АНО «Развитие человеческого капитала»
Обособленное подразделение «Техноград»

УТВЕРЖДЕНО
Приказом Генерального директора
АНО «Развитие человеческого капитала»
№ ОД-332/25 от «1» октября 2025 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

Наименование программы

«Программист Python (уровень) № 1»

Категория слушателей: начинающие аналитики данных, Python-программисты, разработчики бэкенда, data scientist, технические специалисты по информационной безопасности, пентестеры, программисты встраиваемых систем, разработчики роботизированных систем, специалисты по машинному обучению, консультанты в области информационных технологий, менеджеры продуктов в области информационных технологий, разработчики моделей Big Data, IT-консультанты, автоматизаторы тестирования, DevOps-инженеры, IT-директора, веб-разработчики, AI-инженеры, Embedded-разработчики, лица старше 18 лет, заинтересованные в получении новых компетенций, студенты вузов и образовательных учреждений СПО, обучающиеся по направлениям подготовки «Информационные системы и технологии», «Информатика и вычислительная техника», «Программная инженерия»

Объем: 72 часа.

Форма обучения: электронная.

Форма документа, выдаваемого по результатам освоения программы: сертификат о прохождении курса.

УЧЕБНЫЙ ПЛАН[1]

Наименование компонентов программы	Обязательные аудиторные учебные занятия (ч.)		Внеаудиторная (самостоятельная) учебная работа (ч.)	Всего учебной нагрузки (ч.)
	всего	в т.ч. практических и семинарских занятий		
Модуль 1. Основы программирования на Python.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 2. Основные конструкции языка.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 3. Возможности объектно-ориентированного программирования (ООП) в Python.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 4. Синтаксический сахар и полезные пакеты в Python.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 5. Введение в архитектуру клиент-сервер.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 6. Промышленные средства разработки.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 7. Создание новых приложений.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 8. Работа с базами данных.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 9. Разработка серверных приложений.	**-**	-	4	**4**
Текущий контроль: тестирование.	**2**	2	-	**2**
Модуль 10. Тестирование и оптимизация приложений.	**-**	-	5	**5**
Текущий контроль: тестирование.	**2**	2	-	**2**

[1] В учебном плане отражены занятия, проводимые в формате электронного обучения.

Модуль 11. Развертывание приложений.	**-**	-	5	**5**
Текущий контроль: тестирование.	**2**	2	-	**2**

Наименование тем, разделов (разделов)	**Содержание тем, разделов (модулей)**
Модуль 1. Основы программирования на Python.	Ключевые особенности языка (многопарадигмальность, интерпретация, многоплатформенность, масштабируемость, программирование GUI). Дзен Python. Стандартизация. Установка интерпретатора. Переменные. Базовые типы данных. **Текущий контроль № 1.** Тестирование. **Самостоятельная работа № 1.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 2. Основные конструкции языка.	Логические операторы. Укороченная оценка. Циклы в Python. Присваивание кортежей. Операторы break и continue внутри циклов. Запись цикла в одну строку. Генерация списков и словарей. Функции и лямбда-конструкции. Параметры типа ключ-значение. **Текущий контроль № 2.** Тестирование. **Самостоятельная работа № 2.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 3. Возможности объектно-ориентированного программирования	Волшебный метод инициализации класса. Пример наследования. Динамическое изменение классов. Статические и классовые методы. Методы с property. Обработка исключений.

СОДЕРЖАНИЕ ПРОГРАММЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ, РАЗДЕЛАМ (МОДУЛЯМ)

(ООП) в Python.	Итераторы. Генераторы. **Текущий контроль № 3.** Тестирование. **Самостоятельная работа № 3.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 4. Синтаксический сахар и полезные пакеты в Python.	Декораторы в Python. Подключение модуля из стандартной библиотеки. Использование псевдонимов. Инструкция from. Создание своего модуля на Python. Применение логирования. JSON. Типы Counter, OrderedDict. **Текущий контроль № 4.** Тестирование. **Самостоятельная работа № 4.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 5. Введение в архитектуру клиент-сервер.	Сетевой протокол. Семь уровней модели OSI. Разделение приложений по уровням. Типы клиент-серверной архитектуры. Преимущества и недостатки использования архитектуры клиент-сервер. Архитектура микросервисов. **Текущий контроль № 5.** Тестирование. **Самостоятельная работа № 5.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 6. Промышленные средства разработки.	Начало работы над проектом. Настройки PyCharm. Настройка виртуального окружения. Работа с отладчиком. Инициализация репозитория. Создание и слияние ветвей. Разрешение конфликтов. Размещение кода. Рефлексия по процессу. Интерфейс Postman. Коллекции запросов. **Текущий контроль № 6.** Тестирование. **Самостоятельная работа № 6.** Изучение видеолекций и учебно-методического материала

	для подготовки к итоговой аттестации (пункт 6.3 программы).
Промежуточная аттестация	**Выполнение практического задания.**
Модуль 7. Создание новых приложений.	Проектирование REST API. Диаграммы последовательности (sequence диаграммы). MVC Pattern. Преимущества и недостатки Django. Запуск сервера разработки. Логирование ошибок в Sentry. **Текущий контроль № 7.** Тестирование. **Самостоятельная работа № 7.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 8. Работа с базами данных.	Типы баз данных. Django ORM. Модели данных. Метаданные. Фильтрация данных. Получение одного объекта, Сортировка. Формирование цепочки запросов. Работа с миграциями. Добавление миграций в приложение. **Текущий контроль № 8.** Тестирование. **Самостоятельная работа № 8.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 9. Разработка серверных приложений.	Подключение административной панели. Создание суперпользователя. Аутентификация и авторизация. Создание пользователей и групп. Django Rest Framework (DRF). Сериализаторы. Представления (views). OpenAPI. Пример подключения swagger. **Текущий контроль № 9.** Тестирование. **Самостоятельная работа № 9.** Изучение видеолекций и учебно-методического материала

	для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 10. Тестирование и оптимизация приложений.	Тестирование серверных приложений. Уровни тестирования. Описание цикла TDD. Оптимизация производительности. Техники отладки. Брейкпоинты. JMeter, ApacheBench и Locust. Рефакторинг исходного кода. **Текущий контроль № 10.** Тестирование. **Самостоятельная работа № 10.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Модуль 11. Развертывание приложений.	Автоматизация развёртывания с помощью Docker. Архитектура Docker. Запуск нескольких контейнеров с Docker Compose. Непрерывная интеграция и доставка. Конфигурация GitLab CI/CD. Причины возникновения уязвимостей. Экранирование специальных символов. Межсайтовый скриптинг. SQL-инъекции. **Текущий контроль № 11.** Тестирование. **Самостоятельная работа № 11.** Изучение видеолекций и учебно-методического материала для подготовки к итоговой аттестации (пункт 6.3 программы).
Итоговая аттестация	**Выполнение практического задания.**