BO·TT·EGA
IT minds

Program szkolenia:

Reactor i Spring WebFlux: programowanie reaktywne

Informacje:

Nazwa:	**Reactor i Spring WebFlux: programowanie reaktywne**
Kod:	**Spring -**
Kategoria:	Spring Framework
Odbiorcy:	developerzy, architekci
Czas trwania:	3 dni
Forma:	10% wykład, 90% ćwiczenia

Szkolenie wprowadza w temat programowania reaktywnego na przykładzie biblioteki Reactor. Kładzie duży nacisk na zrozumienie fundamentalnych pojęć i mechanizmów oraz świadomie stosowanie w rzeczywistych projektach.

Szkolenie ma bardzo praktyczny charakter, uczestnicy najpierw pracują z serią małych ćwiczeń pod okiem instruktora. Na koniec wspólnie przygotujemy większą reaktywną aplikację opartą o framework Spring. Może być to rozproszona baza danych, strumień zdarzeń, itp. - w zależności od potrzeb uczestników.

Zalety szkolenia:

- niemal całe szkolenie to praktyczny warsztat i kodowanie na żywo
- idiomatyczne podejście do programowania reaktywnego, od kontrolera po bazę
- całkowity brak slajdów

BO·TT·EGA
IT minds

Szczegółowy program:

1. Podstawy

- 1.1. Czym jest programowanie reaktywne
- 1.2. CompletableFuture i pule wątków
- 1.3. Biblioteka Reactor
- 1.4. Jak stworzyć strumień
  - 1.4.1. `just()`, `generate()`, `create()`, `fromCallable()`, `fromStream()`
- 1.5. Leniwość, strumienie zimne i gorące
- 1.6. Podstawowe operatory
  - 1.6.1. `map()`, `filter()`, `filterWhen()` `flatMap()`, `handle()`, `take()`, `skip()`
  - 1.6.2. `doOn*()` operators
  - 1.6.3. `window()`, `buffer()`, `distinct()`
  - 1.6.4. `cast()`, `ofType()`, `index()`
  - 1.6.5. `timestamp()`, `elapsed()`
  - 1.6.6. `zip()`, `merge()`
- 1.7. Obsługa błędów
  - 1.7.1. `timeout()`, `retry*()`, `retryBackoff()`
  - 1.7.2. `onError*()`
- 1.8. Blokowanie i wielowątkowość
  - 1.8.1. `subscribeOn()`, `publishOn()`
  - 1.8.2. `parallel()`
- 1.9. Testowanie jednostkowe

2. Zaawansowane

- 2.1. Nieblokujący kod

BOTTEGA IT minds