

Курс Full Stack разработчика. Часть 1

[Full Stack Developer Course](#) (Python, [Vue.js](#), [FastAPI](#))

Этап 1: Основы Python и асинхронного программирования

1.1 Основы Python

- ☐ Типы данных и структуры данных:
 - Списки, кортежи, множества, словари
 - Введение в строки, регулярные выражения, обработку ошибок
 - ООП (Классы, наследование, инкапсуляция, полиморфизм)
- ☐ Работа с файлами и библиотеками:
 - Чтение и запись в файлы (JSON, CSV)
 - Использование стандартных библиотек: `os`, `sys`, `datetime`, `logging`

Практика:

- ☐ Написание программы для обработки данных (например, анализ данных из CSV)
- ☐ Создание калькулятора с использованием ООП

1.2 Асинхронное программирование с Python

- ☐ Основы `asyncio`:
 - Понимание асинхронности в Python: `async def`, `await`
 - Основы работы с циклами событий
- ☐ Асинхронные библиотеки:
 - Работа с `aiohttp` для выполнения асинхронных HTTP-запросов
 - Асинхронная работа с файловой системой с использованием `aiofiles`

Практика:

- ☐ Написание асинхронного веб-скрапера
- ☐ Реализация асинхронной отправки сообщений в Telegram

Этап 2: Создание API с FastAPI

2.1 Основы FastAPI

- ☐ Маршруты и обработка запросов:
 - Создание REST API с FastAPI
 - Работа с HTTP методами (`GET`, `POST`, `PUT`, `DELETE`)
- ☐ Модели данных с Pydantic:
 - Валидация данных с помощью Pydantic
 - Создание моделей для ввода и вывода данных

Практика:

- ☐ Создание простого API для управления пользователями (регистрация, вывод профиля)

2.2 Работа с базой данных

- ☐ Интеграция с SQLAlchemy и PostgreSQL:
 - Создание моделей и миграции базы данных
 - Асинхронная работа с базой данных через `Databases` или `Tortoise-ORM`
- ☐ Аутентификация и авторизация:
 - Реализация JWT для аутентификации пользователей
 - Реализация разных уровней доступа (администраторы, пользователи)

Практика:

- ☐ Разработка CRUD-операций для пользователя
 - ☐ Реализация системы аутентификации и авторизации с JWT
-