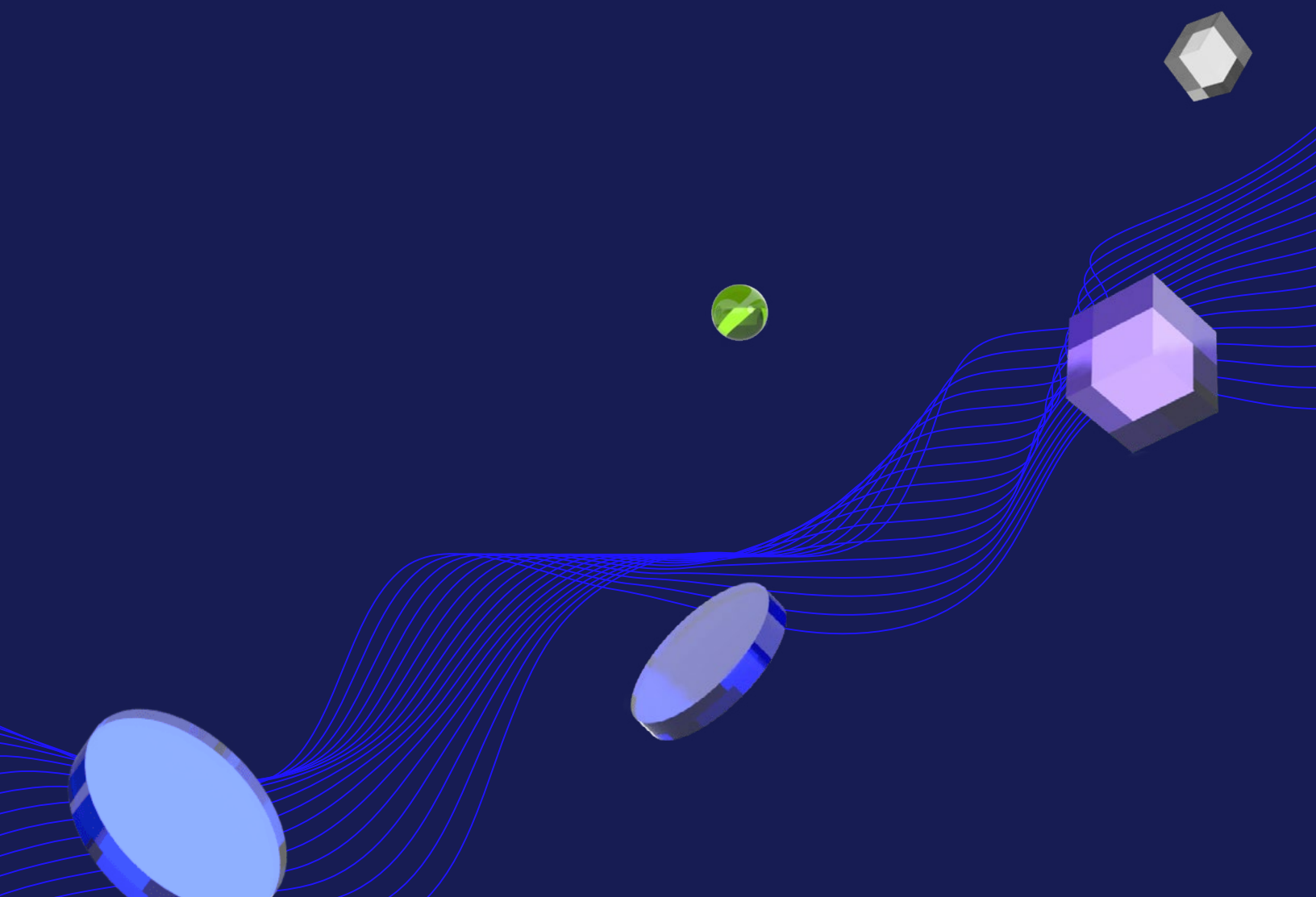
INDIVIDUAL LEARNERS

UDACITY

SCHOOL OF PROGRAMMING & DEVELOPMENT

Golang

Course Syllabus



Overview

Learn the important and foundational skills of the Go programming language. Learners will discover Go's implementation of traditional programming language features such as variables, functions, conditionals, loops, and data structures—as well as features that make Go powerful and performant, such as structs, interfaces, and Goroutines.

This course prepares learners with foundational skills they need for careers involving cloud services, web applications, DevOps tools, and backend development. Upon completion of the course, learners will be able to build a web server through which end-users can interact.

Program information

Estimated Time

1 month

Skill Level

Beginner

Prerequisites

A well-prepared learner has experience with basic computer programming, preferably with a statically-typed language (e.g. Java, C, C++). Learners who are not comfortable with these skills are welcome to take our Intro to programming Nanodegree program to get up to speed.

Required Hardware/Software

Workspaces are provided for students to complete all exercises and the project. For local install if needed (optional):

- Visual Studio Code - the preferred code editor in this course
- Go - a Visual Studio Code extension to help with Go development. Download and install official Go documentation to get started
- Postman app : to test and make API requests

Course Project

CRM Backend

Build the backend for a CRM (Customer Relationship Management) tool used for managing customers. Though learners won't be building the front end user interface for such an app, their role as a back end engineer is an important one: to build the API with which customers will make requests to and get responses from. They will use all of the programming constructs and commands of Golang to implement the required functions to support getting a list of all customers, getting data for a single customer, adding a customer, updating a customer's information, and removing a customer. All of this will be implemented through the Go http package so it is web accessible.

Lesson 1

Introduction to Go

- Explain the positives and shortcomings of Go.
- Develop a business case for why a business should move over to Go.
- Define what the Go language is.

Lesson 2

Go Fundamentals I

- Set up and install Go on a local machine.
- Store values by declaring variables.
- Write code that executes under defined conditions.
- Evaluate values by using comparison operators.
- Write groups of related statements to perform a common, specific task.
- Create an array to store multiple values of the same type.
- Create a flexible, dynamically-sized array.
- Iterate over values across data structures.

Lesson 3

Go Fundamentals II

- Create a collection of unordered key-value pairs.
- Group data together by building a typed collection of fields that reflect traditional classes.
- Create blueprints for method sets.
- Build for concurrency by leveraging Goroutines.

Lesson 4

Building for the Web

- Use HTTP methods for asynchronous requests to a REST API.
- Provide a server implementation using the net/http package.
- Handle HTTP requests sent to defined paths by using the Handler interface.
- Implement an HTTP request router.
- Serve static HTML via a backend server.

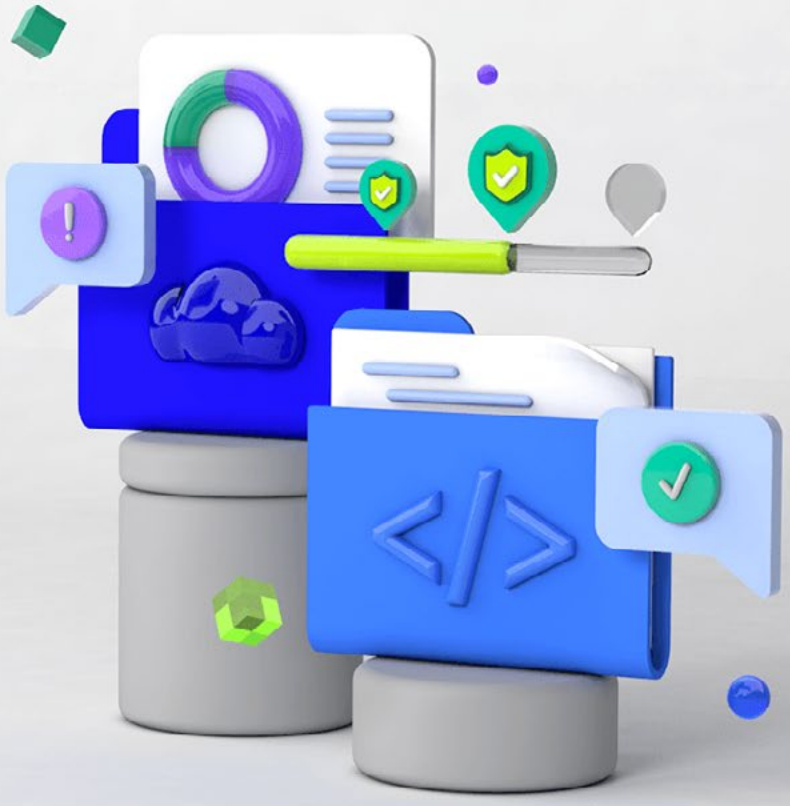
Meet your instructor.

Andrew Wong

Software Engineer

Andrew is a full stack engineer who enjoys making the world a better place through code. He first discovered his passion for teaching as an instructor at App Academy, and continues to enjoy empowering students to advance their education.

Udacity's learning experience



Hands-on Projects

Open-ended, experiential projects are designed to reflect actual workplace challenges. They aren't just multiple choice questions or step-by-step guides, but instead require critical thinking.

Quizzes

Auto-graded quizzes strengthen comprehension. Learners can return to lessons at any time during the course to refresh concepts.

Knowledge

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover how to solve the challenges that you encounter.

Custom Study Plans

Create a personalized study plan that fits your individual needs. Utilize this plan to keep track of movement toward your overall goal.

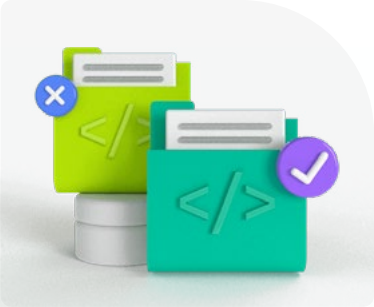
Workspaces

See your code in action. Check the output and quality of your code by running it on interactive workspaces that are integrated into the platform.

Progress Tracker

Take advantage of milestone reminders to stay on schedule and complete your program.

Our proven approach for building job-ready digital skills.



Experienced Project Reviewers

Verify skills mastery.

- Personalized project feedback and critique includes line-by-line code review from skilled practitioners with an average turnaround time of 1.1 hours.
- Project review cycle creates a feedback loop with multiple opportunities for improvement—until the concept is mastered.
- Project reviewers leverage industry best practices and provide pro tips.

Technical Mentor Support

24/7 support unblocks learning.

- Learning accelerates as skilled mentors identify areas of achievement and potential for growth.
- Unlimited access to mentors means help arrives when it's needed most.
- 2 hr or less average question response time assures that skills development stays on track.

Mentor Network

Highly vetted for effectiveness.

- Mentors must complete a 5-step hiring process to join Udacity's selective network.
- After passing an objective and situational assessment, mentors must demonstrate communication and behavioral fit for a mentorship role.
- Mentors work across more than 30 different industries and often complete a Nanodegree program themselves.

UDACITY

Learn more at

www.udacity.com/online-learning-for-individuals →

07.06.22 | V1.0