



Общество с ограниченной ответственностью «Ребреин»

ИНН 7727409582, ОГРН 1197746106161

Адрес: 123056, город Москва, Большая Грузинская ул, д. 36а стр. 5а, офис 13

Утверждено

Приказом № ПР-1 от 17.06.2025 г.

Генеральный директор

 Фролкина Е.А.
«17» июня 2025 г.

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА
– ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ
**«ПРОГРАММИРОВАНИЕ НА PYTHON: ОСНОВЫ ДЛЯ АВТОМАТИЗАЦИИ И
DEVOPS»**

Срок реализации: 1 месяц

Количество часов: 40 акад. ч.

Форма обучения: заочная форма

Формат обучения: с применением
исключительно дистанционных технологий

Возраст обучающихся: для лиц старше 17
лет, имеющих или получающих среднее
профессиональное и (или) высшее
образование

Москва, 2025 г.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Настоящая дополнительная профессиональная программа – программа повышения квалификации «Программирование на Python: основы для автоматизации и DevOps» (далее – Программа) разработана в соответствии с:

- Федеральным законом от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказом Министерства образования и науки РФ от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- Постановлением Правительства РФ от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Профессиональным стандартом 06.001 «Программист», утвержден приказом Министерства труда и социальной защиты Российской Федерации от 20.07.2022 № 424н;
- ФГОС высшего образования – бакалавриат по направлению 09.03.02 Информационные системы и технологии, утв. приказом Минобрнауки России от 19.09.2017 №926;
- Локальными нормативными актами ООО «Ребреин».

В данной программе учтены основные идеи формирования универсальных учебных действий учащихся и соблюдена преемственность с программами высшего и/или среднего профессионального образования.

Направленность программы: Программа имеет техническую направленность.

Адресат: программа ориентирована на специалистов, связанных с эксплуатацией и развитием информационных систем:

- системных администраторов и DevOps-инженеров, которым требуется освоить инструменты автоматизации и оптимизации рабочих процессов с использованием Python;
- SRE-специалистов, заинтересованных в применении Python для автоматизации задач сопровождения и интеграции, а также для работы с веб-приложениями;
- инженеров, планирующих профессиональное развитие в направлении DevOps и нуждающихся в базовых навыках программирования.

Требования к входным знаниям обучающегося: для освоения программы слушателям рекомендуется базовое понимание принципов работы операционных систем и компьютерных сетей, навыки работы с командной строкой, а также общий опыт эксплуатации программного обеспечения. Опыт программирования не обязателен, но будет полезен для более быстрого усвоения материала.

Актуальность реализации:

Python сегодня является одним из ключевых языков программирования в сфере DevOps и автоматизации. Его простота, широкий выбор библиотек и интеграция с современными инструментами позволяют специалистам решать задачи администрирования, автоматизации процессов и работы с инфраструктурой. Знание Python повышает эффективность работы инженеров, сокращает время на выполнение рутинных операций и открывает возможности для более глубокого взаимодействия с системами. Освоение базового уровня языка особенно важно для тех, кто строит карьеру в ИТ-инфраструктуре, администрировании или DevOps.

Отличительные особенности программы:

Программа сочетает классическое изучение языка с практической ориентацией на задачи DevOps и автоматизации. Участники не только осваивают базовый синтаксис и основы программирования, но и учатся применять Python в реальных рабочих сценариях: написание скриптов для администрирования, автоматизация рутинных задач, работа с API, основы фреймворка Django Rest Framework. Такой подход позволяет максимально быстро перейти от теории к практике, формируя прикладные навыки для профессиональной деятельности.

Объем и срок освоения программы: 40 академ. ч. в течение 1 мес.

Доступ к материалам Программы у обучающихся остаётся и после окончания периода обучения. Это позволяет повторять изученный материал в удобное время, восполнять пробелы в знаниях, а также возвращаться к практическим заданиям при решении рабочих задач. Такой формат способствует более глубокому закреплению навыков и поддерживает профессиональное развитие выпускников даже после завершения обучения.

Выдаваемый документ о квалификации: удостоверение о повышении квалификации и/или сертификат об успешном освоении программы.

Цели и задачи программы:

Цель программы — сформировать у слушателей базовые знания и практические навыки программирования на Python, необходимые для решения задач автоматизации и интеграции в сфере DevOps, а также для дальнейшего профессионального развития в области администрирования и эксплуатации ИТ-систем.

Программа направлена на решение следующих основных задач:

Обучающие:

- дать слушателям представление о синтаксисе и структуре языка Python;
- обучить основным типам данных, управляющим конструкциям и функциям;
- познакомить с принципами работы с файлами, модулями и библиотеками;
- сформировать базовые знания объектно-ориентированного программирования;
- показать практические примеры применения Python для задач DevOps и автоматизации.

Развивающие:

- развить алгоритмическое мышление и навыки постановки задач;
- сформировать умение анализировать и отлаживать программный код;
- развить способность выбирать оптимальные решения для автоматизации процессов;
- стимулировать самостоятельную работу с документацией и внешними библиотеками.

Воспитательные:

- формировать ответственное отношение к качеству кода и документации;
- воспитывать культуру командной работы и обмена знаниями;
- развивать интерес к постоянному профессиональному развитию в сфере DevOps и автоматизации.

Планируемые результаты:

Знания:

- синтаксис языка Python и его основные конструкции;
- встроенные типы данных и способы их обработки;

- основы работы с функциями, модулями и библиотеками;
- принципы объектно-ориентированного программирования (ООП) в Python;
- основы работы с файлами и обработкой ошибок (исключениями);
- базовые подходы к созданию и использованию API (на примере Django Rest Framework);
- роль Python в автоматизации процессов и в DevOps-среде.

Умения:

- писать простые и средние по сложности программы на Python;
- использовать ветвления и циклы для управления логикой программы;
- структурировать код с помощью функций и модулей;
- применять ООП для моделирования и решения прикладных задач;
- работать с файлами, данными и внешними библиотеками;
- находить и исправлять ошибки в коде;
- использовать Python для автоматизации задач системного администрирования и DevOps.

Навыки:

- разработки рабочих скриптов для автоматизации повседневных задач;
- оптимизации кода и поиска узких мест в программах;
- интеграции Python-скриптов с внешними сервисами и инструментами;
- применения Python в типичных сценариях DevOps (работа с конфигурациями, логами, API);
- использования Python-библиотек и инструментов для решения практических задач.

Перечень профессиональных компетенций, на получение которых направлено обучение:

На основе профстандарта 06.001 «Программист»:

- А/02.3 Написание программного кода с использованием языков программирования, определения и манипулирования данными в базах данных;
- А/04.3 Работа с системой управления версиями программного кода
- В/04.4 Рефакторинг, оптимизация и инспекция программного кода

Таким образом, в результате освоения программы у обучающихся формируются следующие профессиональные компетенции:

ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;

Организационно-педагогические условия реализации программы дополнительного профессионального образования

Язык реализации образовательной программы: обучение проводится на русском языке.

Форма обучения: заочная форма.

Особенности реализации программы: программа реализуется с использованием электронного обучения и исключительно дистанционных образовательных технологий.

Условия набора: на обучение принимаются все желающие лица, оплатившие обучение и заключившие договор об образовании. Обучение проходит в индивидуальном формате без

формирования учебных групп. Обучающийся самостоятельно определяет время освоения Программы.

Формы проведения занятий:

- занятия в текстовом формате;
- практическая работа;
- самостоятельная работа с литературой;
- индивидуальные вопросы.

Материально-техническое оснащение

Материальное обеспечение программы

Занятия проводятся в системе дистанционного обучения «Rebrain». Каждый обучающийся и педагог оснащены доступом к системе дистанционного обучения: <https://rebrainme.com/>.

У педагога дополнительного профессионального образования имеется необходимое оборудование средства для реализации программы: ноутбук с подключением к интернету, программное обеспечение.

Методическое обеспечение программы

Программа обеспечена:

- учебно-методическими материалами (текстовые занятия, полезными материалами);
- практическими заданиями.

Кадровое обеспечение:

К реализации программы в качестве педагогов дополнительного образования допускаются лица:

1) отвечающее одному из требований:

а) имеющее высшее образование или среднее профессиональное образование в рамках укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки»;

б) имеющее высшее образование либо среднее профессиональное образование в рамках иных укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительной общеобразовательной общеразвивающей программе, реализуемой ООО «Ребреин», и получение при необходимости дополнительного профессионального образования педагогической направленности;

в) успешно прошедшее промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующей направленности дополнительной общеобразовательной общеразвивающей программе;

2) не имеющее ограничений на занятие педагогической деятельностью, установленных законодательством Российской Федерации;

3) прошедшее обязательный предварительный (при поступлении на работу) и периодические медицинские осмотры (обследования), а также внеочередные медицинские осмотры (обследования) в порядке, установленном законодательством Российской Федерации.

Реализация Программы также возможна лицами, привлекаемыми на условиях гражданско-правового договора в соответствии с действующим законодательством РФ.

2. УЧЕБНЫЙ ПЛАН

№ п/ п	Наименование модуля	Количество часов			Формы контроля / аттестация
		Всего	Теория	Практика	
1	Модуль 1 “Онбординг”	1	0,5	0,5	Входное тестирование
2	Модуль 2 “Python”	35	15	20	Практическое задание
3	Итоговая аттестация	4		4	Итоговое практическое задание

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

№ п/п	Наименование модуля	1 неделя	2 неделя	3 неделя	4 неделя
1	Модуль 1 “Онбординг”	1			
2	Модуль 2 “Python”	9	10	10	6
3	Итоговая аттестация				4 А

4. РАБОЧАЯ ПРОГРАММА

Модуль 1. Онбординг

Теория 0,5 академ. ч. Практика 0,5 академ. ч.

Модуль состоит из следующих тем:

Тема 1: Онбординг

В модуле обучающемуся предоставляется вводный конспект, содержащий общую информацию о программе, структуре курса, форматах взаимодействия с материалами и ожидаемых результатах обучения.

Предусмотрено прохождение входного тестирования, включающего 7 вопросов, направленных на закрепление информации из онбординга. В рамках блока обучающийся выполняет задание по целеполаганию: формулирует свою цель прохождения программы, указывает желаемые навыки по окончании обучения, а также оценивает текущий уровень своих знаний, выбрав один из предложенных вариантов.

Модуль 2. Python

Теория 15 академ. ч. Практика 20 академ. ч.

Модуль состоит из следующих тем:

Тема 1: Введение

Содержание: Установка Python. Hello, World. IDE. Ввод данных с консоли. Практическое задание.

Тема 2: Типы данных. Часть 1

Содержание: Динамическая типизация. Типы данных. Введение. Числа. Стока. Логический тип данных. Практическое задание.

Тема 3: Типы данных. Часть 2

Содержание: Списки. Кортежи. Словари. Множества. Комбинирование типов данных. Дополнительные возможности коллекций. Практическое задание.

Тема 4: Ветвление

Содержание: Условный оператор if. Алгоритмы работы оператора. Логические операторы. Тернарные операторы. Практическое задание.

Тема 5: Циклы

Содержание: Цикл while. Цикл for. Итераторы. Практическое задание.

Тема 6: Работа с файлами

Содержание: Чтение файлов. Запись файлов. Кодировка файлов. Режимы b и +. Закрытие файлов. Структуры данных и файлы. Практическое задание.

Тема 7: Функции. Часть 1

Содержание: Функция Python. Области видимости. Вызов функции из функции. Имена функций. Практическое задание.

Тема 8: Функции. Часть 2

Содержание: Генераторы. lambda-функция. Функции map() и filter(). Практическое задание.

Тема 9: Модули

Содержание: Импорт модулей. Виртуальное окружение venv. Практическое задание.

Тема 10: Библиотеки

Содержание: Модули: os, requests, time, logging. Запуск скрипта с параметрами. Практическое задание.

Тема 11: ООП. Часть 1.

Содержание: основы Объектно-ориентированного программирования. Практическое задание.

Тема 12: ООП. Часть 2.

Содержание: Инкапсуляция. Наследование. Полиморфизм. Практическое задание.

Тема 13: Исключения

Содержание: что такое “исключения”. Написание собственного исключения. Практическое задание.

Тема 14: Введение в API. Django Rest Framework.

Содержание: знакомство с RESTful API на примере Django-Rest-Framework. Поля, используемые в Django. Практическое задание.

Модуль направлен на формирование у обучающихся базовых знаний и навыков программирования на языке Python. Каждая тема включает текстовое занятие с теоретическим материалом и пошаговыми инструкциями, после изучения которого предлагается практическое задание. Практические задания рассчитаны на 1-2 академических часа. Выполнение заданий предполагает отправку решения на проверку через личный кабинет обучающегося. Критерии оценки прописаны в описании к каждому заданию. В случае корректного выполнения выставляется зачёт. Если работа содержит ошибки, задание возвращается на доработку. При повторной неудачной попытке (после двух доработок) обучающийся получает «незачёт».

5. МЕТОДИЧЕСКИЕ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Программа обеспечена системой дистанционного обучения <https://rebrainme.com/>.

Педагогические технологии:

- технология дифференцированного обучения;
- технология разноуровневого обучения;
- технология развивающего обучения;
- технология проблемного обучения;
- технология дистанционного обучения.

Методы обучения:

- словесный, наглядный практический;
- объяснительно – иллюстративный;
- частично-поисковый, исследовательский проблемный;
- игровой, дискуссионный.

Электронно-библиотечные ресурсы и системы, информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.RU.
2. Собственные учебные материалы [Электронный ресурс]: <https://rebrainme.com/courses/python>
3. Официальная документация Python [Электронный ресурс]: <https://www.python.org/>
4. Официальная документация Django [Электронный ресурс]: <https://docs.djangoproject.com/en/3.2/>

6. ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ

Оценочные материалы:

Для отслеживания результатов освоения программы среди слушателей проводится текущий контроль и итоговое оценивание.

Текущий контроль

Осуществление текущего контроля проводится после занятий в виде написания практических заданий или тестирований. Тематика и условия выполнения практических заданий расписаны в личном кабинете обучающегося в СДО. Педагог проверяет решение и принимает решение о принятии решения (зачет), о необходимости доработать решение или о незачете. Если промежуточный контроль представлен в виде тестирования, подсчет верных ответов и выставление оценки «зачёт» и «незачёт» происходят в автоматическим решиме в СДО.

Тема 3: Типы данных. Часть 2

У вас есть несколько строк лога:

May 18 11:59:18 PC-00102 plasmashell[1312]: kf.plasma.core: findInCache with a lastModified timestamp of 0 is deprecated

May 18 13:06:54 ideapad kwin_x11[1273]: Qt Quick Layouts: Detected recursive rearrange. Aborting after two iterations.

May 20 09:16:28 PC0078 systemd[1]: Starting PackageKit Daemon...

May 20 11:01:12 PC-00102 PackageKit: daemon start

Скопируйте их к себе и создайте из них список (список строк).

2.1. Создайте алгоритм заполнения словаря, подходящий для любой строчки лога. Словарь должен содержать в себе такую информацию:

```
'time': <дата/время>
'pc_name': <имя компьютера>
'service_name': <имя сервиса>
'message': <сообщение лога>
```

Еще раз обращаю ваше внимание на то, что алгоритм заполнения должен быть универсальным для всех данных строк.

Итоговое оценивание

В конце программы обучающиеся сдают итоговую аттестацию.

Практическое задание

Используя полученные знания, напишите клиентскую часть нашего приложения.

Идея клиента следующая: сбор и отправка данных о системе на сервер, который мы сделали в прошлом блоке. Это свободное задание, в котором вам нужно проявить фантазию. Тем не менее, программа должна обеспечивать следующие функции:

1. При запуске программы, она должна регистрировать наш клиент на сервере (POST запрос на api/servers/add)
 - Для определения внешнего IP адреса используйте `requests.get('https://ifconfig.me/ip').text`, либо используйте любой другой доступный способ определения IP.
 - В поле name подавать значение `hostname`
 - В поле description брать значение из переменной окружения
2. Программа должна обрабатывать данные о системе, полученные от утилиты `psutil`, и отправлять их на сервер с интервалом в 1 минуту в формате:

[...]

Это не окончательный список данных. Вы можете добавить те данные, которые посчитаете нужными. И даже что-то убрать.

3. Программа должна логировать следующие этапы работы:
 - INFO Старт программы,
 - INFO Успешная регистрация сервера,
 - INFO Полученный в результате работы массив данных о системе,
 - ERROR При отказе отправки данных или регистрации сервера.

Вы также можете дописать нашу серверную часть и добавить модель и соответствующее представление для приема данных. Но это не обязательно.

Обрабатывайте исключения, которые вызывают запросы. В этом вам помогут исключения из `requests.exceptions`.

Можете организовать проверку наличия данного сервера в регистрируемых серверах, чтобы не делать повторную регистрацию.

Также вы можете добавить логирование тех моментов, которые посчитаете нужными.

Результаты текущего контроля и итогового оценивания отображаются в личном кабинете слушателя в системе дистанционного обучения <https://rebrainme.com/>.

По результатам сдачи текущего контроля и итогового оценивания педагог даёт обратную связь слушателям, отмечает их сильные стороны и обращает внимание на зоны для развития. При необходимости педагог может повторить пройденные темы со слушателями, если установлен факт плохого закрепления и усвоения темы у слушателей.