**The Go programming language** was created by Google to **do what Google does**: performant web applications at scale.

Open-sourced in 2009 and reaching version one in 2012, the Go programming language is **the best choice for web development** programming today.

Ruby on Rails, Python, Django, Node.js, PHP, and ASP all fall short.

Go is the most **powerful, performant, and scalable** programming language today for creating web applications, web API’s, microservices, and other distributed services.

In this course, you will **gain a solid foundation in web development**. You will learn all of the following **and more:**

**Architecture**

* networking architecture
* the client / server architecture
* the request / response pattern
* the RFC standards defined by the IETF
* the format of requests from clients and responses from servers

**Templates**

* the role that templates play in server-side programming
* how to work with templates from Go’s standard library
* modifying data structures to work well with templates

**Servers**

* the relationship between TCP and HTTP
* how to build a TCP server which responds to HTTP requests
* how to create a TCP server which acts as an in-memory database
* how to create a restful TCP server that handles various routes and methods
* the difference between a web server, a servemux, a multiplexer, and a mux
* how to use a third-party server such as julien schmidt’s router
* the importance of HTTP methods and status codes

**The net/http package**

* streamlining your web development with the net/http package
* the nuances of the net/http package
  + the handler interface
  + http.ListenAndServe
  + creating your own servemux
  + using the default servemux
  + http.Handle & http.Handler
  + http.Handlefunc, func(ResponseWriter, \*Request), & http.HandlerFunc
  + http.ServeContent, http.ServeFile, & http.FileServer
  + http.StripPrefix
  + http.NotFoundHandler

**State & Sessions**

* how to create state: UUID’s, cookies, values in URL’s, security
* how to create sessions: login, permissions, logout
* how to expire a session

**Deployment**

* how to purchase a domain
* how to deploy an application to **Google Cloud**

**Amazon Web Services**

* how to use Amazon Web Services (AWS)
* how to create a virtual linux machine on AWS EC2 (Elastic Cloud Compute)
* how to use secure shell (SSH) to manage a virtual machine
* how to use secure copy (SCP) to transfer files to a virtual machine
* what load balancers are and how to use them on AWS

**MySQL**

* how to use MySQL on AWS
* how to connect a MySQL workbench to AWS

**MongoDB**

* understanding CRUD
* how to use MongoDB & Go

**MVC (Model View Controller) Design Pattern**

* understanding the MVC design pattern
* using the MVC design pattern

**Docker**

* virtual machines vs containers
* understanding the benefits of using Docker
* Docker images, Docker containers, and Docker registries
* implementing Docker and Go
* deploying Docker and Go

**Google Cloud**

* Google Cloud Storage
* Google Cloud no-sql datastore
* Google Cloud memcache
* Google Cloud PAAS App Engine

**Web Dev Toolkit**

* AJAX
* JSON
* json.Marhsal & json.Unmarshal
* json.Encode & json.Decode
* Hash message authentication code (HMAC)
* Base64 encoding
* Web storage
* Context
* TLS & HTTPS
* JSON with Go using Tags

**Building Applications**

* a photo blog
* a twitter clone

By the end of this course, you will have mastered the fundamentals of web development.

My name is Todd McLeod. I am tenured faculty in Computer Information Technology at Fresno City College and adjunct faculty in Computer Science at California State University Fresno. I have taught enough students over 17 years to know that by the end of this course, you will be an outstanding web developer.

You will have the best skills available today.

You will know the best way to do web development today.

You will have the hottest, most demanded, and highest paid skills in the marketplace.

Join me in this outstanding course. Come learn best practices for web development. Sign up for this course now and open doors to a great future.