

Flask

Название курса: Flask

Количество уроков: 10

Продолжительность в часах: 10 часов

Продолжительность в днях: 10 дней

Описание курса

Курс Flask состоит из 10 логически взаимосвязанных уроков, на которых учащиеся применяют на практике теоретические знания, используя Python и Flask для построения RESTful API.

На этом курсе вы познакомитесь с Flask, получите необходимые теоретические и практические знания для построения собственного приложения или для поддержки уже существующего проекта. На протяжении всего курса мы будем строить веб-приложение, которое в дальнейшем будет вашим базисом, на который можно ссылаться. Кроме того, мы обсудим все подводные камни, на которые может «наступить» новичок и также поговорим о лучших практиках, которые используются в разработке веб-приложений с помощью Flask Fr. По окончании курса, вы сможете реализовать любую вашу идею для проекта, упаковать его в контейнер и развернуть на любом хостинге, также узнаете, что такое ORM и для чего она нужна, как можно ускорить ваше приложение и как быть уверенным, что новые изменения, добавленные в код, не сломали всю систему.

Предварительные требования

- Этот курс не является курсом для новичков. Мы предполагаем, что у вас есть базовые навыки программирования на Python. Мы собираемся использовать такие вещи, как классы, декораторы и функции и мы не будем описывать, что такое класс или что такое функция. Таким образом, вы должны уметь уверенно использовать эти знания на практике.
- Для эффективного прохождения на курсе всего этапа разработки желательно использовать POSIX ОС (Linux, MacOS).

Целевая аудитория

Курс ориентирован на разработчиков, у которых до 1 года опыта разработки или без опыта но, у которого есть уверенные знания Python или другого ЯП.

По завершении курса Вы сможете

- Разрабатывать REST API на Python.
- Настраивать среду разработки.
- Best practices для работы с базами данных.
- Тестировать Python Web Applications.
- Упаковывать веб-приложение в Docker.
- Работать с потоками, и получите базовое представление об асинхронности.

Список уроков курса:

Урок 1: Знакомство с Flask

Первым делом, на этом уроке настроим среду для разработки, установим все зависимости. Также, мы познакомимся с Flask, узнаем, как он работает и из каких компонентов состоит: представления, модели и контроллеры. Построим первое веб-приложение с использованием Jinja2 Templates: работа с циклами и условиями, наследование форм.

1. Настройка среды разработки
2. Знакомство с Flask
3. Построение первого веб-сайта с помощью Flask Framework и Jinja2 Templates

Урок 2: Создание RESTful API

В крупных проектах разделяют frontend и backend части, чтобы они могли развиваться независимо друг от друга. Поэтому мы познакомимся с архитектурным стилем построения веб-сервисов REST. Начнем работу над полноценным веб-приложением для контроля расходов, познакомимся с SQLAlchemy и свяжем приложение с SQLite3. Узнаем, что такое SwaggerUI и как можно автоматически генерировать для него документацию. Также рассмотрим альтернативы REST: gRPC и GraphQL.

1. Что такое REST?
2. Проектирование RESTful API: CRUD запросы
3. Моделирование схемы БД
4. SwaggerUI
5. Альтернативы REST

Урок 3: Валидация пользовательских данных

Обсудим, зачем нужна валидация пользовательских данных на стороне сервера, и что такое сериализация и десериализация данных. Улучшим структуру приложения.

1. Использование Marshmallow
2. Улучшенная структура приложения

Урок 4: Использование SQLAlchemy.

Целью данного урока будет использование SQLAlchemy, рассмотрение более сложных запросов. Обсудим, зачем нужны ORM и какие существуют подводные камни при работе с ORM.

1. Зачем нужны ORM
2. Продвинутые возможности SQLAlchemy.
3. Подводные камни ORM: проблема n+1 запроса

Урок 5: Alembic миграции

На этом уроке поговорим о том, что такое миграции и для чего они используются. Также, рассмотрим инструменты для миграций во Flask.

1. Alembic
2. Flask-Migrate

Урок 6: Аутентификация

В этом уроке поговорим о том, что такое аутентификация и чем она отличается от авторизации. Обсудим способы аутентификации REST приложения. Познакомимся с JWT.

1. Авторизация и аутентификация
2. Способы аутентификации REST приложений
3. Аутентификация на основе JWT

Урок 7: Pytest, Travis CI

В этом уроке поговорим о том, как правильно тестировать Flask приложение, какие бывают тесты. Также, научимся использовать Mock и настроим CI процесс для нашего приложения.

1. pytest и unittest.mock
2. TravisCI

Урок 8: Dockerize flask app

В этом уроке поговорим о том, что такое Docker и как доставлять приложения клиенту. Научимся писать Dockerfile и развернем наше приложение на Heroku.

1. Docker и Dockerfile
2. Gunicorn
3. Heroku деплой

Урок 9: Многопоточное программирование

В этом уроке обсудим, что такое потоки, и как они могут ускорить ваше приложение. Поговорим, что такое GIL и какие бывают примитивы синхронизации в Python.

1. Потоки
2. GIL и примитивы синхронизации
3. ThreadPoolExecutor in Flask

Урок 10: Async and await, концепты асинхронности

В этом уроке поговорим о том, что такое асинхронное программирование и какие у него есть преимущества и недостатки. Рассмотрим, какие бывают асинхронные фреймворки на Python. Обсудим разницу между синхронным кодом и асинхронным и обсудим области применения каждого.

1. Async and await
2. Преимущества и недостатки асинхронного кода
3. Asyncio-based фреймворки