

Kod szkolenia: **JAVA/MODERN**

Tytuł szkolenia: **Nowoczesna Java od v. 8.0**

Modern Java

Dni: 2

Opis:

Cel szkolenia

Celem szkolenia jest zapoznanie uczestników z nowymi funkcjonalnościami języka Java (od wersji 9 w górę), maszyny wirtualnej, a także narzędziami developerskimi. Zdobyte umiejętności pozwolą kursantom na bezproblemową migrację na nowszą wersję Javy w swoim projekcie oraz na zdobycie argumentów za taką migracją dla osób decyzyjnych.

Mocne strony szkolenia

Praktyczne wykorzystanie nowych funkcjonalności języka Java oraz przetrenowanie migracji. Skupienie się na rzeczach najważniejszych dla programistów - "syntactic sugar"

Grupa docelowa

Programista

Adresaci szkolenia

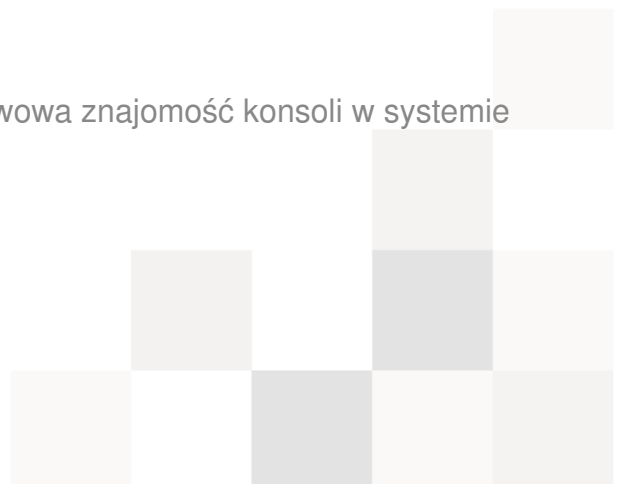
Programiści chcący:

- zapoznać się z nowościami w języku Java oraz JVM
- zdobyć argumenty do wdrożenia tych nowości w swoich projektach

Wymagania dla uczestników

Umiejętność programowania w języku Java Podstawowa znajomość konsoli w systemie Windows / Linux

Parametry szkolenia



2 * 8h (2 * 7h netto)

Wykłady i warsztaty, z przewagą warsztatów

maksymalna liczba uczestników: 8 - 10

Program szkolenia:

1. Zmiany w ekosystemie Javy od wersji 9 w górę (0.5h)
 - Nowy proces releasowania Javy
 - Kwestie licencyjne
2. JShell (Java 9) (0.5h)
 - Uruchamianie (w tym ustawianie classpath i modułów)
 - Definiowanie zmiennych, metod, klas
 - Zapisywanie / ładowanie sesji
 - Nawigacja i edycja
 - Korzystanie z wbudowanych poleceń JShell
3. System modułów - projekt Jigsaw (Java 9) (1h)
 - Przyczyny wprowadzenia modułów
 - Zasady działania systemu modułów
 - Definiowanie modułu (właściwości, zasady dostępu)
 - Budowanie modułu (kompilacja, pakowanie do JARa)
 - Uruchamianie modułu (przekazywanie parametrów, ładowanie zasobów)
 - Debugowanie programu w wersji modułowej
4. Modularyzacja aplikacji - typowe pułapki i problemy (1h)
 - Moduły JEE
 - Dostęp do wewnętrznych API
 - Podzielone pakiety
 - Moduł nienazwany - czyli co się dzieje z legacy code
 - Strategie modularyzacji
5. Warsztat I - upgrade istniejącej aplikacji do JDK 9 oraz modularyzacja (3h)
6. Uruchamianie programów Javy jako skryptów (Java 11) (1h)
 - Wprowadzenie i przykład użycia
 - Argumenty
 - Wiele klas w pliku
 - Użycie z modułami
 - Shebang - czyli jak program w Javie staje się skrypcem powłoki
7. Nowy klient HTTP (Java 11) (1h)
 - Podstawy użycia - prosty request / response
 - Wysyłanie i odbieranie body
 - Requesty asynchroniczne
 - Obsługa WebSockets
8. Preview features (Java 12) (0.5h)
 - Koncepcja tzw. ryzykownych (wycofywalnych) zmian w języku Java
 - Nowe parametry poleceń javac i java
9. Zmiany składniowe (Java 9-12) (1h)

- try-with-resources w nowej odsłonie (Java 9)
- Autodetekcja typów zmiennych lokalnych (Java 10)
- Autodetekcja typów zmiennych w wyrażeniach lambda (Java 11)
- Nowa wersja instrukcji switch (Java 12)

10. Nowe funkcje w standardowym API (Java 9 - 12) (1h)

- Metody fabryczne kolekcji
- Nowe metody w klasie Stream
- Niemodyfikowalne kolekcje
- Dodatkowe metody klasy String
- Process API
- inne

1. Warsztat II - upgrade istniejącej aplikacji do JDK 12 z wykorzystaniem nowych API i składni (3h)
2. Propozycje nowych funkcjonalności w kolejnych wersjach Javy (0.5 h)

