



**приоритет2030<sup>+</sup>**  
**лидерами становятся**



## **Zero coding и генеративный искусственный интеллект в образовании**

Программа дополнительной профессиональной переподготовки «Zero coding и генеративный искусственный интеллект в образовании» предназначена для подготовки руководителей служб и подразделений в сфере информационно-коммуникационных технологий.

**Срок обучения – 9 месяцев**

**Форма обучения – очно-заочная с применением дистанционных образовательных технологий (онлайн-обучение)**

**Присваиваемая квалификация – Менеджер по информационным технологиям**

**Стоймость - бесплатно**

### **В ходе обучения Вы научитесь:**

- применять алгоритмы и функции на языке Python, практически работать с библиотеками для языка программирования;
- применять конкретные конструкторы для создания цифровых решений – приложения, сайты, информационные ресурсы;
- использовать нейросети для создания контента, визуализации, видео и аудио материалов, а также грамотно писать промты (запросы) для работы с нейросетями (на примере GPT и аналогов).

## Содержание программы

№ п/п	Наименование и краткое содержание раздела (модуля)	Объем, часов
<b>Модуль 1 «Современные тенденции цифровой трансформации образования»</b>		
1.1	Современные тренды образования <i>Мультимодальная педагогика. Интеграция учебных сред. Проблемно-ориентированный подход в образовательном процессе. Метавселенная образования.</i>	4
1.2	Сквозные цифровые технологии в образовании <i>Обзор актуальных сквозных цифровых технологий, используемых в образовании, и применимых в образовательном процессе</i>	4
1.3	Искусственный интеллект и машинное обучение <i>Современное понятие машинного обучения и искусственного интеллекта. Определение. Основные термины. Классификация алгоритмов.</i>	6
1.4	Искусственные нейронные сети (НС) <i>Понятие НС. Модель нейрона, функция активации. Полносвязанные НС, сверточные и рекуррентные НС и их применение в различных отраслях.</i>	8
1.5	Генеративный искусственный интеллект (GPT) <i>Понятие генеративного искусственного интеллекта. Виды генеративного ИИ. Возможности применения в различных отраслях. Возможности применения генеративного ИИ в отрасли «Образование». Автоматизация процессов. Генерация контента. Персонализация и создание индивидуальных образовательных треков</i>	8
1.6	Этика искусственного интеллекта <i>Этические принципы в основе ИИ решений. Предвзятость алгоритмов. Недоверие к профессиональному интеллекту. Приватность данных. Этические кодексы в сфере ИИ.</i>	8
<b>ВСЕГО</b>		<b>38</b>
1.15	Промежуточная аттестация. Тестирование	2
<b>Модуль 2 «Введение в программирование на языке Python»</b>		
2.1	Введение в Python <i>Основные принципы и понятия языка Python. Основные встроенные типы данных. Основные операции языка Python. Консольный ввод и вывод.</i>	6
2.2	Алгоритмы <i>Понятие алгоритма, формы записи и свойства алгоритма. Вычислительная сложность алгоритма и методы ее оценки.</i>	6
2.3	Алгоритмы общего назначения <i>Алгоритмы сортировки, поиска максимума, минимума и заданного значения в списке. Быстрые и небыстрые алгоритмы. Реализация их на языке Python</i>	6
2.4	Управляющие конструкции <i>Алгоритмы. Программные блоки. Логические операторы. Циклы. Условные операторы</i>	6
2.5	Функции	6

	<i>Определение функции. Аргументы функции: обязательные, необязательные. Передача аргументов: по значению, по ссылке. Области видимости переменных. Рекурсия</i>	
2.6	Работа с коллекциями <i>Коллекция. Индексация. tuple. list. Срезы. Списковые включения. Операции над списками.</i>	6
2.7	Организация кода. Работа с файловой системой <i>Модули. Пакеты, файл __init__.py. Зависимости: ключевое слово import, конструкция from ... import ... PYTHONPATH. Точка входа в приложение. Работа с файлами.</i>	6
2.8	Основы объектно-ориентированного программирования (ООП) <i>ООП. Наследование. Полиморфизм. Инкапсуляция. ООП в Python. Метод __init__. Ключевое слово self. Свойства/атрибуты класса. Методы/функции. Наследование в Python. Статические методы, атрибуты класса.</i>	6
2.9	Исключения <i>Инкапсуляция/скрытие реализации в Python. Обработка исключительных ситуаций. Конструкция: try ... catch ... Пользовательские классы исключений.</i>	8
2.10	Внешние библиотеки <i>Загрузка и подключение (импорт) внешних библиотек Python. Библиотеки NumPy, Matplotlib, skimage. Их назначение</i>	8
<i>ВСЕГО</i>		64
2.12	Промежуточная аттестация. Тестирование	2
<b>Модуль 3 «Low code/no code программирование: альтернатива разработке»</b>		
3.1	Low code/no code программирование <i>Основные понятия. Примеры применения. Обзор готовых решений</i>	8
3.2	Web – дизайн <i>Основные понятия веб-дизайна. Планирование пользовательского интерфейса, создание дизайна (цветовые решения, раскладки и графики страниц). Основные принципы и правила веб-дизайна Шрифты и типографика в веб-дизайне.</i>	10
3.3	Low code платформы и сервисы с поддержкой языка Python: разработка решений с минимумом кода <i>Развертывание продукта</i>	10
3.4	No code платформы и сервисы: разработка решений из готовых элементов без программирования <i>Разработка сайтов</i>	12
3.5	No code платформы и сервисы: разработка решений из готовых элементов без программирования <i>Разработка веб-приложений</i>	12
<i>ВСЕГО</i>		52
3.6	Промежуточная аттестация. Тестирование	2
<b>Модуль 4 «Цифровые решения на основе нейросетей»</b>		
4.1	Применение нейросетей для создания контента для образования. Обзор готовых решений <i>Возможности использования НС в образовании: примеры. Обзор готовых решений на рынке (бесплатных, платных) для создания аудио, видео, визуального контента, текстовой информации.</i>	8
4.2	Работа с текстом (файлы, контент план)	10
4.3	Работа с визуалом (презентации, фотографии, стиль)	10

4.4	<b>Создание видеоконтента</b> <i>Создание и монтаж видеоконтента. Обработка аудио</i>	12
4.5	Промпting в работе GPT ( <i>на примере ChatGPT</i> ) <i>Понятие промта. Правила составления промта для получения качественного ответа. Правильные приемы при общении с нейросетью. Шаблоны написания промтov.</i>	12
	<i>ВСЕГО</i>	52
4.5	Промежуточная аттестация. Тестирование	2
	<i>Практика: защита проектной работы</i>	
5.1	Определение тематического поля и темы проекта. Поиск и анализ проблемы. Постановка цели проекта.	6
5.2	Анализ имеющейся информации. Поиск оптимального способа достижения цели проекта (анализ альтернативных решений), построение алгоритма деятельности. Анализ ресурсов.	12
5.3	Составление плана реализации проекта: пошаговое планирование работ.	6
5.4	Изучение возможностей использования результатов проекта (выставка, продажа, включение в банк проектов, публикация).	6
5.5	Анализ результатов выполнения проекта	6
	<i>ВСЕГО:</i>	36