

Applying Key Operators in the Flux Class:

Case Study ex1

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

- Case study ex1 shows how to use Flux operators just(), fromIterable(), fromArray(), from(), doOnNext(), map(), mapNotNull(), mergeWith(), repeat(), & subscribe() to create, reduce, multiply, & display BigFraction objects synchronously

Flux

```
.just(BigFraction.valueOf(100,3),  
      BigFraction.valueOf(100,4),  
      BigFraction.valueOf(100,2),  
      BigFraction.valueOf(100,1))  
  
.map(fraction -> fraction  
    .multiply(sBigReducedFraction))  
  
.subscribe  
  (fraction -> sb.append(" = "  
    + fraction.toMixedString()  
    + "\n"),  
   error -> sb.append("error"),  
   () -> BigFractionUtils  
    .display(sb.toString()));
```

See github.com/douglasraigschmidt/LiveLessons/tree/master/Reactive/flux/ex1

Learning Objectives in this Part of the Lesson

- Case study ex1 shows how to use Flux operators just(), fromIterable(), fromArray(), from(), doOnNext(), map(), mapNotNull(), mergeWith(), repeat(), & subscribe() to create, reduce, multiply, & display BigFraction objects synchronously
- It also shows how to use the Mono fromCallable() operator

```
Flux<BigFraction> f2 = Flux
    .from(Mono
        .fromCallable(() ->
            makeBigFraction
                (random, true)))
    .repeat(4);
...
```

Applying Key Operators in the Flux Class to ex1

Applying Key Operators in the Flux Class to ex1

```
11 /**
12  * This class shows how to apply Project Reactor features
13  * synchronously to perform basic Flux operations, including just(),
14  * map(), and subscribe().
15  * https://projectreactor.io/docs/core/release/api/reactor/core/publisher/Flux.html
16  */
17 public class FluxEx {
18     /**
19      * Test BigFraction multiplication using a synchronous Flux
20      * stream.
21      */
22     public static Mono<Void> testFractionMultiplication() {
23         StringBuilder sb =
24             new StringBuilder(">> Calling testFractionMultiplication()\n");
25
26         Flux
27             // Use just() to generate a stream of big fractions.
28             // https://projectreactor.io/docs/core/release/api/reactor/core/publisher/Flux
29             .just(BigFraction.valueOf( numerator: 100, denominator: 3),
30                 BigFraction.valueOf( numerator: 100, denominator: 4),
31                 BigFraction.valueOf( numerator: 100, denominator: 2),
32                 BigFraction.valueOf( numerator: 100, denominator: 1))
33     }
```

See github.com/douglasraigschmidt/LiveLessons/tree/master/Reactive/flux/ex1

End of Applying Key Operators in the Flux Class: Case Study ex1