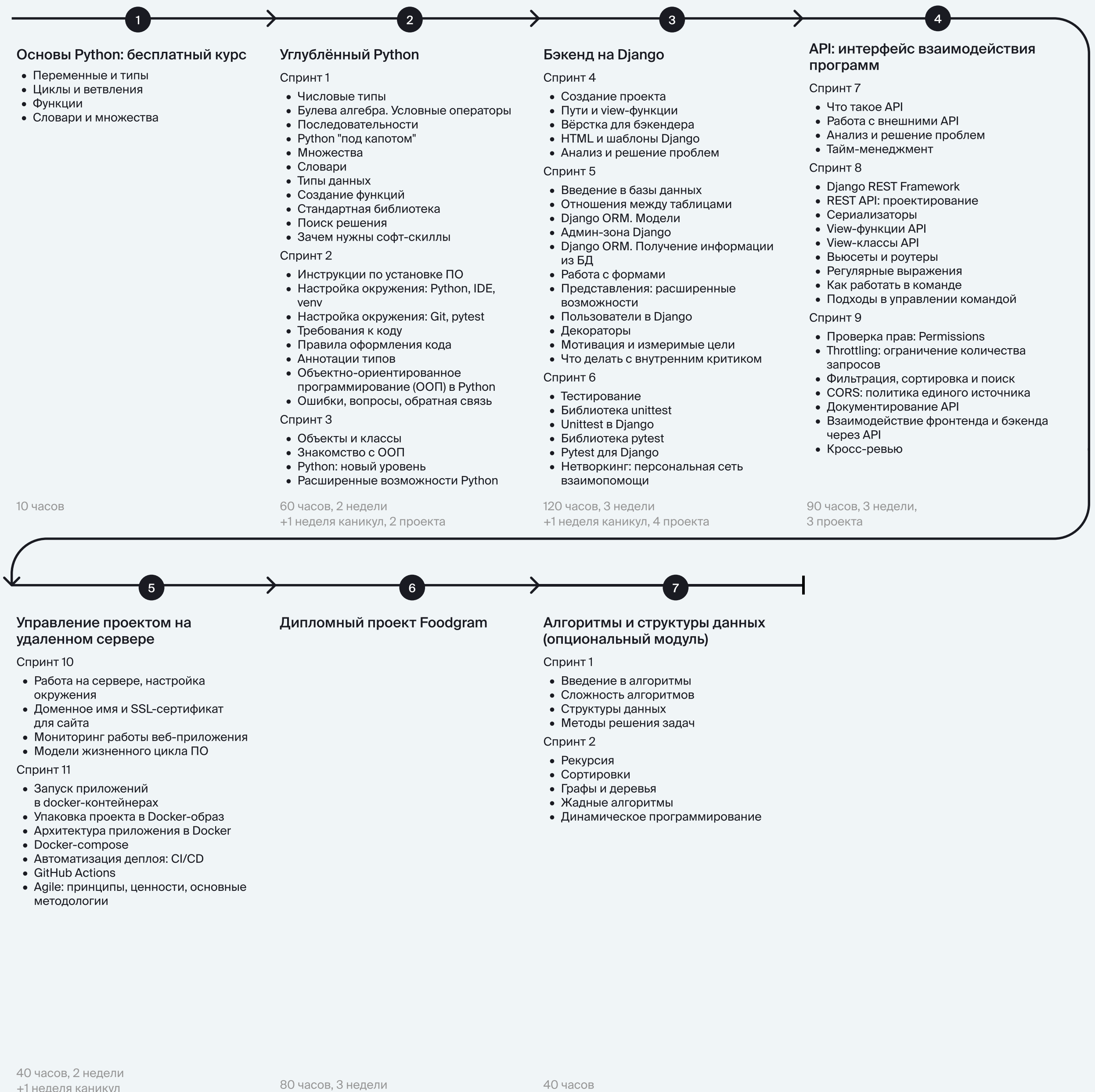


Python-разработчик буткемп



1. Основы Python: бесплатный курс

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.

Темы курса

- Знакомство с Python
- Циклы и ветвления
- Функции
- Словари и множества

2. Углублённый Python

60 часов, 2 недели
+ 1 неделя каникул,
2 проекта

Языки: Python 3.9

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.

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

Проект: игра «Изгиб Питона»

Реализация классической игры «Змейка» на Python.

Спринт 3

- Объектно-ориентированное программирование (ООП) в Python
- Самое необходимое об ООП

3. Бэкенд на Django

Языки: Python 3.9, 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.

120 часов, 3 недели
+ 1 неделя каникул,
4 проекта

Спринт 4

- 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 5

- 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 6

- 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.

4. API: интерфейс взаимодействия программ

90 часов, 3 недели,
3 проекта

Языки: Python 3.9, 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.

Спринт 7

- 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 8

- 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 9

- 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. Работа в команде: Создание и документирование REST API.

5. Управление проектом на удаленном сервере

40 часов, 2 недели
+ 1 неделя каникул,
2 проекта

Языки: Python 3.9, 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.

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.

Каникулы —————> 1 неделя

6. Дипломный проект Foodgram

70 часов, 3 недели,
1 проект

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.

7. Алгоритмы и структуры данных (опциональный модуль)

40 часов

Языки: Python 3.9

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.

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