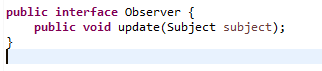
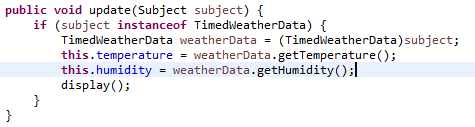
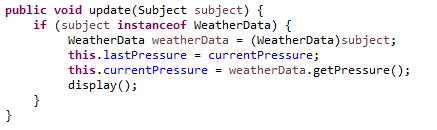
1. notifyObservers() 메소드에서 호출하는 observer의 update() 메소드 는 Push 방식으로 값을 전달하는데, 이를 Pull 방식으로 동작하도록 수정하여라.

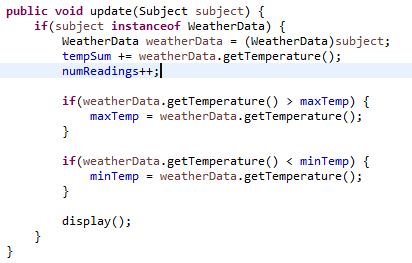




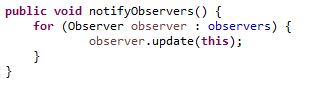
CurrentConditonDisplay 클래스



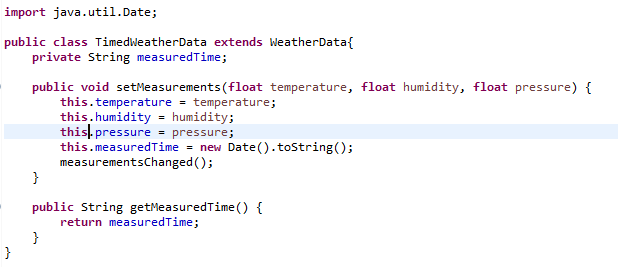
ForecastDisplay 클래스



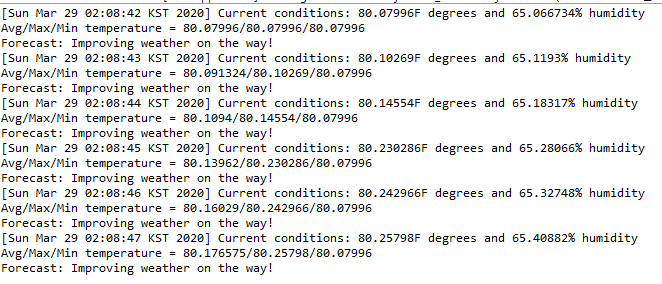
StatisticsDisplay 클래스



1. WeatherData 의 서브클래스인 TimedWeatherData 를 정의하고, 상태 변화가 측정된 시간 (measuredTime)을 String 타입의 속성으로 추가 하라. 또한, setMeasurements() 메소드가 호출될 때마다, 현재 시간을 java.util.Date 클래스를 이용하여 얻어와서 (힌트, new Date().toString()) measuredTime 속성의 값으로 설정하도록 setMeasurement() 메소드 를 오버라이딩 하여라.



1. CurrentConditionsDisplay 클래스를 수정하여 측정된 시간이 출력될 수 있도록 하여라. 



1. 위의 결과를 바탕으로 푸시 방식과 풀 방식의 장단점에 대해서 논의 하시오.

push방식

장점 full방식의 단점의 시점의 차이로 인한 데이터 변경의 위험성이 없다.

단점 모든 상태정보를 넘기지만 모든 상태정보를 사용하지는 않는다.

full방식

장점 자신이 원하는 정보만 WeatherData로부터 가져올 수 있다.

단점

상태정보를 가져오기 위해서 추가적인 메소드 호출이 필요하다.

메소드의 호출 시점과 실제 데이터를 가져오는 시점의 달라질 수 있기 때문에 원하는 정보를 가져오지 못할 수도 있다.