml_url>
</station>
```

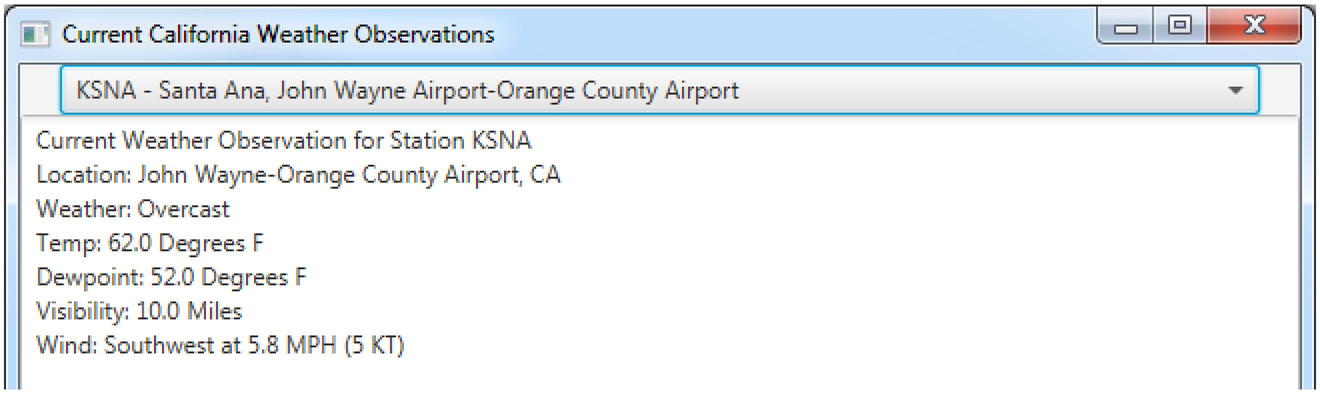
Please download **IC15_WXObserver.zip** and import this existing project into Eclipse (you do not need to unzip the file). In this exercise, we will be adding design and functionality to allow a user to select one of the many California weather stations (as identified by NOAA) and get the most current weather observation.

WXObserver upon Launch



Please select station above for current weather observation.

WXObserver after selecting station KSNA (John Wayne Airport)


Current California Weather Observations
KSNA - Santa Ana, John Wayne Airport-Orange County Airport

```
Current Weather Observation for Station KSNA
Location: John Wayne-Orange County Airport, CA
Weather: Overcast
Temp: 62.0 Degrees F
Dewpoint: 52.0 Degrees F
Visibility: 10.0 Miles
Wind: Southwest at 5.8 MPH (5 KT)
```

Please note this application is currently displaying two types of objects (already built for you):

1. **WeatherStation** – this type of object is being displayed in the ComboBox. You can see the toString() method is producing the message "KSNA – Santa Ana, John Wayne Airport-Orange County Airport", which is the concatenation of mStationId and mStationName. The ComboBox items are set to an ObservableList<WeatherStation>.

2. **WeatherObservation** – this type of object is being displayed in the larger TextArea beneath the ComboBox. The WeatherObservation object contains the latest weather information for the specific WeatherStation selected in the ComboBox, which will be streamed in real-time from the <xml_url> location of the WeatherStation. Note the toString() method produces the formatted text for the weather observation.

Here's what you'll need to complete for the WXObserver application:

In **<u>WeatherController.java</u>** (controller package)

1. Implement the method **getCAWeatherStations()**, which will process the master file weather_stations.xml and build an ObservableList of those stations whose state is "CA".

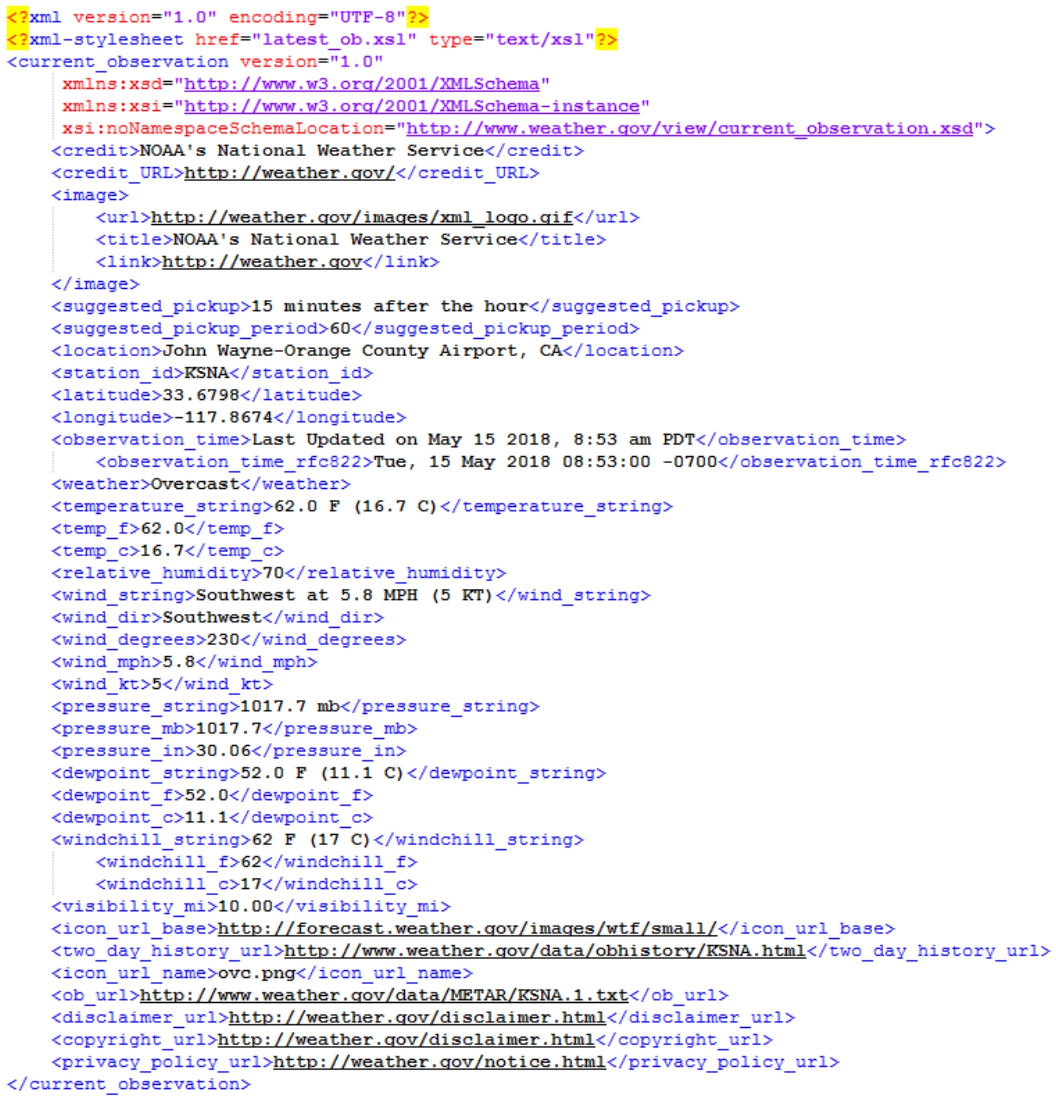
```java
/**
 * Gets a list of California weather stations as recognized by (NOAA).
 * This list is parsed from a master file, weather_stations.xml, which
 * contains all the reporting stations throughout the US and Canada.
 * The method extracts only those stations in California.
 * @return A list of California weather stations.
 */
public ObservableList<WeatherStation> getCAWeatherStations()
```

- Create a DocumentBuilder/Document object connected to the weather_stations.xml file
- Create a XPath object which allows traversal of elements in the XML file
- Loop through all station elements, if the state for the station is "CA":
- Extract the station id, name, and XML URL
- Create a new WeatherStation object and add it to the observable list
- Should an Exception occur, print the stack trace
- Return the list of CA weather stations

2. Implement the method **getCurrentWeather(String xmlURL)**, which will process the master file weather_stations.xml and build an ObservableList of those stations whose state is “CA”.

```java
/**
 * Gets the current weather observation using NOAA weather data
 * from a given XML URL (station id).  This XML URL is a resource on the web,
 * which will be parsed in real-time to extract the station id, location,
 * weather, temperature, dewpoint, visibility and wind.
 *
 * @param xmlURL The URL to the XMl resource on the web containing weather
 * data for a particular station.
 * @return The current weather observation for the station.
 */
      public WeatherObservation getCurrentWeather(String xmlURL) {
```

- Create a DocumentBuilder/Document object connected to the XML URL
- Create a XPath object which allows traversal of elements in the XML stream
- From the current_observation element, extract the:
  * station id, location, weather, temp, dewpoint, visibility and wind
- Instantiate a new WeatherObservation object from those 6 fields and return it.
- Below is a screen shot indicating the format of the XML structure, which we will be streaming. This example shows a current observation for KSNA – John Wayne Airport.



```xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="latest_ob.xsl" type="text/xsl"?>
<current_observation version="1.0"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://www.weather.gov/view/current_observation.xsd">
    <credit>NOAA's National Weather Service</credit>
    <credit_URL>http://weather.gov/</credit_URL>
    <image>
        <url>http://weather.gov/images/xml_logo.gif</url>
        <title>NOAA's National Weather Service</title>
        <link>http://weather.gov</link>
    </image>
    <suggested_pickup>15 minutes after the hour</suggested_pickup>
    <suggested_pickup_period>60</suggested_pickup_period>
    <location>John Wayne-Orange County Airport, CA</location>
    <station_id>KSNA</station_id>
    <latitude>33.6798</latitude>
    <longitude>-117.8674</longitude>
    <observation_time>Last Updated on May 15 2018, 8:53 am PDT</observation_time>
        <observation_time_rfc822>Tue, 15 May 2018 08:53:00 -0700</observation_time_rfc822>
    <weather>Overcast</weather>
    <temperature_string>62.0 F (16.7 C)</temperature_string>
    <temp_f>62.0</temp_f>
    <temp_c>16.7</temp_c>
    <relative_humidity>70</relative_humidity>
    <wind_string>Southwest at 5.8 MPH (5 KT)</wind_string>
    <wind_dir>Southwest</wind_dir>
    <wind_degrees>230</wind_degrees>
    <wind_mph>5.8</wind_mph>
    <wind_kt>5</wind_kt>
    <pressure_string>1017.7 mb</pressure_string>
    <pressure_mb>1017.7</pressure_mb>
    <pressure_in>30.06</pressure_in>
    <dewpoint_string>52.0 F (11.1 C)</dewpoint_string>
    <dewpoint_f>52.0</dewpoint_f>
    <dewpoint_c>11.1</dewpoint_c>
    <windchill_string>62 F (17 C)</windchill_string>
        <windchill_f>62</windchill_f>
        <windchill_c>17</windchill_c>
    <visibility_mi>10.00</visibility_mi>
    <icon_url_base>http://forecast.weather.gov/images/wtf/small/</icon_url_base>
    <two_day_history_url>http://www.weather.gov/data/obhistory/KSNA.html</two_day_history_url>
    <icon_url_name>ovc.png</icon_url_name>
    <ob_url>http://www.weather.gov/data/METAR/KSNA.1.txt</ob_url>
    <disclaimer_url>http://weather.gov/disclaimer.html</disclaimer_url>
    <copyright_url>http://weather.gov/disclaimer.html</copyright_url>
    <privacy_policy_url>http://weather.gov/notice.html</privacy_policy_url>
</current_observation>
```

In **<u>MainView.java</u>** (view package)

1. Implement the method **initialize(..)**, which will initialize the components (nodes) on the JavaFX scene. Specifically, the ComboBox should be loaded with all the California weather stations and the TextArea should have a helpful message for the user.

    a. Set the combo box items to the CA Weather Stations
    b. Set the text of the TextArea to "Please select station above for current weather observation."

2. Implement the method **getCurrentWeather()**, which will stream the latest weather observation for the station selected in the ComboBox.
    a. Get the selected weather station from the ComboBox.
    b. If it's not null, get the current weather observation for that station.
    c. Set the text of the TextArea to the current weather observation

****When finished, please remember to write Javadocs for ALL public and protected members of your classes and export your project as a zip file (IC15_WXObserver_Complete.zip) and upload it to Canvas****