

Comarch Bootcamp

Junior Java Developer

Cele szkolenia

Kurs ma na celu przygotowanie do roli Młodsze Programisty Java oraz podjęcia pracy w charakterze programisty. Podczas kursu uczestnik buduje pierwsze programy komputerowe, korzysta w stopniu zaawansowanym z narzędzi niezbędnych w codziennej pracy programisty, wybiera zaawansowane funkcje języka Java usprawniające produkowanie programów oraz korzysta wielu technologii pozwalających na tworzenie profesjonalnych projektów komercyjnych.

Efektem kursu będzie przygotowanie samodzielne projektu o charakterze zaawansowanym.

Umiejętności

Po ukończeniu intensywnego kursu Bootcamp - Junior Java Developer uczestnik/czka będzie:

- Tworzyć aplikacje w języku Java
- Korzystać z wielu bibliotek wbudowanych języka Java
- Korzystać z systemu kontroli wersji GIT
- Używać narzędzia Maven do zarządzania projektem.
- Stosować się do zasad programowania obiektowego
- Zastosować programowanie aspektowe
- Tworzyć aplikacje webowe przy użyciu Spring
- Wykorzystywać wzorzec odwrócenia zależności (IoC)
- Tworzyć i posługiwać się relacyjnymi bazami danych
- Używać Hibernate oraz JDBC do operacji bazodanowych z poziomu Java
- Tworzyć aplikacje okienkowe przy pomocy JavaFX
- Tworzyć testy automatyczne aplikacji Java
- Zarządza swoją pracą w metodyce Agile
- Tworzyć i konsumować Web Services REST oraz SOAP
- Tworzyć podstawowe diagramy UML

Profil uczestników

Bootcamp dedykujemy wszystkim, którzy chcą podjąć pracę jako programista JAVA.

Szczegółowy program szkolenia

Narzędzia deweloperskie:

Przed napisaniem jakiegokolwiek kawałka kodu niezbędne jest poznanie narzędzi i oprogramowania, dzięki któremu wytwarzanie oprogramowania będzie proste i przyjemne. Konieczne jest również zapoznanie się z

zasadą działania aplikacji napisanych w języku Java i sposobie zamieniania kodu napisanego przez programistę w działający program.

- Środowisko programistyczne IntelliJ
 - instalacja
 - obsługa
- narzędzia języka Java
 - JDK
 - JRE
 - JVM

Java – podstawy:

Programowanie w dowolnym języku składa się z pewnych podstawowych koncepcji i założeń, dlatego naukę programowania rozpoczynamy od zapoznania się właśnie z nimi.

- Pierwsza aplikacja
- struktura aplikacji w Java
 - struktura i znaczenie katalogów
 - pakiety
 - External Libraries
 - klasa startowa
- uruchamianie projektu
 - kompilacja
 - walidacja
 - uruchamianie pliku .jar
- typy danych w Java
 - liczbowe
 - zmiennoprzecinkowe
 - znaki i łańcuchy znaków
 - wartości logiczne
- zmienne
 - definicja zmiennych
 - przypisywanie, odczytywanie i zmiana wartości
 - inicjalizacja zmiennych
- operatory wbudowane w język Java
- instrukcje warunkowe
 - if
 - switch
- pętle
 - for
 - foreach
 - while
 - do-while
 - instrukcje break i continue
- tablice
 - definicja tablic

- inicjalizacja tablic
- iteracja po tablicach
- tablice dwuwymiarowe oraz wielowymiarowe

Obiektowość w Java:

Kiedy tworzone programy stają się coraz większe i bardziej skomplikowane podstawowe mechanizmy programowania przestają nam wystarczać. Tu do gry wchodzi idea programowania obiektowego. Musimy zapoznać się z tą techniką programowania, zasadami obiektowości w języku Java oraz narzędziami języka Java związanymi z obiektowością

- klasy i obiekty
 - zawartość klas
 - konstrukcja obiektów
 - typy wyliczeniowe
 - referencje
 - typy niemutowalne
 - klasy wewnętrzne
- metody
 - definicja metod
 - funkcje i procedury
 - metody inline
 - przekazywanie parametrów
 - wartości zwracane
 - przesłanianie metod
 - przeciążanie metod
- enkapsulacja
 - modyfikatory dostępu
 - getter'y
 - setter'y
 - ukrywanie metod
- dziedziczenie
- polimorfizm
- abstrakcja w języku Java
 - interfejsy
 - implementacja interfejsów
 - klasy abstrakcyjne
 - metody abstrakcyjne
 - klasy anonimowe
- static oraz final
- konwersja i rzutowanie

Narzędzia wbudowane w Java:

Oprócz samego sterowania aplikacją niezwykle ważne są również sposoby przetwarzania i przechowywania danych w pamięci komputera. Z tego właśnie powodu musimy zapoznać się ze strukturami danych w języku Java, sposobami wielowątkowego, równoległego przetwarzania tych danych i sposobem zarządzania pamięcią

w Java. Poza działającą aplikacją wymiernym efektem pracy programisty jest jakość kodu, który wytwarza - poznamy również mechanizmy pozwalające upraszczać, skracać i porządkować kod który piszemy.

- kolekcje
 - listy
 - zbiory
 - mapy
 - operacje na kolekcjach
 - sortowanie kolekcji
- wyjątki
 - przechwytywanie wyjątków
 - przekazywanie wyjątków
 - tworzenie własnych wyjątków
 - podział wyjątków
- wielowątkowość
 - thread
 - runnable
 - executor'y wątków
 - pule wątków
 - synchronizacja wątków
- strumień wejścia/wyjścia
- obsługa plików
- typy generyczne
- interfejs funkcyjny
- strumień oraz wyrażenia lambda
- wartości opcjonalne w Java
- zarządzanie pamięcią –GC
- debugowanie aplikacji

Wykorzystanie AI w programowaniu:

Sztuczna inteligencja zmienia sposób tworzenia oprogramowania automatyzując i przyspieszając pracę programistów. Na przykładzie Copilot - pionierskiego i najpopularniejszego asystenta kodowania - nauczymy się wykorzystywać AI w codziennej pracy programisty.

- inteligentna generacja kodu
 - szybkie odpowiedzi w czasie rzeczywistym
 - generacja pełnych funkcjonalności i komponentów
- wykorzystanie interaktywnego asystenta AI
- generowanie dokumentacji kodu
- Code Explanation
- planowanie wytwarzania kodu i realizacja planu przy pomocy AI
- zarządzanie kontekstem
- wykorzystanie różnych modeli językowych
- techniki promptowania, tworzenie precyzyjnych promptów
- code review przy pomocy AI

Bazy danych:

Oprócz danych przechowywanych w pamięci aplikacji, musimy również przechowywać wiele danych "na stałe". Do tego wykorzystujemy bazy danych. Musimy zapoznać się z działaniem baz danych oraz ich obsługą. Nauczmy się również sposobów łączenia aplikacji napisanych w języku Java z bazami danych wykorzystując nowoczesne biblioteki i API języka Java.

- relacyjne bazy danych
 - uruchamianie serwera baz danych
 - zasada działania
 - tworzenie baz danych
 - relacje
- język SQL
 - wstawianie rekordów do bazy
 - zapytania
 - modyfikacja danych
 - usuwanie rekordów
 - JOIN
 - operacje na zbiorach danych
 - grupowanie wyników
 - sortowanie
 - zapytania zagnieżdżone
 - klucze prywatne
 - auto inkrementacja
 - wartości domyślne
- JDBC
 - połączenie Java z bazą danych
 - operacje na bazie danych przy użyciu Java

Technologie i narzędzia:

Będziemy również potrzebować narzędzi do utrzymywania i zarządzania projektami oraz kontroli wersji oprogramowania. Kiedy nasze aplikacje staną się bardziej skomplikowane sam język Java przestanie nam wystarczać. Nauczmy się wykorzystywać dodatkowe narzędzia zewnętrzne, nie wbudowane w język Java. Poznamy nowe idee i koncepcje tworzenia komercyjnych aplikacji. Aby zobaczyć efekty naszej pracy wprowadzimy interfejs użytkownika do aplikacji Java i napiszemy testy automatyczne sprawdzające poprawność ich działania.

- Maven
 - zarządzanie zależnościami
 - budowanie aplikacji
 - wtyczki do Maven
 - etapy budowania projektu
 - zdalne repozytorium Maven
- GIT
 - tworzenie repozytorium
 - inicjalizacja projektu git
 - dodawanie zdalnego repozytorium do projektu
 - tworzenie commit'ów
 - wrzucanie zmian do zdalnego repozytorium
 - pobieranie repozytorium

- uaktualnianie lokalnych źródeł
 - gałęzie
- Spring Boot
 - uruchamianie kontenera HTTP
 - kontenery wbudowane w Spring
- Spring IoC
 - beany
 - konteksty
 - wstrzykiwanie zależności (Dependency Injection)
- Metody konfiguracji Spring
 - pliki XML
 - Klasy konfiguracyjne
 - Adnotacje
- Spring MVC
 - wzorzec MVC
 - kontrolery
 - szablony widoków
 - mapowanie zapytań
 - metody zapytań HTTP
 - przekazywanie modelu do widoków
- Thymeleaf
- Hibernate
 - konfiguracja hibernate
 - ORM
 - zapytanie w hibernate
 - HQL
 - zapytania przy pomocy kryteriów
 - sesje
 - transakcje
- XML i JSON
- Podstawy front-end'u
 - JS
 - HTML
 - CSS
 - JQuery
 - AJAX
- Testowanie aplikacji w Java
 - JUnit
 - Mockito
 - TDD

WebServices:

Kiedy aplikacje stają się ogromne chcemy zmienić podejście do ich wytwarzania. Korzystamy w metody “dziel i zwyciężaj”, czyli dzielimy nasz wielki projekt na kilka mniejszych programów. Żeby te mniejsze “moduły” naszego systemu mogły ze sobą współpracować musimy nauczyć się sposobów na zaimplementowanie komunikacji między nimi.

- REST
 - definicja API przy pomocy JSONSchema
 - generacja API przy pomocy Maven
 - klient REST API
 - provider REST API

Metodyki wytwarzania oprogramowania i dokumentacje:

Oprócz wytwarzania oprogramowania programista musi również dokumentować wytwarzany kod oraz projekty aplikacji. Nauczymy się, jakie dokumenty funkcjonują w świecie IT i dowiemy się jak przekładać nasz kod na formę papierową w postaci diagramów UML. Poznamy również strategię pracy podczas wytwarzania oprogramowania.

- Scrum
- Dokumentacje w IT
 - LLD
 - HLD
 - CR
 - PID
 - UM
 - DoR
 - DoD
- UML
 - diagramy przypadków użycia
 - diagramy klas
 - diagramy sekwencji
 - diagramy aktywności

Metoda realizacji szkolenia

W ramach naszego intensywnego kursu Java do zrealizowania **będą wyznaczane przez prowadzących zadania** domowe oraz jeden większy projekt (ok 50 roboczo godzin) wykorzystujący umiejętności zdobyte na całym szkoleniu.

Prace domowe

- pozwolą na obycie się z kodem, środowiskiem i zdobyciem podstawowego doświadczenia w pisaniu i projektowaniu oprogramowania.
- pomogą w przyswojeniu dobrych praktyk programowania i zrozumieniu problemów wynikających z niestarannie pisanego kodu oraz nieprzemyślanego projektu.
- pozwalają na utrwalenie wiedzy zdobytej na zajęciach, lepsze zrozumienie poznanych technologii, rozwijają zdolności algorytmiczne oraz projektowe.
- część zadań będzie polegała na samodzielnym zgłębieniu konkretnego tematu i przedyskutowaniu go na zajęciach.

Projekt końcowy to w pełni działająca aplikacja tworzona w metodyce agile. Wymagane jest przygotowanie dokumentacji technicznej aplikacji oraz wykorzystanie jak największej ilości technologii. **Projekt musi spełniać trzy główne założenia – gromadzenie danych, przetwarzanie danych oraz możliwość manipulacji danymi.** Dzięki takiemu podejściu należy wykazać się umiejętnościami pisania przemysłowego kodu back-end'owego, front-end'owego oraz projektowania i implementowania prostych algorytmów. Skończony projekt jest również dobrym elementem składowym do portfolio przyszłego programisty. Temat projektu można wybrać z puli przygotowanej przez trenerów lub po konsultacji i zaakceptowaniu napisać wymyśloną samodzielnie aplikację. Podczas wykonywania zadań oraz projektu kursanci mogą liczyć na pomoc mentorów i konsultować napotkane problemy.

Podczas zajęć z Androida i Springa kursanci tworzą aplikacje od podstaw. Poznają specyficzne technologie, zapoznają się z metodami iteracyjnego rozwijania oprogramowania i realizują scenariusze, z którymi zetkną się w pracy na stanowisku programisty.

Aplikacje utworzone podczas tych modułów będą stanowić **portfolio uczestnika**, potwierdzające umiejętność korzystania ze zdobytej wiedzy, a mentoring trenerski przez doświadczoną osobę w programowaniu Java – odkrywanie i rozwijanie potencjału ucznia.

Liczba dni, liczba godzin szkoleniowych

208 godzin + 4 godziny lekcyjne konsultacji HR

Harmonogram zjazdów:

- 25-26.09.2021
- 02-03.10.2021
- 16-17.10.2021
- 06-07.11.2021

- 20-21.11.2021
- 04-05.12.2021
- 18-19.12.2021
- 08-09.01.2022
- 22-23.01.2022
- 29-30.01.2022
- 12-13.02.2022
- 26-27.02.2022
- 12-13.03.2022
- 19.03.2022 – konsultacje HR

Egzamin

W cenę kursu wliczony jest EGZAMIN

- **1Z0 811 JAVA FOUNDATIONS** (dla uczestników Kierunek Kariera Zawodowa)

Ścieżka rozwoju po szkoleniu

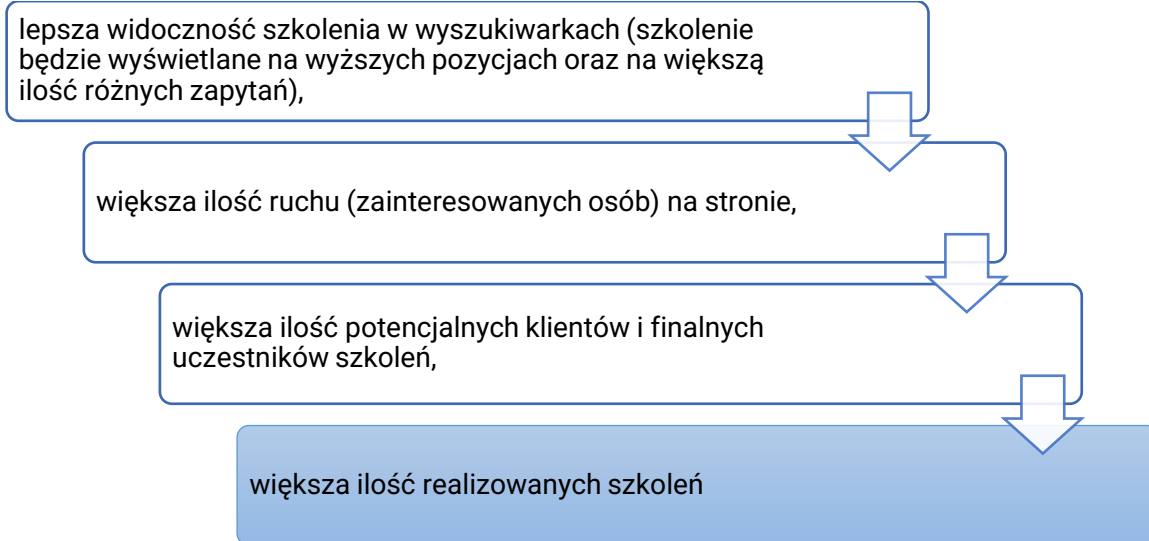
Tworzenie programu z uwzględnieniem SEO - zalecenia

Co to jest SEO?

SEO oznacza optymalizację stron internetowych pod kątem wyszukiwarek np. Google.

Internauci podczas **poszukiwań szkoleń** w wyszukiwarkach wykorzystują **frazy (słowa) lub pytania**. Dobrze zoptymalizowana treść szkolenia powinna **opisywać potencjalnie wykorzystywaną przez internautę frazę, problematykę lub pytanie** dotyczące danego szkolenia.

Dlaczego warto tworzyć programy z uwzględnieniem SEO?



Jak tworzyć dobry jakościowo i kompleksowy opis szkolenia i jego warunków?

Kluczowymi czynnikami wpływającymi na pozycjonowanie szkoleń są:

1. Treść opisu szkolenia -jednym z najważniejszych czynników rankingowych dla Google:

- ✓ **Prosty, przejrzysty, zrozumiały język**
- ✓ **Słownictwo dostosowane poziomem do odbiorcy**
- ✓ **Język korzyści** eksponujący **zalety** uczestnictwa w danym szkoleniu
- ✓ **Zwroty zachęcające do udziału**
- ✓ **Kompleksowość opisu** - szczegółowy opis każdego punktu programu szkolenia. Opis szkolenia zawierający więcej różnych, przydatnych informacji będzie wyświetlać się na większą liczbę różnych zapytań w wyszukiwarce

Najprostszym sposobem określenia jakości i zakresu opisu szkolenia jest **porównanie z konkurencją**

- ✓ Konkurencyjne opisy szkoleń można sprawdzić wpisując w Google **frazy kluczowe** związane z własnym szkoleniem
- ✓ Opisy i warunki szkoleń na stronach **pojawiających się najwyżej w rankingu (np. 5 pierwszych stron, nie wliczając wyników reklamowych Adwords)** zawierają najcenniejsze informacje
- ✓ Należy korzystać z **trybu przeglądarki incognito**, aby zmniejszyć wpływ naszej historii wyszukiwania na wyniki Google.

2. Zawartość ważnych dla tematyki szkolenia fraz i słów

To poprzez frazy kluczowe wyszukujemy w wyszukiwarce internetowej informacje, które nas interesują. Kiedy wpisujemy takie hasła w wyszukiwarkę, ta dobiera **strony internetowe, których treść najbardziej odpowiada szukanej frazie**.

- ✓ **Opis, tytuły i śródtytuły** powinny zawierać w sobie **frazy kluczowe charakterystyczne** dla danego szkolenia i jego tematyki
- ✓ **Warto stosować frazy w różnych odmianach, rodzajach i ich synonimy**
- ✓ Fraza kluczowa może się składać **z jednego słowa lub z ich określonego ciągu**
- ✓ **Liczba fraz i ich umiejscowienie musi być naturalna**, wynikająca z tematyki szkolenia.
- ✓ Nie można sztucznie umieszczać w opisach fraz.

Przykład:

w opisie o "szkoleniu zaawansowanym z excel" powinny pojawiać się sformułowania:

"szkolenie zaawansowane excel"

"szkolenie ms excel na poziomie zaawansowanym"

"szkolenie zaawansowanej obsługi excel"

"szkolenie zaawansowanej obsługi arkuszy kalkulacyjnych"

Ponadto poniższe narzędzia mogą pomóc określić **możliwe frazy kluczowe dla danej problematyki szkoleniowej** oraz zainspirować do opisanie różnych, dodatkowych aspektów związanych ze szkoleniem (Podczas korzystania z tych narzędzi należy określić kraj/język (Polska/język polski).

- <https://ahrefs.com/pl/keyword-generator>
- <https://answerthepublic.com/>
- <https://keywordtool.io/>