

Python-разработчик буткемп

22 недели

продолжительность курса

10 проектов

в портфолио

01	Основы Python: бесплатный курс	10 часов
02	Углублённый Python	60 часов, 2 проекта
03	Бэкенд на Django	120 часов, 4 проекта
04	API: интерфейс взаимодействия программ	90 часов, 3 проекта
05	Управление проектом на удалённом сервере	60 часов
06	Дипломный проект Foodgram	90 часов
07	Асинхронность и Flask	36 часов, 1 проект
08	Алгоритмы и структуры данных (опциональный модуль)	40 часов

[10 часов]

Во вводном курсе вы почувствуете себя в роли бэкенд-разработчика на Python. Это поможет вам понять, хотите ли вы развиваться в этом направлении.

Вы узнаете, что такое бэкенд и какие задачи он решает, разберётесь, как программы взаимодействуют с пользователями и с другими программами.

В каждом уроке после небольшой теоретической части вы будете писать код на Python — с первого же урока. Работать вы будете в онлайн-тренажёре — нашей интерактивной среде.

Этот курс докажет главное: вы самостоятельно можете написать программный код, который будет работать и делать то, что задумано.

Содержание

Темы	1. Знакомство с Python 2. Циклы и ветвления 3. Функции 4. Словари и множества
------	---

[60 часов]

Языки

Python 3.12

Инструменты

Git, GitHub, редактор кода
Visual Studio Code, Flake8,
Pytest, bash, виртуальное
окружение

Знания

Основы работы в Bash,
правила оформления кода,
работа в виртуальном
окружении, основы ООП,
продвинутые возможности
Python.

Вы детально изучите «грамматику» Python, научитесь применять встроенные инструменты и модули этого языка, познакомитесь с типами данных; выясните, как Python работает с оперативной памятью, и узнаете, почему это знание важно на практике; изучите требования к коду и правила подготовки проектов.

В этом же блоке вы познакомитесь с основами объектно-ориентированного программирования, установите и настроите редактор кода, узнаете, как работать с системой версионирования кода. В уроках вас ждут практические задания, а в конце каждого спринта — самостоятельный проект. Финальный проект блока «Углублённый Python» — самостоятельное создание игры «Изгиб Питона». Эта работа будет проверена ревьюерами.

Содержание спринтов

Спринт 1

- Числовые типы
- Булева алгебра. Условные операторы
- Последовательности
- Python "под капотом"
- Множества
- Словари
- Типы данных
- Создание функций
- Стандартная библиотека

Проект: приложение
«Холодильник»

Система регистрации
и отслеживания продуктов
в холодильнике.

Спринт 2

- Инструкции по установке ПО
- Настройка окружения: Python, IDE, venv
- Настройка окружения: Git, pytest
- Требования к коду
- Правила оформления кода
- Аннотации типов
- Объектно-ориентированное программирование (ООП) в Python
- Самое необходимое об ООП

Проект: игра «Изгиб Питона»

Реализация классической игры
«Змейка» на Python.

[120 часов]

- Языки
- Python 3.12, HTML, CSS, SQL
- Инструменты
- Django Web Framework, Bootstrap, Unittest, Pythonanywhere, Pytest, SQLite
- Знания
- Основы HTML и вёрстки для бэкендера, создание основы проекта и добавление новых приложений в Django, применение MVC на практике, использование шаблонизатора Django, работа с Django ORM, тестирование проекта, деплой проекта в облако.

Главная тема курса — работа с веб-фреймворком Django, это программный «конструктор» для создания веб-проектов. Вы создадите полноценное веб-приложение — это будет платформа для блогов. Вы шаг за шагом создадите полноценный портал — с авторизацией, персональными лентами и комментариями. В процессе работы вы научитесь обрабатывать HTTP-запросы, подключите к приложению базу данных, создадите веб-страницы проекта. Любой код нуждается в проверке — вы научитесь писать автоматические тесты для своих программ. К концу курса вы допишете проект — и он станет ещё одной выполненной работой в вашем портфолио.

Содержание спринтов

Спринт 3	- Создание проекта - Пути и view-функции - Вёрстка для бэкендера - HTML и шаблоны Django - Анализ и решение проблем	Проект: Блогикум. Часть 1. Развёртывание и настройка веб-приложения, которое станет платформой для блогов.
Спринт 4	- Введение в базы данных - Отношения между таблицами - Django ORM. Модели - Админ-зона Django - Django ORM. Получение информации из БД - Работа с формами - Представления: расширенные возможности - Пользователи в Django - Декораторы - Мотивация и измеримые цели - Что делать с внутренним критиком	Проект: Блогикум. Часть 2. Настройка панели администратора, работа с публикациями. Проект: Блогикум. Часть 3. Подключение возможности самостоятельной регистрации пользователей, публикации постов через форму, а также их комментирования.
Спринт 5	- Тестирование - Библиотека unittest - Unittest в Django - Библиотека pytest - Pytest для Django - Нетворкинг: персональная сеть взаимопомощи	Проект: Vice Versa Тестирование проектов YaNote и YaNews на unittest и pytest.

API: интерфейс взаимодействия программ

04

[90 часов]

- Языки
- Python 3.12, RegExp, JSON
- Инструменты
- Django REST Framework, React, Telegram API, JWT, Postman, requests
- Знания
- Настройка взаимодействия Python-приложения с внешними API-сервисами, создание собственного API-сервиса на базе проекта Django, создание Telegram-ботов, подключение SPA к бэкенду на Django через API, основы групповой разработки ПО.

В этой теме вы разберётесь, как веб-проекты взаимодействуют друг с другом, запрашивая и передавая информацию: например, каким образом ваша программа на Python может общаться с серверами Telegram для управления ботом, а тот, в свою очередь может получать информацию от других сервисов в интернете.

Вы узнаете, что такое API, создадите свой собственный REST API сервис, научитесь настраивать взаимодействие между фронтендом и бэкендом, а также напишете бота, который взаимодействует с популярными веб-сервисами. Кроме того, вы узнаете, как может быть организована авторизация в разных проектах.

Содержание спринтов

Спринт 6	• Что такое API • Работа с внешними API • Анализ и решение проблем • Тайм-менеджмент	Проект: Бот-ассистент Создание Telegram-бота, оповещающего о статусе домашнего задания.
Спринт 7	• Django REST Framework • REST API: проектирование • Сериализаторы • View-функции API • View-классы API • Вьюсеты и роутеры • Регулярные выражения • Как работать в команде • Подходы в управлении командой	Проект: YouTube. Часть 1. CRUD API для сервиса YouTube.
Спринт 8	• Проверка прав: Permissions • Throttling: ограничение количества запросов • Фильтрация, сортировка и поиск • CORS: политика единого источника • Документирование API • Взаимодействие фронтенда и бэкенда через API • Кросс-ревью	Проект: YouTube. Часть 2. Полноценный API для сервиса YouTube.

Содержание спринтов

Спринт 9	• Групповой проект	Проект: Создание и документирование REST API
----------	--------------------	---

Управление проектом на удалённом сервере

05

[60 часов]

Языки
Python 3.12, YAML

Инструменты
Docker, Nginx, PostgreSQL, Gunicorn, GitHub Actions

Знания
Создание образов и запуск контейнеров Docker, создание и деплой мультиконтейнерных приложений, основы DevOps, CI/CD..

Работа с облачными сервисами, настройка серверов, тестирование и обновление кода — всё это рутинные операции, с которыми сталкивается любой программист. Вы самостоятельно настроите сервер в облаке, узнаете, как автоматизировать процессы тестирования и обновления кода, развернёте свой проект в docker-контейнерах на сервере.

Содержание спринтов

Спринт 10	• Работа на сервере, настройка окружения • Доменное имя и SSL-сертификат для сайта • Мониторинг работы веб-приложения • Модели жизненного цикла ПО	Проект: Kittygram. Часть 1. Классический деплой проекта на удалённый сервер.
Спринт 11	• Запуск приложений в docker-контейнерах • Упаковка проекта в Docker-образ • Архитектура приложения в Docker • Docker-compose • Автоматизация деплоя: CI/CD • GitHub Actions • Agile: принципы, ценности, основные методологии	Проект: Kittygram. Часть 2. Деплой проекта в контейнерах, автоматизация деплоя с применением CI/CD.

Неделя каникул



[90 часов]	Вы самостоятельно напишете веб-приложение «Foodgram»: сайт, на котором пользователи будут публиковать собственные рецепты, добавлять чужие рецепты в избранное и подписываться на публикации других авторов. Сервис «Список покупок» позволит пользователям создавать список продуктов, которые нужно купить для приготовления выбранных блюд.
------------	--

Асинхронность и Flask

07

[36 часов] Языки Python 3.12 Инструменты GIL, Flask, SQLAlchemy, ORM, SQLite, Dropbox, Jinja2 Знания С помощью асинхронного подхода создадите сервис, который будет сокращать ссылки.	Понимание концепций конкурентности, параллельности и асинхронности поможет вам создавать программы, которые могут эффективно выполнять несколько задач, экономя время и ресурсы.
--	--

Содержание спринтов

Спринт 1	- Конкурентность, параллельность и асинхронность - Основы Flask - API на Flask - Асинхронный Flask
----------	--

[40 часов],

[Оptionальный

модуль]

Языки

Python 3.12

Инструменты

Яндекс Контест

Знания

Понимание применения алгоритмов при решении любых задач, поиск и реализация разных вариантов решения одних и тех же задач, оценка вариантов решения, понимание основных структур данных и применение их для решения конкретных задач.

Любую задачу можно решить медленно и неэффективно, а можно — быстро и экономно. Неэффективное программное решение может впустую занять все вычислительные ресурсы сервера и замедлить его работу, а то и вовсе сломать его. Изучение алгоритмов поможет избежать подобных ошибок и ускорить работу программ. Вы научитесь проектировать решения, которые позволят вашим проектам работать эффективнее.

Содержание спринтов

Спринт 1	• Введение в алгоритмы • Сложность алгоритмов • Структуры данных • Методы решения задач	Финальное задание спринта: Задача «Служба доставки»
Спринт 2	• Рекурсия • Сортировки • Графы и деревья • Жадные алгоритмы • Динамическое программирование	Финальное задание спринта: Шифрованные инструкции