

Python-разработчик расширенный

14 месяцев

продолжительность курса

19 проектов

в портфолио



01	Основы Python: бесплатный курс	12 часов
02	Углублённый Python	60 часов, 2 проекта
03	Алгоритмы и структуры данных	40 часов
04	Бэкенд на Django	120 часов, 4 проекта
05	API: интерфейс взаимодействия программ	90 часов, 3 проекта
06	Управление проектом на удалённом сервере	40 часов, 2 проекта
07	Самостоятельный проект Foodgram	70 часов, 1 проект
08	Асинхронность и Flask	25 часов, 1 проект
09	Парсинг	35 часов, 2 проекта
10	FastAPI	40 часов, 2 проекта
11	Python для бизнеса	20 часов, 1 проект
12	Проектный месяц: командная работа	50 часов, 1 проект

[12 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

1. Знакомство с Python

2. Циклы и ветвления

3. Функции

4. Словари и множества

Углублённый Python

02

[60 часов]

Языки

Python 3.12

Инструменты

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

Знания

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

Проект: игра «Изгиб Питона»
Реализация классической игры
«Змейка» на Python.

Спринт 3

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

Алгоритмы и структуры данных

03

[40 часов]

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 вычислительного устройства и замедлить его работу, а то и вовсе сломать его. Изучение алгоритмов поможет избежать подобных ошибок и ускорить работу программ.

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

Неделя каникул



Бэкенд на Django

04

[120 часов]

Языки

Python 3.12, HTML, CSS

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 1

- Создание проекта
- Пути и view-функции
- Вёрстка для бэкендера
- HTML и шаблоны Django

Проект: Блогикум. Часть 1.
Развёртывание и настройка веб-приложения, которое станет платформой для блогов.

Спринт 2

- Введение в базы данных
- Отношения между таблицами

Содержание спринтов

Спринт 3

- Django ORM. Модели
- Админ-зона Django
- Django ORM. Получение информации из БД

Проект: Блогикум. Часть 2.
Настройка панели администратора и работа с публикациями

Спринт 4

- Работа с формами
 - Представления: расширенные возможности
-

Спринт 5

- Пользователи в Django

Проект: Блогикум. Часть 3.
Подключение возможности самостоятельной регистрации пользователей, публикации постов через форму, а также их комментирование.

Спринт 6

- Тестирование
- Библиотека Unittest
- Unittest в Django
- Библиотека Pytest
- Pytest для Django

Проект: Vice Versa
Тестирование проектов YaNote и YaNews на unittest и pytest

Неделя каникул



API: интерфейс взаимодействия программ

05

[90 часов]

Языки

Python 3.12, RegExp

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 1

- Что такое API
- JSON: формат передачи данных
- API First. Архитектура REST
- Правила именования ресурсов
- Исследование запросов
- Механизмы авторизации, протокол OAuth 2.0
- Client API в Telegram
- Bot API в Telegram

Проект: Бот-ассистент

Создание Telegram-бота, оповещающего о статусе домашнего задания

Спринт 2

- Django REST Framework
- REST API: проектирование
- Сериализаторы
- View-функции API
- View-классы API
- Вьюсеты и роутеры
- Регулярные выражения

Проект: YouTube. Часть 1.

CRUD API для сервиса YouTube

Спринт 3

- Проверка прав: Permissions
- Throttling: ограничение количества запросов
- Фильтрация, сортировка и поиск
- CORS: политика единого источника
- Документирование API
- Взаимодействие фронтенда и бэкенда через API

Проект: YouTube. Часть 2.

Полноценный API для сервиса YouTube. Создание REST API

Спринт 4

- Командная работа над проектом

Итоговый проект модуля:

YaMDB

Создание REST API для сервиса YaMDB — базы отзывов о фильмах, книгах и музыке. Проект выполняется в команде с другими студентами

Неделя каникул



Управление проектом на удалённом сервере

06

[40 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 1

- Работа на сервере, настройка окружения
- Доменное имя и SSL-сертификат для сайта
- Мониторинг работы веб-приложения

Проект: Kittygram. Часть 1.

Классический деплой проекта на удалённый сервер.

Спринт 2

- Запуск приложений в docker-контейнерах
- Упаковка проекта в Docker-образ
- Архитектура приложения в Docker
- Docker-compose
- Автоматизация деплоя: CI/CD
- GitHub Actions

Проект: Kittygram. Часть 2.

Деплой проекта в контейнерах, автоматизация деплоя с применением CI/CD.

Самостоятельный проект Foodgram 07

[70 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.

Неделя каникул



[25 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

Парсинг

09

[35 часов]

Языки

Python 3.12

Инструменты

Requests, requests-cache, requests-html, bs4, re, tqdm, argparse, prettytable, csv, SQLAlchemy, Scrapy, Selenium, threading, multiprocessing, AsyncIO

Знания

Парсинг информации с помощью библиотеки Beautiful Soup, парсинг страницы с необходимостью авторизации и динамическим содержимым, асинхронный код и использование Scrapy для создания асинхронных парсеров. Использование Scrapy для создания асинхронных парсеров.

Вы познакомитесь с парсингом — искусством собирать информацию с сайтов, структурировать её и выводить в удобном формате. Узнаете, что такое «инструменты разработчика в браузере», научитесь исследовать HTML-разметку сайта и поработаете регулярными выражениями. Первое самостоятельное задание курса по парсингу — собрать данные с сайта, структурировать их и сохранить в удобном для обработки формате. Ещё одна тема курса — фреймворк Scrapy, инструмент для создания асинхронных парсеров.

Содержание спринтов

Спринт 1

- Парсинг: начало
- Работа с браузером
- Основные библиотеки для парсинга
- Настройка парсера
- Вывод и хранение результатов парсинга
- Рефакторинг парсера

Проект: Парсинг PEP

Сбор информации о статусах и типах документов в PEP (Python Enhancement Proposals).

Спринт 2

- Работа с БД
- Парсинг и авторизация
- Requests-html
- Scrapy
- Бонус. Selenium

Проект: Асинхронный парсер PEP

Этот парсер будет собирать и сохранять данные о документах PEP

Fast Api

10

[40 часов]

Языки

Python 3.12

Инструменты

FastAPI, Pydantic, Uvicorn, SQLAlchemy, Alembic, fastapi-users

Знания

Создание приложения на FastAPI.

Фреймворк FastAPI предназначен для быстрой (от англ. fast) разработки API-приложений и основан на использовании стандартной аннотации типов Python. В этом спринте вы реализуете два проекта на FastAPI, они дадут вам представление о том, как работает этот фреймворк.

Содержание спринтов

Спринт 1

- Знакомство с FastAPI
- База данных в FastAPI
- Миграции: библиотека Alembic
- CRUD в FastAPI
- Модель бронирования переговоров
- FastAPI Users

Проект: Приложение QRKot

Проект для Благотворительного фонда поддержки котиков

[20 часов]

Языки

Python 3.12

Инструменты

SQLite, FastAPI, Pydantic, Uvicorn, SQLAlchemy, Alembic, fastapi-users, aiogoogle

Знания

Управление сервисами через Google API, интеграция приложения на FastAPI с сервисами Google Sheets API и Google Drive API, разработка системы управления Google-таблицами.

Вы прокачаете навыки работы с документацией и узнаете, как использовать сервисы платформы Google. Это поможет вам понять общий принцип работы подобных сервисов и в будущем использовать этот навык для работы не только с Google, но и с другими внешними сервисами. Это большой фронт работ, на котором вы сможете найти себе применение.

Содержание спринтов

Спринт 1

- Бизнес, Python и новые задачи
- Google Cloud Platform: API и сервисный аккаунт
- Google Sheets API: работа с Google-таблицами
- Google Drive API: работа с объектами Google-диска
- Рефакторинг учебного проекта
- Google API + FastAPI

Проект: Отчёт в Google Sheets для приложения QRKot

Неделя каникул



[50 часов]

Вебинары

Вебинары с наставником проводятся, в среднем, 1 раз в 2 недели

Знания из дополнительных уроков курса про мягкие навыки наконец-то пригодятся на практике.

Вы примете участие в разработке реального проекта, и сделаете работу над ошибками по командному взаимодействию. Всё будет настоящее: сторонний заказчик (некоммерческая организация), техническое задание, команда разработчиков (вы и ваши однокурсники), тимлид, проджект, тестировщик и СТО. Будут ревью, дедлайны, отчёты, релиз — полное погружение в реальность. В течение всей программы вы изучали навыки и осваивали компетенции необходимые разработчикам, а проектный месяц полигон для их испытаний.

Содержание спринтов

Спринт 1

- Что такое проектный месяц
- Первая неделя на новом проекте: онбординг, распределение ролей, знакомство с техзаданием, постановка задач
- Планирование и трекинг проекта: планирование, разработка, ревью, отчёты
- Презентация проекта и работа с фидбэком