

# Tytuł szkolenia: Programowanie w Go

## Kod szkolenia: GO

### Wprowadzenie

Go to język ze wsparciem wytwarzania **aplikacji mikroserwisowych**. Mechanizm zarządzania wątkami dostarczony w Go pozwala relatywnie niewielkim nakładem pracy budować wielowątkowe systemy w środowisku rozproszonym.

Wyjątkowo prosta, przemyślana, przypominająca modeluraryzację kodu w starych językach proceduralnych koncepcja separacji paczek kodu owocuje w projektach opartych o Go przejrzystą strukturą warstwową systemu, łatwością utrzymania i reorganizacji modułów w przestrzeni całego systemu.

Automatyczne zarządzanie pamięcią, wydajny kompilator i multiplatformowość uzyskana poprzez mechanizm natywnego budowania aplikacji pod wybraną platformę systemową sprawiają, że dużo łatwiej zapewnić wydajność działania wdrażanego systemu.

Warto uczyć się nowych **języków programowania**.

Go niewątpliwie jest jednym z tych, który wzbogaca **warsztat developerski**. Koncepcyjnie składniowo i architektonicznie odzwierciedla kierunek w jakim rozwija się dziś inżynieria oprogramowania.

Praca w Go daje wiele satysfakcji.

### Adresaci szkolenia

Szkolenie skierowane jest do **programistów** chcących poznać nowy **język programowania** i inne podejście do sposobu **budowania aplikacji**, a także do osób nie posiadających doświadczenia zawodowego chcących nauczyć się języka programowania, który w naturalny sposób wprowadzi ich w świat inżynierii oprogramowania i będzie dobrą podstawą do dalszego rozwoju.

### Cel szkolenia

### Czas i forma szkolenia

- 14 godzin (2 dni x 7 godzin), w tym wykłady i warsztaty praktyczne.

## Plan szkolenia

### Charakterystyka języka Go

- a. Zastosowania
- b. Architektura
- c. Studium porównawcze
- d. Popularne frameworki oparte na języku Go

### Ekosystem Go

- a. Konfiguracja środowiska projektowego, narzędzia:
  - 1. Zmienne środowiskowe
  - 2. Mechanizm budowania aplikacji
  - 3. Generowanie plików
  - 4. Praca z zależnościami projektowymi
  - 5. Sterowanie testami
  - 6. Zarządzanie dokumentacją
- b. Struktura projektu
- c. Wsparcie IDE
- d. Profilowanie
- e. Debugging

### Struktura języka

- a. Typy danych, wyrażenia, zmienne, struktury
- b. Struktury kontrolne: if for, switch
- c. Funkcje, metody
- d. Obsługa wyjątków
- e. Kolekcje: tablice, listy, mapy
- f. Interfejsy
- g. Wielowątkowość: procesy, komunikacja między wątkami

### Mechanizm zarządzania pamięcią

### Integracja z kodem napisanym w języku C

### Przykłady użycia wybranych pakietów

- a. Przetwarzanie tekstu
- b. Zarządzanie buforami
- c. Operacja na plikach, parsowanie danych, regex, xml, json, csv
- d. Aspekty szyfrowania danych
- e. Funkcje haszujące
- f. Komunikacja między procesami
- g. Użycie mechanizmu refleksji
- h. Aspekty programowania sieciowego: HTTP, RPC
- i. Implementacja testów automatycznych