Java Developer (przyspieszony I + II) (kod: K-JAVA-FAST-1-2)

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

Opis i cel kursu

Solidny kurs Javy dla już programujących w innych językach. Dobre podstawy oraz przekrój różnych zastosowań, z większym naciskiem na backend (bazy danych, aplikacje strony serwera: Web, REST, SOAP), ale z uwzględnieniem także aplikacji desktopowych (z interfejsem graficznym oraz tekstowym).

Celem kursu jest poznanie języka i platformy Java w zakresie wystarczającym do wejścia w pierwsze realne projekty. Solidne poznanie podstaw jest niezbędne, aby dobrze rozumieć zasady działania tego języka i platformy. Na naszym kursie obejmuje to m.in. tematy wątków i synchronizacji. Z drugiej strony, potrzebna jest szeroka orientacja w temacie technologii, narzędzi i zastosowań, aby orientować się co do możliwości, jakie dają. Na tym kursie obejmuje to:
— przetwarzanie danych (kolekcje, strumienie i lambdy, obsługa plików),
— dostęp do baz danych (JDBC, JPA/Hibernate, Spring Data),
— usługi REST na dwa sposoby: za pomocą Spring oraz za pomocą Jakarta EE (JAX-RS),
— usługi SOAP za pomocą Jakarta EE (JAX-WS),
— klasyczne aplikacje webowe za pomocą serwletów oraz Springa (ale zgodnie ze zmianą trendów w realnych projektach, zmniejszamy ilość czasu na to przeznaczoną),
— testy jednostkowe: JUnit, Mockito, AssertJ.

Zajęcia prowadzone są przy komputerach i mają charakter warsztatowy, zgodnie z naszym mottem: *teorię też da się pokazać na praktycznym przykładzie*.

Program

1. **Język i platforma Java – wprowadzenie**
   — Przegląd wersji i edycji. Instalacja i konfiguracja platformy Java SE
   — Typy proste i referencyjne, operatory arytmetyczne i logiczne, instrukcje sterujące
   — Zmienne: rodzaje, inicjalizacja, zasięg i przesłanianie
   — Standardy nazewnicze
   — Struktura projektu, pakiety, kompilacja, bajtkod, ładowanie klas, tworzenie JAR-ów
   — Praca z Javą w konsoli oraz w IDE, przegląd dostępnych IDE
   — Java na tle innych języków programowania
   — Przegląd zastosowań Javy
2. **Programowanie obiektowe w Javie**
   — Składowe klasy: pola, metody, konstruktory, bloki inicjalizacyjne
   — Obiekty i referencje, przekazywanie parametrów, klasy mutowalne i niemutowalne
   — Przeciążanie metod
   — Dziedziczenie i nadpisywanie metod
      — Klasy abstrakcyjne, finalne i zamknięte (*sealed*)
   — Interfejsy
      — Implementacje metod w interfejsach
   — Poziomy widoczności, metody dostępowe, wzorzec JavaBean
   — Kontekst statyczny i instancyjny
   — Wyjątki
   — Najważniejsze wzorce projektowe
   — Uproszczone sposoby tworzenia typowych „nudnych” klas:

   - Rekordy
   - Biblioteka Lombok
3. **Uniwersalne elementy Java Standard Edition**
   - Tablice: zasady używania, metody narzędziowe
   - Kolekcje: listy, kolejki, zbiory i słowniki
     - Zasady wyboru właściwego rodzaju i implementacji kolekcji
   - Zasady porównywania obiektów: metody equals i hashCode, interfejsy Comparable i Comparator, sortowanie danych
   - Klasy opakowujące („wrappers") i inne klasy reprezentujące liczby
   - Użyteczne klasy narzędziowe
   - Klasy String, StringBuilder i inne możliwości przetwarzania tekstu
   - Wyrażenia regularne (regex)
   - Wejście/wyjście i dostęp do plików
   - Obsługa daty i czasu oraz ustawień lokalnych
4. **Wątki i synchronizacja**
   - Struktura maszyny wirtualnej Java (JVM)
   - Tworzenie i konfiguracja wątków
   - Zagrożenia wynikające ze współbieżnego wykonania kodu
   - Synchronizacja za pomocą mechanizmów języka Java
   - Klasy narzędziowe i gotowe schematy synchronizacji (locki, semafory, synchronizowane kolekcje, pule wątków)
5. **Zaawansowane elementy języka Java**
   - Typy wyliczeniowe (enum)
   - Typy generyczne
   - Klasy zagnieżdżone, lokalne i anonimowe
   - Elementy programowania funkcyjnego w Javie
     - Wyrażenia lambda i referencje do metod
     - Strumienie, przetwarzanie danych za pomocą strumieni i kolektorów.
     - Strumienie równoległe – korzyści i zagrożenia; Spliteratory
6. **Aplikacje z interfejsem graficznym** (podstawy)
   - Przegląd technologii GUI dostępnych w Javie
   - Swing: komponenty, panele, layouty
   - Obsługa zdarzeń; różne typy zdarzeń, wątki w Swingu
7. **Dostęp do baz danych z aplikacji Java**
   - JDBC: bezpośredni dostęp do bazy danych
     - Zadawanie zapytań i wydawanie poleceń modyfikujących dane
     - Obsługa parametrów, zapobieganie *SQL injection*
     - Zaawansowane możliwości JDBC: modyfikacja poprzez wynik, wywoływanie procedur składowanych, transakcje
   - Jakarta Persistence API (JPA / Hibernate)
     - Pomysł ORM, generowanie klas na podstawie tabel lub odwrotnie
     - Klasy encji, adnotacje JPA, zależności między encjami
     - Odczyt wszystkich i pojedynczych rekordów
     - Tworzenie i modyfikacja rekordów; transakcje
     - Zapytania JPQL, parametry, kryteria
8. **Serwlety**
   - Czym jest Java Enterprise Edition / Jakarta EE?
     - Specyfikacje, implementacje, serwery aplikacji
   - Technologia serwletów
     - Obsługa zapytań i tworzenie odpowiedzi HTTP
     - Nagłówki, parametry, ciasteczka
     - Listenery i filtry
   - Podstawy Java Server Pages (JSP – szablony stron WWW)

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

- Skryptlety, wyrażenia EL (*expression language*), dostęp do komponentów JavaBean
- Biblioteki tagów (taglibs), JSTL Core
- Architektura MVC za pomocą serwletów i JSP

9. **Usługi sieciowe w Jakarta EE**
   - Koncepcja, rodzaje i zastosowania usług sieciowych (*web services*)
   - Usługi typu SOAP: JAX-WS
     - Klasa i interfejs usługi
     - Obsługa XML w Javie, w tym standard JAXB
     - Strona klienta usługi SOAP, w tym generowanie kodu na podstawie WSDL
   - Usługi typu REST: JAX-RS
     - Klasy zasobów i mapowanie adresów
     - Rola i właściwa implementacja zapytań GET, POST, PUT, DELETE
     - Obsługa różnych formatów danych, w tym JSON i XML oraz własne rozszerzenia na przykładzie PDF
10. **Dodatkowe narzędzia**
    - Narzędzie budowania Maven (+pobieżnie Ant i Gradle)
      - Projekt wielomodułowy Maven przy okazji serwera i klienta usługi sieciowej
    - Popularne biblioteki open-source
    - Testy jednostkowe JUnit
      - Dodatkowe biblioteki związane z testowaniem: Mockito, AssertJ
    - Budowanie aplikacji Java na serwerze Jenkins
11. **Spring**
    - Tworzenie i uruchamianie projektu Spring Boot
    - Framework Spring MVC
      - Klasy kontrolera, mapowanie adresów, obsługa parametrów
      - Generowanie treści HTML za pomocą szablonów JSP i Thymeleaf (na szybko - głównie gotowe przykłady)
      - Usługi typu REST za pomocą Spring
    - Fundamentalne zasady działania Spring
      - Komponenty – tworzenie, priorytety, zakresy
      - Wstrzykiwanie zależności (*dependency injection*) – za pomocą adnotacji i przez konstruktor
      - Odwrócenie kontroli (*inversion of control*, IoC)
    - Spring Data
      - Stosowanie JPA/Hibernate w ramach Spring
      - Interfejsy typu @Repository i magiczne nazwy metod
      - „Natywne” zapytania SQL
    - Podstawy Spring Security
      - Definiowanie użytkowników i haseł lokalnie oraz w bazie danych
      - Kontrola dostępu do elementów aplikacji
    - Przegląd dodatkowych możliwości Spring
      - Usługi REST za pomocą Jersey (JAX-RS)
      - Schematyczne usługi REST za pomocą RestRepositories

Przeznaczenie i wymagania

Kurs przeznaczony jest dla programistów innych języków, którzy w dowolnym języku potrafią już samodzielnie pisać programy, niekoniecznie złożone. Wymagamy dobrej znajomości takich pojęć jak zmienna i funkcja oraz umiejętności pisania prostych algorytmów z wykorzystaniem pętli i struktur danych (np. tablic). Przyda się znajomość chociaż podstaw programowania

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

obiektowego, SQL i HTML. Kurs cieszy się dużą popularnością wśród osób używających do tej pory PHP, C i C++, C#, PL/SQL lub T-SQL, JavaScript, Python czy nawet VBA, ale także znających tylko podstawy Javy, a chcących nauczyć się elementów zaawansowanych.

Certyfikaty

Uczestnicy szkolenia otrzymują imienne certyfikaty sygnowane przez ALX.

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl