Applying Key Operators in the Flowable 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 applies RxJava Flowable features to demonstrate various types of backpressure strategies (e.g., MISSING, BUFFER, ERROR, LATEST, & DROP) between a Publisher & a Subscriber that run in the context of different Scheduler objects.

```java
return Flowable
  .create(NonBackpressureEmitter
          .makeEmitter(count,
                        maxValue,
                        mPendingItemCount),
          overflowStrategy)
          .onErrorResumeNext(error -> {
            debug(error.getMessage());
            return Flowable.empty();
          })
          .subscribeOn(scheduler);
```
Applying Key Operators in the Flowable Class to ex1
Applying Key Operators in the Flowable Class to ex1

See [github.com/douglasraigschmidt/LiveLessons/tree/master/Reactive/Flowable/ex1](https://github.com/douglasraigschmidt/LiveLessons/tree/master/Reactive/Flowable/ex1)
End of Applying Key Operators in the Flowable Class: Case Study ex1