

# reactor-examples (0.0.1)

Maksim Kostromin

Version 0.0.1, 2018-10-27 03:06:29 UTC

# Table of Contents

- 1. Introduction ..... 1
- 2. Implementations ..... 2
- 3. Links ..... 4

# Chapter 1. Introduction

Read [project reference documentation](#)

*docker / docker-compose*

```
./mvnw  
./mvnw package docker-compose:up -P docker  
./mvnw docker-compose:down -P docker  
  
./gradlew  
./gradlew composeUp  
./gradlew composeDown
```

# Chapter 2. Implementations

links:

- [Helpful presentation](#)
- [Reactor intro](#)

*build*

```
package daggerok;

import lombok.extern.slf4j.Slf4j;
import reactor.core.publisher.Flux;

@Slf4j
public class App {
    public static void main(String[] args) {
        Flux.range(0, 10)
            .map(integer -> integer * integer)
            .filter(integer -> integer % 3 != 0)
            .buffer(3)
            .subscribe(integers -> log.info("result: {}", integers));
    }
}
```

*build*

```
./mvnw -p ./map-filter-buffer/pom.xml
java -jar ./map-filter-buffer/target/*-all.jar

# gradle jar
./gradlew -b ./map-filter-buffer/build.gradle
java -jar ./map-filter-buffer/build/libs/*-all.jar
```

links:

- [Helpful presentation](#)
- [Reactor intro](#)

*build*

```
package daggerok;

import lombok.extern.slf4j.Slf4j;
import reactor.core.publisher.Flux;

@Slf4j
public class App {
    public static void main(String[] args) {
        Flux.range(0, 10)
            .map(integer -> integer * integer)
            .filter(integer -> integer % 3 != 0)
            .retry()
            .subscribe(integers -> log.info("result: {}", integers));
    }
}
```

*build*

```
./mvnw -p ./map-filter-retry/pom.xml
java -jar ./map-filter-retry/target/*-all.jar

# gradle jar
./gradlew -b ./map-filter-retry/build.gradle
java -jar ./map-filter-retry/build/libs/*-all.jar
```

# Chapter 3. Links

- [GitHub repo](#)
- [GitHub pages](#)