

Kod szkolenia: **SCALA/J**

Tytuł szkolenia: **Programowanie obiektowo-funkcyjne w języku Scala dla programistów Javy**

Dni: 4

Opis:

Adresaci szkolenia

Programiści Javy zainteresowani budową aplikacji z wykorzystaniem języka i ekosystemu Scala lub tematyką programowania funkcyjnego.

Cel szkolenia

Uczestnicy szkolenia poznają możliwości języka Scala, dobre praktyki programowania oraz polecane narzędzia wspierające język Scala. Zbudują też przykładową aplikację z interfejsem REST łączącą prezentowane na szkoleniu tematy.

Mocne strony szkolenia

Forma warsztatowa, wiele zadań z przygotowanymi testami które pozwalają na szybką weryfikację poprawności rozwiązania.

Wymagania

- Znajomość programowania obiektowego
- Doświadczenie w programowaniu w języku Java

Specjalne wymagania techniczne

- Komputer (8GB RAM i 4-core CPU).
- Java 8 SDK
- IDE (preferowane IntelliJ IDEA) z wtyczką do języka Scala.

Parametry szkolenia

4*8 godzin (4*7 godzin netto) wykładów i warsztatów (z wyraźną przewagą warsztatów).

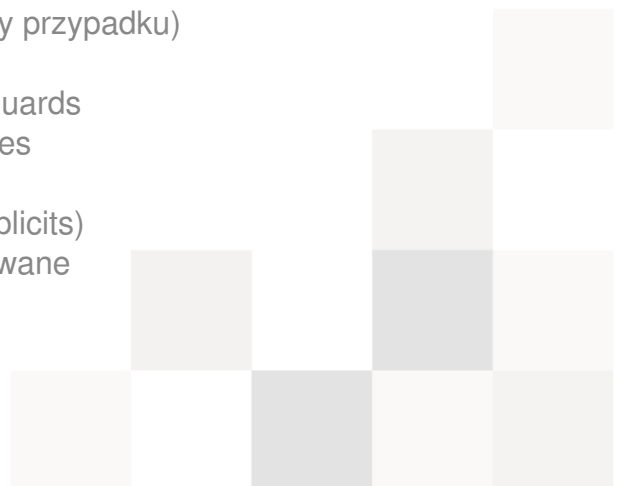
Każdy dzień szkolenia będzie składał się z krótkiego wykładu i warsztatów, podczas których uczestnicy będą wykonywać liczne ćwiczenia na bazie wcześniej przygotowanego kodu. Ostatniego dnia kursanci będą mieli za zadanie zbudowanie w pełni funkcjonalnej aplikacji

oferującej REST API.

Wielkość grupy: maks. 8-10 osób

Program szkolenia:

1. Wstęp
2. Konfiguracja środowiska i pierwszy program
 - Interpreter (REPL)
 - IDE
 - Pierwszy program
 - Wyrażenia (statement vs expression)
 - Simple Build Tool (SBT) oraz organizacja kodu źródłowego
3. Podstawy programowania obiektowego w Scali
 - Klasy i cechy (traits)
 - Obiekty
 - Dziedziczenie
 - Pola mutowalne i niemutowalne
 - Kontrola widoczności
 - Case class i obiekty towarzyszące
 - Operatory
 - Hierarchia typów
 - Nadpisywanie funkcji (override)
4. Kolekcje i programowanie funkcyjne
 - Kolekcje w bibliotece standardowej
 - Funkcje wyższego rzędu
 - Rozwijanie funkcji (currying)
 - Parametry przekazywane przez nazwę
 - Kolekcje mutowalne i niemutowalne
 - Popularne operacje na kolekcjach (tail, head, zip itp)
5. Testowanie aplikacji
 - Specs2, ScalaTest, ScalaCheck
 - Mockowanie
 - ScalaTest DSL
 - ScalaTest matchers
6. Instrukcja for - for loop/for comprehension
7. Dopasowanie wzorców - Pattern matching
 - Proste zastosowania Some/None
 - Dopasowanie do case class (pl. klasy przypadku)
 - Wyrażenia regularne
 - Dopasowanie warunkowe - pattern guards
 - Dopasowanie po typie i sealed classes
 - Definiowanie własnych ekstraktorów
8. Klasy, metody i parametry domniemane (Implicits)
9. Simple Build Tool (SBT) - tematy zaawansowane
10. Obsługa wyjątków



- w stylu obiektowym (try/catch)
 - w stylu funkcyjnym (Try/Success/Failure)
11. Współbieżność i leniwe przetwarzanie
 - Wywołania asynchroniczne (futures and promises)
 - kontekst uruchomienia (execution context)
 - leniwość (lazy val)
 - strumienie
 12. Klasy generyczne
 - Wariacje
 - Ograniczanie parametrów typów generycznych
 13. Integracja z kodem Javy
 14. Przykładowa aplikacja
 - Przegląd najpopularniejszych frameworków
 - Implementacja interfejsu REST
 - Połączenie z bazą danych
 - Języki dziedzinowe - DSL

