Applying Key Operators in the Observable Class: Case Study ex2 (Part 1)

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

Institute for Software Integrated Systems

Vanderbilt University
Nashville, Tennessee, USA

Learning Objectives in this Part of the Lesson

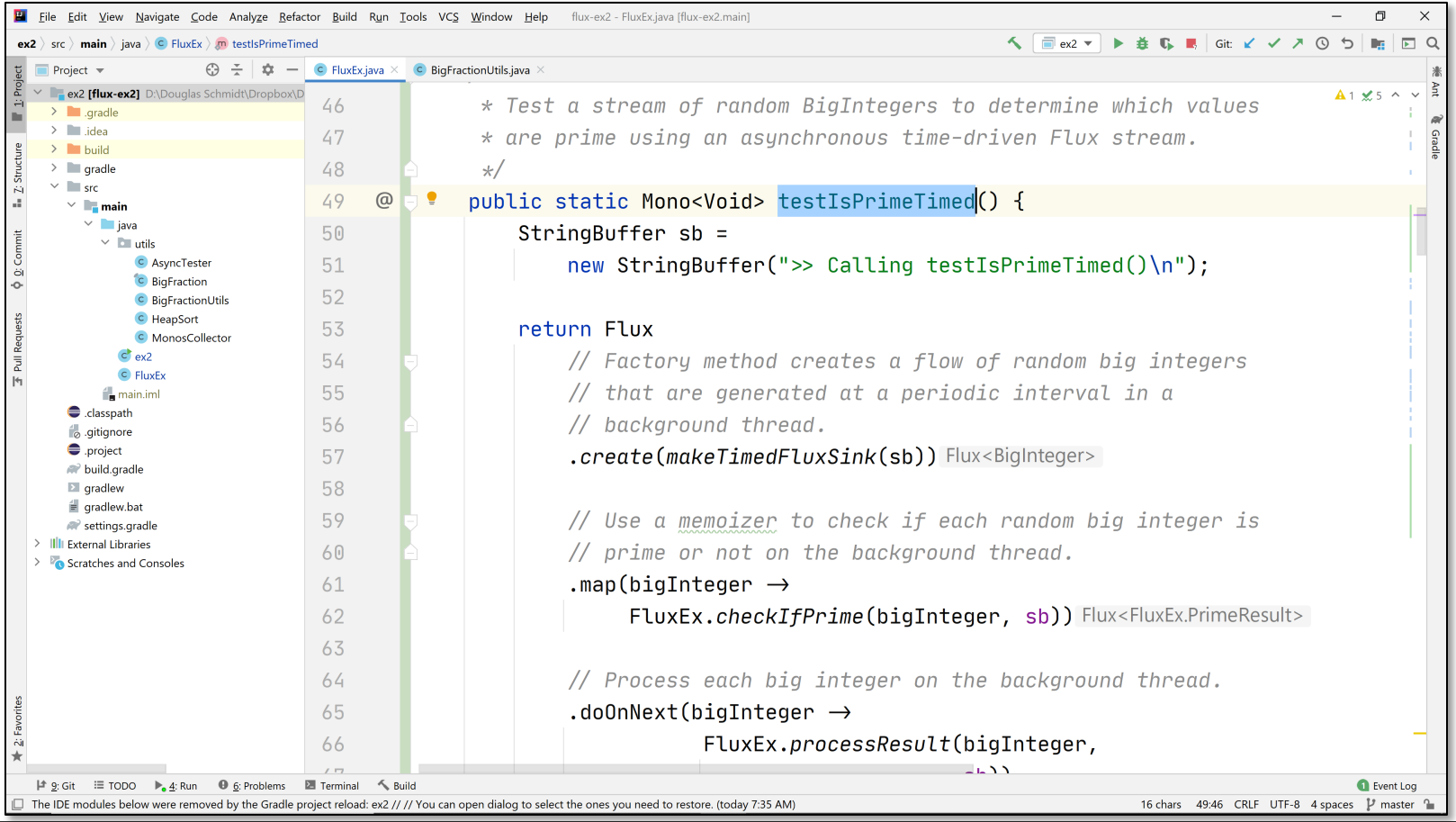
- Part 1 of this case study explores how to use the RxJava Observable operators map(), create(), interval(), ignoreElements(), doOnNext(), take(), subscribeOn(), subscribe(), filter(), & doOnComplete() to create large random BigInteger objects & check asynchronously if they are prime in a background thread

```
Observable
   .create
     (ObservableEx::emitInterval)

   .map(bigInteger ->
        ObservableEx.checkIfPrime
          (bigInteger, sb))

   ...

   .doOnComplete(() ->
          BigFractionUtils.
          display(sb.toString()))

   .ignoreElements();
```

See github.com/douglascraigschmidt/LiveLessons/tree/master/Reactive/Observable/ex2

Applying Key Operators in the Observable Class to ex2

Applying Key Operators in the Flux Class to ex2



See github.com/douglascraigschmidt/LiveLessons/tree/master/Reactive/Observable/ex2

End of Applying Key Methods in the Observable Class: Case Study ex2 (Part 1)