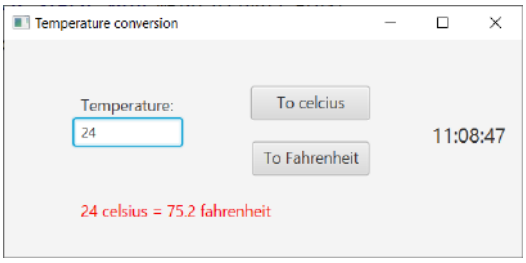
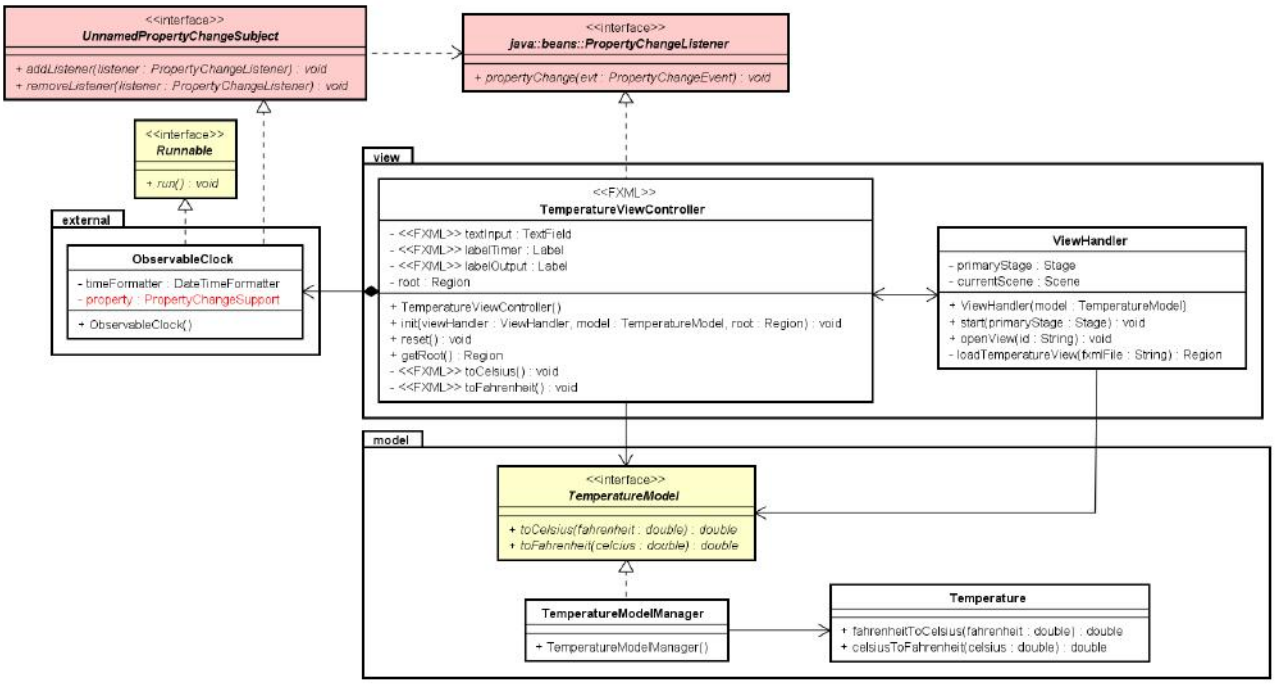
A JavaFX temperature Converter with *an observable* Clock

Re-implement the temperature converter exercise using the Observer design pattern.

The clock (`RunnableClock` class) is a subject to be observed by one or many observers. Class `RunnableClock` do not know the GUI (no variable of type `TemperaturConverterController`), and instead fires events every second. Thus, making it a general class to be used in many different applications. You could rename class `RunnableClock` to `ObservableClock`.





Step 1: Update class `RunnableClock/ObservableClock`

- Delete the `TemperatureViewController` instance variable and the parameter to the constructor.
- The parts related to the Observer pattern:
  - Implement an interface with methods to add and remove a listener, e.g. interface `UnnamedPropertySubject` as defined in the `MyObserver.jar` file
  - Add an instance variable of type `PropertyChangeSupport` and initialize it in the constructor (with `this` as the argument)
  - Delegate to the `PropertyChangeSupport` instance variable in the two methods from the interface
  - In the `run` method, replace the call to method `showTime` (and the printout) with a statement firing an event (e.g. with the string version of the current time). *Note that you may have to wrap the statement(s) into a Platform.runLater to be able to update components in a JavaFX GUI.*

Step 2: Update class `TemperatureViewController`

- Implement interface `PropertChangeListener`
    - In the `PropertyChange` method, get the time string from the event (convert it to the correct type) and make the same statement as you have in the `showTime` method (set the label text to the time string – and wrap the statement in a `Platform.runLater`)
- Delete method `showTime`.

Step 3: Run the `main` method in class `Main` and observe the result.