

# Langage GO

3 j (21 heures)

Ref : LGGO

## Public

Chefs de projets et développeurs

## Pré-requis

Aucun

## Moyens pédagogiques

Formation réalisée en présentiel ou à distance selon la formule retenue  
Présentation des concepts, discussion technique, démonstrations, exercices simples et TP  
Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique  
Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard  
Support de cours fourni à chaque stagiaire

## Modalités de suivi et d'évaluation

Feuille de présence émargée par demi-journée par les stagiaires et le formateur  
Exercices de mise en pratique ou quiz de connaissances tout au long de la formation permettant de mesurer la progression des stagiaires  
Questionnaire d'évaluation de la satisfaction en fin de stage  
Auto-évaluation des acquis de la formation par les stagiaires  
Attestation de fin de formation

## Objectifs

Développer une application en Go en respectant les idiomes et les nomenclatures  
Maîtriser les concepts de concurrence et de programmation parallèle  
Structurer, tester et déboguer votre application

## Programme détaillé

### INTRODUCTION

---

Historique sur les origines de Go  
Convention de nommage  
Documentation du code source  
Exposition de structures et d'interfaces

## **PREMIERS PAS**

---

- Les variables
- Les types de bases
- Les entrées/sorties
- Les conditionnelles
- Les boucles

## **LES TYPES DE COLLECTIONS**

---

- Les tableaux
- Les slices
- Les maps

## **LES TYPES ET LES STRUCTURES**

---

- Création d'alias de type
- Création de structure de type
- Travailler avec des constantes

## **LES FONCTIONS ET LES METHODES**

---

- Les paramètres de fonction
- Création de méthodes sur une structure
- Les choix entre valeur par pointeur ou par copie

## **LES INTERFACES**

---

- Implémenter une interface existante
- Créer sa propre interface
- Composition d'interface

## **LES TESTS UNITAIRES**

---

- Création d'un jeu de tests unitaires
- Couverture du code et génération de documents
- Création de benchmarks

## **LA CONCURRENCE**

---

- Utilisation de WaitGroupes et Mutexes
- Les fonctions atomiques
- Création de canaux
- Parallélisme vs concurrence
- Design Patterns de Goroutines

