

**Учебная программа курса “Python FullStack Developer. Разработчик на языке программирования Python”**

Учебный центр MYEDU, принадлежащий Myedu OÜ Утверждено 19.09.2024	
Название учебной программы	<b>Python FullStack Developer. Разработчик на языке программирования Python</b>
Группа учебных программ	153135 Разработка и анализ программного обеспечения и приложений / Tarkvara ja rakenduste arendus ning analüüs
Цели обучения	К концу обучения учащийся соответствует уровню Middle, знает основы программирования на языке Python и может разрабатывать простые приложения
Знания и навыки, приобретенные при успешном завершении учебной программы. Достижимые результаты обучения	Компетенции студента к концу обучения: <b>Python:</b> знает синтаксис и ключевые библиотеки <b>SQL и базы данных:</b> уверенно работает с SQL, PostgreSQL, SQLite и NoSQL базами данных (Redis, MongoDB) <b>Веб-разработка:</b> знает HTML, CSS, JavaScript для создания динамических веб-страниц <b>Контейнеризация:</b> управляет приложениями в Docker <b>Фреймворки:</b> разрабатывает веб-приложения и API на Django, Flask и FastAPI <b>API и интеграции:</b> создаёт и управляет REST API с использованием Django REST Framework, Flask и FastAPI; интегрирует с Google API <b>Парсинг и автоматизация:</b> настраивает парсинг с BeautifulSoup, Selenium и Scrapy <b>Асинхронность:</b> работает с WebSockets, Aiohttp и RabbitMQ для обмена сообщениями <b>Тестирование:</b> пишет тесты с Pytest и Unittest <b>DevOps и облака:</b> использует GitHub для управления кодом, автоматизирует процессы CI/CD и развертывает приложения на AWS, Azure и Google Cloud
Для кого этот курс. Целевая группа	Специалисты, которые работают или планируют работать в сфере ИТ, и хотят получить базовые знания и навыки, необходимые для работы в качестве Python Software Developer в Эстонии и за ее пределами
Требования на начало обучения, предъявляемые к обучаемому лицу, если они являются предпосылкой достижения результатов обучения	Для участия в обучении требуются базовые знания работы с компьютером. Учащийся должен иметь компьютер или ноутбук с оперативной памятью не менее 4 Гб с доступом в Интернет
Язык обучения	Русский язык
Общий объем обучения, в том числе доля аудиторной, практической и самостоятельной работы	400 академических часов
Самостоятельная работа	240 академических часов
Работа в аудитории	160 академических часов
Продолжительность обучения	12 месяцев
Содержание обучения: - Структура и объем обучения - Содержание обучения	Программа обучения создана на основе профессионального стандарта Noorem tarkvaraarendaja, tase 4 (Junior Software Developer, EstQF Level 4, Младший разработчик программного обеспечения, уровень 4) Теоретический материал и практическая работа под руководством преподавателя

	<p><b>Модуль 1. Python Core</b>  <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i>          Основы программирования на Python: установка среды разработки, создание и структура проекта, работа с переменными, типами данных, циклы, ветвления, функции и коллекции.          Классы и объекты: создание классов, конструкторы, наследование, множественное наследование.          Расширенные возможности: итераторы, генераторы, лямбда-функции, декораторы.          Исключения и работа с файлами: обработка ошибок, работа с файлами, сериализация.          Многопоточность: основы многозадачности в Python          Алгоритмы и структуры данных: линейный и бинарный поиск, рекурсия, сортировки, графы, оценка сложности (Big O).          Завершающий проект модуля.</p>
	<p><b>Модуль 2. Fullstack</b>  <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i>          Основы Web: HTTP, сеть, работа с HTML, CSS, JavaScript, ООП и асинхронностью.          Взаимодействие с бэкендом: работа с API и обмен данными.          Docker: контейнеризация, настройка Docker, работа с Docker Images и Docker Compose.          Основы SQL: запросы SELECT, JOIN, агрегатные функции, ключи и связи между таблицами.          Закрепление знаний через разработку проекта модуля.</p>
	<p><b>Модуль 3. Django</b>  <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i>          Знакомство с Django: Создание и настройка проекта, view-функции, пути.          HTML и шаблоны: Работа с тегами, стилями, фильтрами, подключение статических файлов.          Django ORM: Описание таблиц, CRUD-операции, связи между моделями.          Админзона: Настройка суперпользователя и админ-интерфейса.          Формы: Обработка данных и стилизация форм.          Тестирование: Pytest и Unittest для Django.          Миграции и развертывание: Управление миграциями, деплой на сервер, работа с Docker и CI/CD.          Облачные технологии: Работа с AWS, Azure, Google Cloud.          Проект модуля: Реализация полного веб-приложения.</p>
	<p><b>Модуль 4. API &amp; REST</b>  <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i>          Архитектура: Основы клиент-серверной архитектуры, MVC.          API и REST: Введение в API, REST, взаимодействие через сеть, API First.          Работа с внешними API: Примеры на основе Telegram API, обработка сообщений.          Django REST Framework: Создание REST API, сериализация данных, view-классы, аутентификация (JWT, Djoser).          Взаимодействие фронтенда и бэкенда: SPA vs MPA, CORS.          Документация и лимиты: Документация API, лимиты запросов, пагинация, фильтрация.          Парсинг сайтов: BeautifulSoup, Selenium, Scrapy.          Проект модуля: Реализация API с использованием Django REST Framework.</p>

	<p><b>Модуль 5. Flask &amp; FastAPI</b> 16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</p> <p>Flask и FastAPI: Введение в альтернативные фреймворки. Создание проектов на Flask и FastAPI.</p> <p>Миграции и базы данных: Alembic для миграций, SQLAlchemy для работы с базами данных.</p> <p>Асинхронность: WebSockets, Aiohttp.</p> <p>Работа с очередями и NoSQL: RabbitMQ, Redis, MongoDB.</p> <p>Google API: Интеграция с Google Sheets и Google Drive.</p> <p>Проектирование БД: Транзакции, ACID, повышение скорости, NoSQL и BigData.</p> <p>Проект модуля: FastAPI-приложение с интеграцией Google API.</p>
<p>Методы обучения</p>	<p><b>Аудиторная (онлайн) работа:</b> теоретический материал (лекции и обсуждение практических примеров)</p> <p><b>Практическая работа (онлайн и самостоятельно):</b> выполнение заданий, разработка программного обеспечения, оформление документации. Все практические задания выполняются в среде IntelliJ IDEA, github и проверяются преподавателем. Дается детальная обратная связь в письменном виде. Преподаватель объясняет новый теоретический материал, отвечает на вопросы студентов по новой теме и предыдущему домашнему заданию.</p> <p>Занятия проходят в Zoom и Discord. Каждый урок записывается, запись доступна закрытой группе в Slack.</p> <p>Между лекциями проходят квесты, задач, тесты, дополнительные видео уроки.</p> <p>Преподаватель доступен для связи каждый день. Если студент пропустил лекцию, посмотрел её в записи и появились вопросы по задачам/теории – он может написать преподавателю свои вопросы и получить ответ в любой день</p> <p><b>Практические упражнения:</b> задания на разработку алгоритмов, применение типов данных и структур, освоение различных инструментов сборки и запуска приложений</p> <p><b>Самостоятельная работа:</b> прочтение дополнительных материалов по теме и просмотр обучающих видео</p> <p>Общение с одноклассниками и преподавателями в закрытом чате в <b>Slack</b></p>
<p>Описание учебной среды и учебных помещений, предметов обстановки и оборудования, необходимых для достижения целей, поставленных в учебном плане, а также приобретения навыков и их соответствия законодательным требованиям охраны здоровья, если таковые установлены</p>	<p>Занятия с преподавателем проходят в форме онлайн видео конференций на платформе Zoom.</p> <p>В учебной среде проводятся теоретические и практические занятия, рассылаются домашние задания.</p> <p>Учащийся задает свои вопросы и получает обратную связь от учителя.</p> <p>Количество студентов в одной группе: до 10 человек.</p> <p>Один академический час длится 45 минут.</p> <p>Каждая встреча длится 2 академических часа.</p> <p>Занятия проводятся два раза в неделю по будням вечером.</p>
<p>Перечень учебных материалов</p>	<p>Учебный материал предоставляется учащимся в электронном виде.</p> <ul style="list-style-type: none"> <li>- конспекты лекций</li> <li>- шаблоны проектов и документации</li> <li>- учебные тренажеры</li> </ul>
<p>Требования к завершению обучения, включая методы оценки и критерии оценки</p>	<p>Посещение онлайн-занятий; выполнение домашних заданий</p> <p>Для оценки результатов обучения используется итоговое тестирование</p>

<p>Условия окончания обучения и выдаваемые документы</p>	<p>Для успешного завершения обучения требуется принять участие по крайней мере в 80% академических и практических занятий и выполнить все домашние задания. Достижение результатов обучения оценивается через практическую работу и итоговое тестирование.</p> <p><b>Сертификат</b> выдается учащемуся, который посетил не менее 80% уроков, выполнил все практические задания и успешно сдал итоговый тест.</p> <p><b>Свидетельство</b> об участии в обучении или об окончании обучения выдается ученику в том случае, если результаты обучения не были достигнуты, но ученик принимал участие в обучении.</p> <p><b>Справка</b> выдается в соответствии с количеством посещенных часов обучения, если учащийся участвовал по крайней мере в половине занятий.</p>
<p>Описание квалификации, опыта учёбы или работы проводящего обучение лица, которые необходимы для прохождения обучения</p>	<p>Python Software Developer с многолетним опытом в разработке и менторингу по Python и карьерным вопросам</p>