

ВВЕДЕНИЕ В РАБОТУ С API

ВВЕДЕНИЕ

Задачи урока

- 01** Узнать основные принципы FastAPI и его преимущества перед другими фреймворками
- 02** Продемонстрировать создание простого веб-приложения с использованием FastAPI
- 03** Объяснить архитектурный стиль RESTful API и его ключевые принципы
- 04** Объяснить принципы работы асинхронного выполнения запросов в FastAPI

FastAPI

это современный фреймворк для создания веб-приложений на языке программирования Python. Он основан на стандарте ASGI и обладает высокой производительностью и простотой использования. FastAPI предоставляет декларативный подход к определению обработчиков API с использованием аннотаций типов Python. Это позволяет автоматически генерировать интерактивную документацию API и обеспечивает валидацию данных входных запросов и выходных ответов. FastAPI также поддерживает асинхронное выполнение запросов, что дает возможность эффективно обрабатывать большое количество запросов одновременно

FastAPI используется для создания веб-приложений и микросервисов. Он может использоваться во многих различных сферах, включая разработку RESTful API, веб-сервисы, системы управления контентом и т. д. FastAPI является отличным выбором для проектов, где требуется высокая производительность и эффективная обработка параллельных запросов. Он также подходит для создания масштабируемых систем и приложений, которым требуется взаимодействие с базами данных и другими сторонними сервисами

Существует множество других фреймворков для разработки веб-приложений на языке Python

Некоторые из популярных фреймворков **включают**

Django

мощный фреймворк с полным набором инструментов для создания веб-приложений любой сложности

Flask

легкий и гибкий фреймворк, позволяющий быстро создавать простые и средние веб-приложения

Pyramid

фреймворк, ориентированный на гибкость и расширяемость, подходящий для разработки разнообразных веб-приложений

CherryPy

простой и эффективный фреймворк, написанный на Python. Он предлагает легкую и быструю разработку веб-приложений

Tornado

фреймворк, специализирующийся на асинхронном программировании, позволяющий обрабатывать много подключений одновременно

Bottle

компактный и простой фреймворк, специально разработанный для маленьких веб-приложений и API

RESTful API Representational State Transfer

это архитектурный стиль для проектирования и разработки веб-сервисов и API, которые основаны на использовании протокола HTTP. RESTful API является популярным и широко используемым подходом, который позволяет взаимодействовать с серверными ресурсами, представленными в виде URL-адресов

Основные принципы **RESTful API** включают

✓ Клиент-серверная архитектура

RESTful API является разделенным между клиентской и серверной частями, что позволяет им независимо развиваться и масштабироваться

✓ Без состояния *Stateless*

RESTful API является разделенным между клиентской и серверной частями, что позволяет им независимо развиваться и масштабироваться

✓ Кэширование *Caching*

RESTful API может использовать механизмы кэширования HTTP для улучшения производительности и сокращения нагрузки на сервер

✓ Единообразный интерфейс *Uniform Interface*

RESTful API использует однородный набор методов **HTTP (GET, POST, PUT, DELETE)** для работы с ресурсами, а также принципы именования ресурсов и коды состояния HTTP

✓ Слоистая система *Layered System*

RESTful API может быть построен в виде слоистой архитектуры, где каждый слой отвечает за определенные аспекты обработки запросов

RESTful API предоставляет операции для работы с данными, такие как получение (GET), создание (POST), обновление (PUT) и удаление (DELETE) ресурсов. Каждый ресурс представлен уникальным URL-адресом, и клиент может использовать HTTP-методы и параметры для выполнения нужных операций

ИТОГИ

- ✓ Вы узнали, что преимущества FastAPI, способны создавать простые веб-приложения и RESTful API с использованием этого фреймворка
- ✓ Вы познакомились с принципами RESTful API и сравнили FastAPI с другими популярными фреймворками для создания веб-приложений на Python