rxjava-examples (0.0.2)

Maksim Kostromin

Version 0.0.2, 2018-07-08 13:11:57 UTC

Table of Contents

Travis CI status: [Build Status]

Chapter 1. Introduction

Read project reference documentation

projects

1. stock (rxjava reactive getting stated)

build

```
./mvnw
./gradlew
```

inks:

1. talk

Chapter 2. Implementation

2.1. gradle-starter

```
cd spring-akka-gradle/
./gradlew
bash build/libs/*.jar
```

```
@Slf4j
public class PrinterActor extends AbstractActorWithTimers {

  @Getter
  @RequiredArgsConstructor
  public static class EchoMessage {
    final String message;
  }

  @Override
  public Receive createReceive() {
    return receiveBuilder()
        .match(EchoMessage.class, echoMessage -> {
          final String message = echoMessage.message;
          System.out.println("message = " + message);
          log.info("received message: {}", message);
          this.getContext().getSystem().log().info("received: {}", message);
        })
        .build()
        ;
  }
}
```

```
@RequiredArgsConstructor
public class GreetingActor extends AbstractActorWithTimers {

  final ActorRef printerActor;

  @Getter
  @RequiredArgsConstructor
  public static class EchoMessage {
    final String message;
  }

  @Override
  public Receive createReceive() {
    return receiveBuilder()
        .match(Object.class, anyObject -> {
          System.out.println("sending message..");
          getContext().getSystem().log().info("O.oOooOOOoOO__");
          printerActor.tell(new EchoMessage("olololo!"), self());
        })
        .build()
        ;
  }
}
```

```
@Slf4j
@Configuration
public class AkkaConfig {

  @Bean
  public ActorSystem actorSystem() {
    return ActorSystem.create("spring-akka-gradle-system");
  }

  @Bean
  public ActorRef printerActor(final ActorSystem actorSystem) {
    return actorSystem.actorOf(Props.create(PrinterActor.class), "PrinterActor");
  }

  @Bean
  public ActorRef greetingActor(final ActorSystem actorSystem, ActorRef printerActor)
{
    return actorSystem .actorOf(Props.create(GreetingActor.class, printerActor),
"GreetingActor");
  }

  @Bean
  public ApplicationRunner actorsApp(final ActorRef greetingActor) {
    return args -> {
      actorSystem().log().info("just greeting...");
      greetingActor.tell(new PrinterActor.EchoMessage("ololo!"), ActorRef.noSender());
    };
  }
}
```

2.2. gradle-starter

```
mkdir gradle-starter
cd gradle-starter/
gradle init --type java-application
```

```
package daggerok;

import io.reactivex.Flowable;

public class App {
  public static void main(String[] args) {
    Flowable.fromArray(args)
            .subscribe(System.out::println);
  }
}
```

build

```
package daggerok

import io.reactivex.rxkotlin.subscribeBy
import io.reactivex.rxkotlin.toObservable

fun main(args: Array<String>) {
  val list = listOf("Alpha", "Beta", "Gamma", "Delta", "Epsilon")

  list.toObservable() // extension function for Iterables
      .filter { it.length >= 5 }
      .subscribeBy(  // named arguments for lambda Subscribers
          onNext = { println(it) },
          onError = { it.printStackTrace() },
          onComplete = { println("Done!") }
      )
}
```

Initially generated by using generator-jvm yeoman generator (kotlin)

2.3. reactivex-api

build

```
@Slf4j
public class App {
  public static void main(String[] args) {
    Flowable.just("ololo", "trololo")
            .subscribe(log::info);
  }
}
```

api

```
Flowable | Publisher | Subscriber
Maybe | MaybeSource | MaybeObserver
Completable | CompletableSource | CompletableObserver

Observable | ObservableSource | Observer
Observers
Operations
Schedule
Single | SingleSource | SingleObserver
Subject
```

Initially generated by using generator-jvm yeoman generator (java)

2.4. javaee-starter

build

```
./mvnw -pl javaee-starter clean package com.dkanejs.maven.plugins:docker-compose-
maven-plugin:1.0.1:up
./mvnw -pl javaee-starter com.dkanejs.maven.plugins:docker-compose-maven-
plugin:1.0.1:down

./gradlew clean build :javaee-starter:composeUp
./gradlew :javaee-starter:composeDown
```

2.5. starter

build

```
./mvnw -pl javaee-starter clean package com.dkanejs.maven.plugins:docker-compose-
maven-plugin:1.0.1:up
./mvnw -pl javaee-starter com.dkanejs.maven.plugins:docker-compose-maven-
plugin:1.0.1:down

./gradlew clean build :javaee-starter:composeUp
./gradlew :javaee-starter:composeDown
```

2.6. stock

build

```
./mvnw -pl javaee-starter clean package com.dkanejs.maven.plugins:docker-compose-
maven-plugin:1.0.1:up
./mvnw -pl javaee-starter com.dkanejs.maven.plugins:docker-compose-maven-
plugin:1.0.1:down

./gradlew clean build :javaee-starter:composeUp
./gradlew :javaee-starter:composeDown
```

Chapter 3. Links

- GitHub repo
- GitHub pages