Implementing the AsyncTaskBarrier Framework Using RxJava (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

- Understand the API of the AsyncTaskBarrier class for RxJava

Class AsyncTaskBarrier

```
public class AsyncTaskBarrier
extends java.lang.Object
```

This class asynchronously runs tasks that use the RxJava framework and ensures that the calling method doesn't exit until all asynchronous task processing is completed.

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
`static void`	`register (io.reactivex.rxjava3.functions.Supplier<io.reactivex.rxjava3.core.Completable> task)`	Register the `task` task so that it can be run asynchronously.
`static io.reactivex.rxjava3.core.Single<java.lang.Long>`	`runTasks()`	Run all the register tasks.

See Reactive/Observable/ex4/src/main/java/utils/AsyncTaskBarrier.java

Learning Objectives in this Part of the Lesson

- Understand the API of the AsyncTaskBarrier class for RxJava
- Know how to use AsyncTaskBarrier in practice

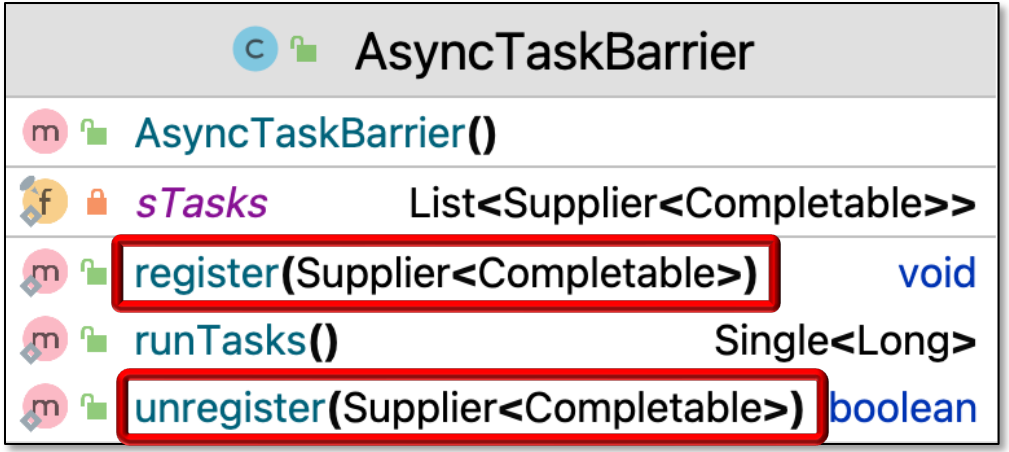
```
AsyncTaskBarrier.register(this::syncThrowException);
AsyncTaskBarrier.register(this::asyncThrowException);
AsyncTaskBarrier.register(this::syncNoException);
AsyncTaskBarrier.register(this::asyncNoException);

long testCount = AsyncTaskBarrier
    .runTasks()
    .blockingGet();

assertEquals(testCount, 2);
```

See Reactive/Observable/ex4/src/test/java/utils/AsyncTaskBarrierTests.java

The AsyncTask Barrier Class API

The AsyncTaskBarrier Class API

- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks
  - It provides methods to register & unregister tasks



Interface Supplier<T>

Type Parameters:

T - the type of results supplied by this supplier

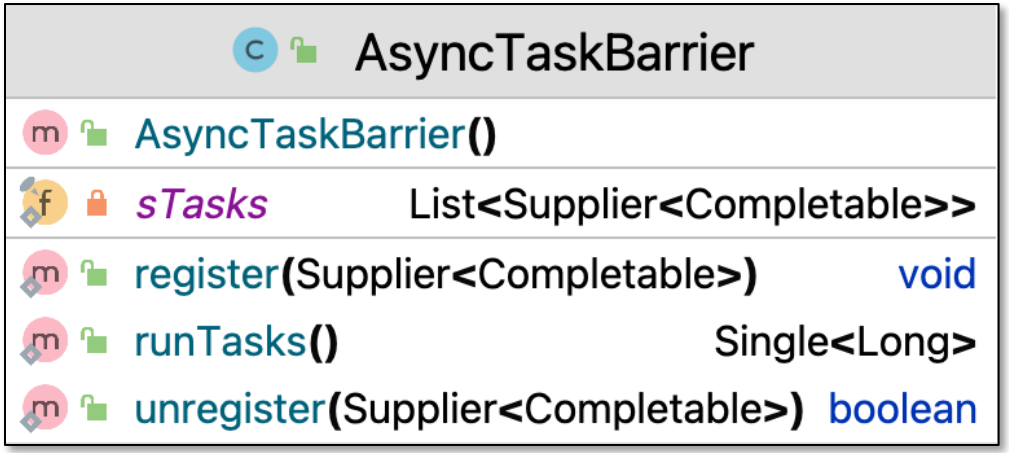
Functional Interface:

This is a functional interface and can therefore be used as the assignment target for a lambda expression or method reference.

See reactivex.io/RxJava/3.x/javadoc/io/reactivex/rxjava3/functions/Supplier.html

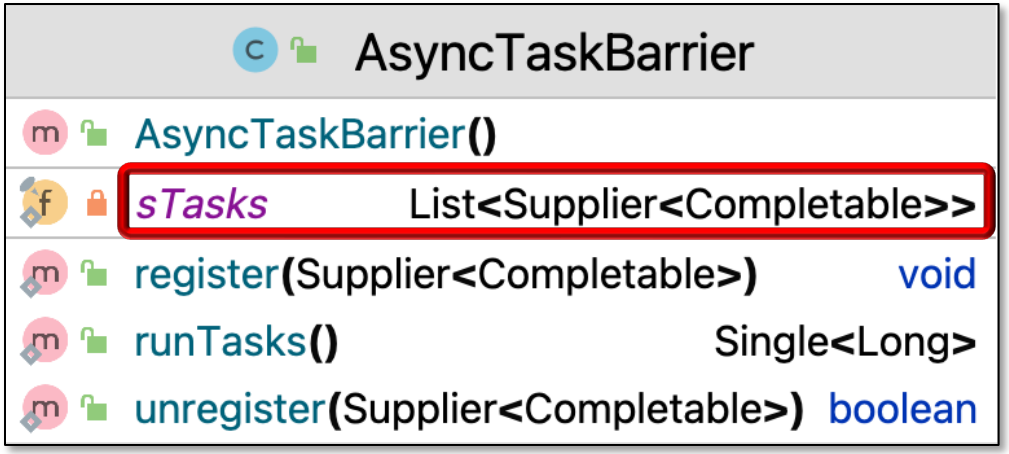
The AsyncTaskBarrier Class API

- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks



The AsyncTaskBarrier Class API

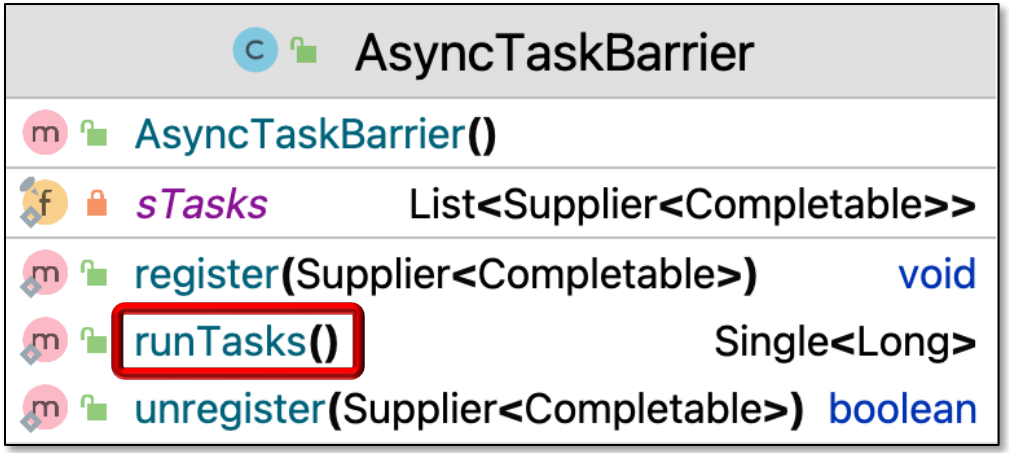
- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks
  - It provides methods to register & unregister tasks
    - These tasks are stored in a List



See docs.oracle.com/javase/8/docs/api/java/util/List.html

The AsyncTaskBarrier Class API

- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks
  - It provides methods to register & unregister tasks
  - It also provides a method that runs all registered tasks (a)synchronously



The AsyncTaskBarrier Class API

- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks
  - It provides methods to register & unregister tasks
  - It also provides a method that runs all registered tasks (a)synchronously
    - This method doesn't block

AsyncTaskBarrier		
AsyncTaskBarrier()		
sTasks	List<Supplier<Completable>>	
register(Supplier<Completable>)		void
runTasks()	Single<Long>	
unregister(Supplier<Completable>)		boolean

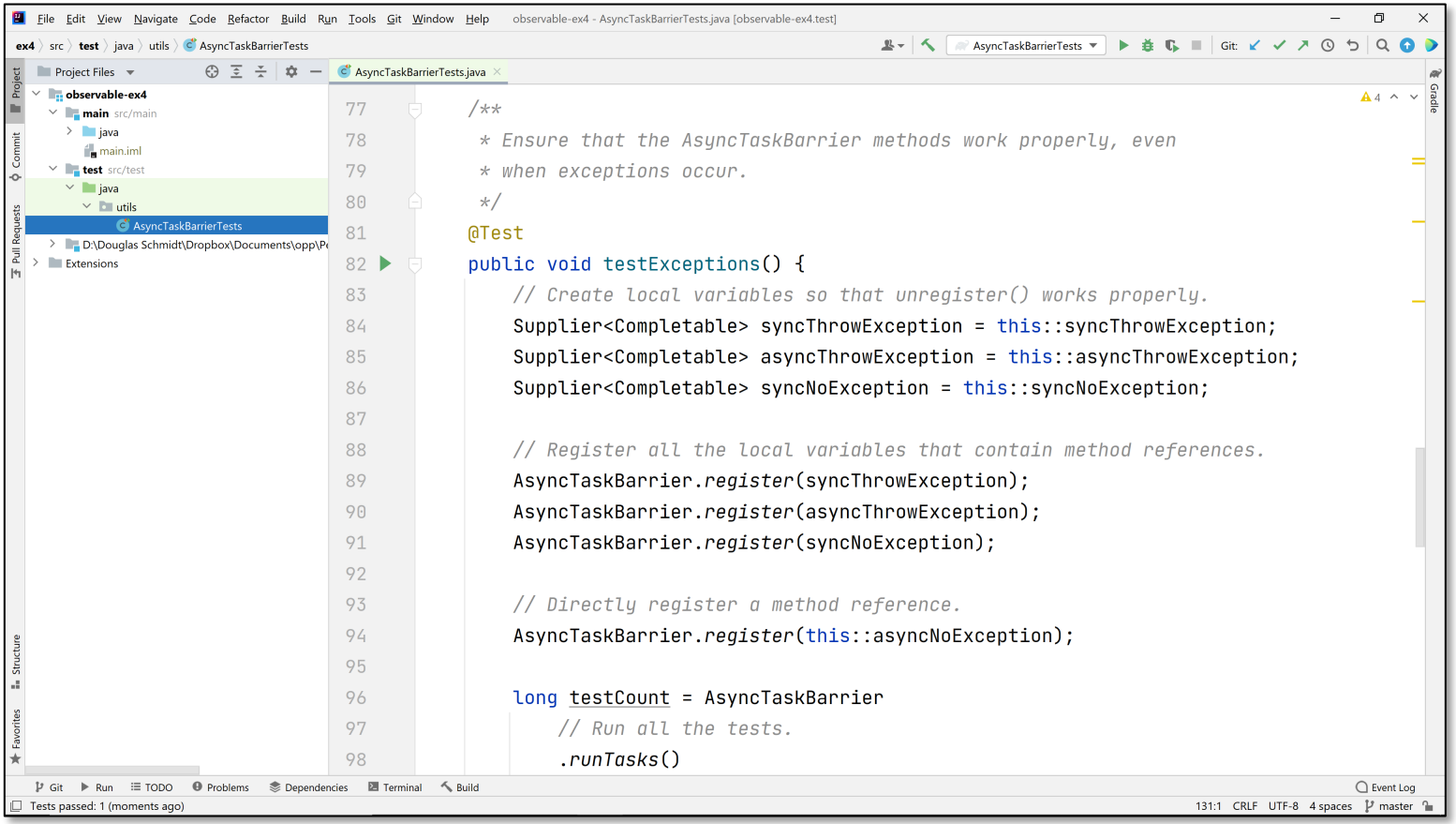
The AsyncTaskBarrier Class API

- The AsyncTaskBarrier API contains methods that register, unregister, & (a)synchronously run tasks
  - It provides methods to register & unregister tasks
  - It also provides a method that runs all registered tasks (a)synchronously
    - This method doesn't block
    - When combined with Single. blockingGet() the calling thread won't exit until all asynchronous task processing completes

AsyncTaskBarrier		
AsyncTaskBarrier()		
sTasks	List<Supplier<Completable>>	
register(Supplier<Completable>)		void
runTasks()	Single<Long>	
unregister(Supplier<Completable>)		boolean

Applying the AsyncTask Barrier in Practice

Applying the AsyncTaskBarrier in Practice



See Reactive/Observable/ex4/src/test/java/utils/AsyncTaskBarrierTests.java

End of Implementing the AsyncBarrierTask Framework Using RxJava (Part 1)