

SZKOLENIE ŚREDNIO ZAAWANSOWANE

Programowanie w języku Rust

RUST/PROG

Czas trwania: 3 dni (24h)

Nowoczesny, bezpieczny i niezwykle wydajny język programowania, ogólnego przeznaczenia

Cele szkolenia

- Gruntowne wprowadzenie do składni i najważniejszych elementów języka Rust
- Poznanie i zrozumienie unikalnego modelu zarządzania pamięcią
- Zapoznanie z istniejącym ekosystemem i dostępnymi narzędziami deweloperskimi
- Przedstawienie obowiązujących konwencji i dobrych praktyk programistycznych

Zalety

- Szkolenie ma charakter warsztatowy - duża ilość przykładów i ćwiczeń ułatwiających przyswojenie zdobywanej wiedzy
- Praktyka przed teorią - wszystkie szkolenia technologiczne prowadzone są w formie warsztatowej. Konieczna teoria jest wyjaśniana na przykładzie praktycznych zadań
- Konkretna umiejętność - w ramach każdego szkolenia rozwijamy praktyczne umiejętności związane z daną technologią i tematyką
- Nauka z praktykami - wszyscy trenerzy na co dzień pracują w projektach, gwarantuje to dostęp do eksperckiej wiedzy i praktycznego know-how

Dla kogo?

- Programiści chcący kompleksowo poznać język Rust, poszerzyć posiadane umiejętności, a także poprawić wydajność i bezpieczeństwo wytwarzanego oprogramowania

Wymagania

- Podstawowa umiejętność programowania w dowolnym języku będzie dużym ułatwieniem

Program

1. Podstawy języka
 - a. Charakterystyka, zastosowanie i najważniejsze cechy Rust
 - b. Konfiguracja środowiska deweloperskiego
 - c. Podstawy składni i obowiązujące konwencje kodowania
 - d. Kompilowanie, debugowanie i uruchamianie aplikacji



- e. Dostępne typy danych
 - f. Zmienne, stałe oraz koncepcja niezmienności
 - g. Operatory
 - h. Instrukcje sterujące przepływem
 - i. Wyrażenia, funkcje i domknięcia
 - j. Struktury
 - k. Wyliczenia
2. Model zarządzania pamięcią
- a. Wyjaśnienie koncepcji Ownership
 - b. Referencje, wskaźniki i reguły bezpiecznego zarządzania pamięcią
 - c. Lifetimes
 - d. Smart Pointers
 - e. Rozwiązywanie typowych problemów zgłaszanych przez kompilator
3. Codzienne programowanie w Rust
- a. Praca z kolekcjami
 - b. Pattern matching i jego formy
 - c. Język Rust a podejście obiektowe
 - d. Traits - tworzenie, implementacja, zastosowanie
 - e. Programowanie w stylu funkcyjnym
 - f. Obsługa błędów
 - g. Typy generyczne
 - h. Iteratory
 - i. Organizacja kodu aplikacji z wykorzystaniem modułów
 - j. Istotne elementy SDK
 - k. Obsługa wejścia / wyjścia
 - l. Tworzenie narzędzi działających z poziomu command line
 - m. Testowanie
4. Wybrane zagadnienia zaawansowane
- a. Unsafe Rust
 - b. Makra
 - c. Podstawy wielowątkowości
 - d. Programowanie asynchroniczne
 - e. Foreign functions

