

Go-разработчик с нуля

8 месяцев
10 проект

Чему вы научитесь на курсе Go-разработчик с нуля:

- изучите синтаксис и базовые возможности языка научитесь работать с IDE и GitHub'ом
- сможете создавать бэкенд для веб-приложений освоите работу с протоколом HTTP
- изучите архитектуру REST API и научитесь создавать простые API
- научитесь обслуживать базы данных и подключать их к приложениям
- освоите SQL-запросы для работы с базами данных
- изучите основы работы с многопоточностью и горутинами — отличительной особенностью языка Go
- сможете разворачивать приложения с использованием Docker и настраивать простой CI/CD изучите базовую автоматизацию работы с помощью командной строки Linux

Вебинары

Один раз в спринт наставники проводят вебинары. Они дополняют теоретический материал и помогают разобраться с заданиями. Кроме того, это отличная возможность провести работу над ошибками и узнать ответы на волнующие вас вопросы по курсу, Go и программированию в целом.

Содержание модуля

00. Бесплатная часть	Введение Основы Go	2 часа
01. Основы GO	Спринт 1: Введение в курс, Переменные, Типы данных, Вывод на экран Спринт 2: Условный оператор, Циклы, Массивы и слайсы, Коммуникация как инструмент разработчика, Мапы Спринт 3: Функции, Обработка ошибок, Настройка окружения: IDE, Настройка окружения: Git Спринт 4: Работа со временем, Пакеты и модули, Требования к коду, Тестирование, Основы отладки программ Спринт 5: Структуры, методы и эмбеддинг: первый взгляд, Введение в структуры Go, Методы в Go, Эмбеддинг в Go, Указатели, Интерфейсы	10 недель 128 часов
02. HTTP в Go/REST API	Спринт 6: Работа с HTTP в Go, Работа с разными форматами данных, Работа с файлами, Создание REST API, Спринт 7: Хеширование и шифрование, Тестирование	4 недели 54 часа
03. SQL и базы данных	Спринт 8: Введение в базы данных, Работа с базами данных в Go	2 недели 46 часов
04. Многопоточность в Go	Спринт 9: Многопоточность	2 недели 36 часов
05. Базовый Linux	Спринт 10: Основы Linux, Работа в Linux: каталоги и файлы, Управление правами доступа, Промежуточная аттестация. Финальный проект	2 недели 23 часа
06. Основы CI/CD и работы с Docker	Спринт 11: Docker Спринт 12: CI / CD и deploy приложения	4 недели 30 часов
07. Выпускной проект	Работа над проектом	5 недель 60 часов

[2 часа]

Введение

-
- | | | |
|--|---|---|
| 1. Что такое бэкенд- и фронтенд-разработка | 4. Как проходит обучение в Практикуме | 7. Реально ли найти работу после курса |
| 2. Чем занимаются Go разработчики | 5. Что ждёт в платной части программы | 8. Какие навыки нужны Go-разработчику для успешного трудоустройства |
| 3. Почему Go — это перспективно | 6. Кто будет сопровождать вас во время обучения | 9. Что нужно, чтобы начать |
-

Основы Go

-
- | | | |
|------------------------------------|---------------------------------------|---|
| 1. Напишите первую программу на Go | 2. Научитесь выводить данные на экран | 3. Узнаете что такое переменные и типы данных |
|------------------------------------|---------------------------------------|---|
-

ОСНОВЫ GO

01

[10 недель
128 часов],
[3 проекта]

Модуль посвящён изучению основных положений и концепций программирования на примере языка Go. Рассматриваются как общие для языков бэкенда постулаты и принципы создания кода, так и специфические особенности Go. На этом этапе студенты приобретают навыки создания минимально функциональных программ.

Спринт 1:

-
- | | | |
|--------------------|----------------|-------------------|
| 1. Введение в курс | 3. Типы данных | 4. Вывод на экран |
| 2. Переменные | | |

Спринт 2:

1. Условный оператор

2. Циклы

3. Массивы и слайсы

4. Коммуникация как
инструмент разработчика

5. Мапы

Спринт 3:

1. Функции

2. Обработка ошибок

3. Настройка окружения: IDE

4. Настройка окружения: Git

Спринт 4:

1. Работа со временем

2. Пакеты и модули

3. Требования к коду

4. Тестирование

5. Основы отладки программ

Спринт 4:

1. Структуры, методы
и эмбединг: первый взгляд

2. Введение в структуры Go

3. Методы в Go

4. Эмбединг в Go

5. Указатели

6. Интерфейсы

Каникулы

[1 неделя]

Время отдохнуть
или доработать проекты

HTTP в Go/REST API

02

[4 недели
54 часа],
[2 проекта]

В модуле сделан акцент на веб-разработке как на одном из ключевых компонентов создания современных веб-приложений. Студенты изучают основные принципы сетевого взаимодействия программ и учатся их настраивать

Спринт 6:

1. Работа с HTTP в Go

3. Работа с файлами

4. Создание REST API

2. Работа с разными форматами данных

Спринт 7:

1. Хеширование и шифрование

2. Тестирование

SQL и базы данных

03

[2 недели 46 часов],
[1 проект]

Модуль погружает студентов в важный раздел бэкенд-разработки, связанный с логикой создания хранилищ данных и организацией их содержания. Также студенты учатся подключать базы данных к уже готовым сервисам.

Спринт 8:

1. Введение в базы данных

2. Работа с базами данных в Go

Многопоточность в Go

04

[2 недели 36 часов],
[1 проект]

Модуль посвящён многопоточности в Go как одной из уникальных особенностей языка. Студенты учатся оптимизировать нагрузку на приложение при выполнении большого количества операций.

Спринт 9:

1. Многопоточность

Каникулы

[1 неделя]

Время отдохнуть

или доработать проекты

[2 недели 23 часа],
[1 проект]

В модуле рассматриваются основы работы с командной строкой и взаимодействие с программами на уровне операционной системы. Студенты учатся оптимизировать скорость самого процесса разработки и отладки приложений.

Спринт 10:

1. Основы Linux

2. Работа в Linux: каталоги
и файлы

3. Управление правами
доступа

4. Промежуточная аттестация.
Финальный проект

Основы CI/CD и работы с Docker

[4 недели 30 часов],
[2 проекта]

В этом модуле студенты отрабатывают навыки упаковки приложения, чтобы подготовить его к передаче пользователям

Спринт 11:

1. Docker

Спринт 12:

1. CI / CD и deploy приложения

Выпускной проект

[5 недель
60 часов],
[1 проект]

Бэкенд для планировщика задач
В итоговом задании вам нужно написать на Go веб-сервер, который реализует функциональность простейшего планировщика задач. Это будет аналог TODO-листа. Это задание на проверку и закрепление навыков по написанию веб-сервера, работе с REST API и базами данных.

