# Foundation Go

TRAINER

**Magesh Kuppan**

- Profile - http://in.linkedin.com/in/tkmagesh/

OVERVIEW

This workshop aims at equipping the participants with the necessary knowledge and skills required to build and manage applications in Go

OBJECTIVES

At the end of this training course, the participants will:

- Become familiar with the language constructs
- Learn the unique features of the language
- Learn to use the standard library to solve common problems

SUGGESTED AUDIENCE

Software developers who want to build best-of-breed applications in Go

DURATION

2 Days

PARTICIPANT PREREQUISITES

- Should be familiar with any of the programming languages

INFRASTRUCTURE REQUIREMENTS

- Go Tools
- Chrome
- Sublime Text / Visual Studio Code / Any other editor
- Any GIT client
- Internet Connection (Mandatory)

CASE STUDY

## Agenda

### Day-1

- Introduction
  - Overview of Go language
  - Comparison with other language
  - Unique features
- Building Blocks
  - Packages
- Functions in Go
  - Go functions
  - Function parameters
  - Returning results

  - Anonymous functions
  - Higher order functions
    - Functions as arguments
    - Functions as return values
    - Function Composition
  - Deferred functions
- Error handling
  - Errors as values
  - Creating Errors
    - errors.New()
    - fmt.Errorf()
  - Nested Errors
- Panic & Recovery
  - Creating Panics
  - Recovering from panics

**Day-2**

- Pointers
- Modules & Packages
  - Creating Modules
  - Packages
  - Nested Packages
  - Module Management
- OO in Go
  - Objects in Go
  - Go methods
  - Struct Composition (aka inheritance)
  - Method Overriding
- Interfaces
  - Defining contracts
  - Interface Composition
- Concurrency
  - Managed Concurrency
  - Goroutines
  - Channels
  - Passing data in and out of channels
  - Streaming Data through Channels
  - Handling race conditions
  - Parallelism in Go

- sync package
- Synchronization constructs in “sync/atomic” packages