



Правительство Санкт-Петербурга
Комитет по образованию

Государственное бюджетное общеобразовательное учреждение
Лицей №126
Калининского района Санкт-Петербурга

Принята

на педагогическом совете

Протокол № 1 от 30 августа 2024 г.

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Дополнительная общеразвивающая программа

«Введение в Data Science»

возраст обучающихся 15 - 18 лет
срок освоения: 36 недель

Разработчик:
Авдеев Александр Сергеевич,
педагог дополнительного образования

I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Основные характеристики программы

Дополнительная общеразвивающая программа «Введение в Data Science» (далее - ДОП) имеет **техническую направленность**. Data Science — это наука, которая помогает анализировать данные и находить в них важные закономерности. Современные технологии делают возможным сбор и обработку больших объёмов данных, что активно используется в самых разных областях — от медицины до бизнеса.

Актуальность программы:

В условиях роста значимости данных в мире важно, чтобы подростки уже на начальных этапах обучения могли познакомиться с основными методами и инструментами анализа данных. Программа актуальна для школьников, интересующихся технологиями, математикой и программированием, и предоставляет практические знания для работы с данными, построения моделей и их анализа.

Адресат программы:

Курс рассчитан на подростков в возрасте 15-18 лет, обучающихся в 8-11 классах, и требует глубоких начальных знаний в программировании на python и математике.

Отличительная особенность ДОП

Отличительная особенность заключается в том, что программа составлена таким образом, чтобы обучающиеся могли овладеть всем комплексом знаний по организации исследовательской деятельности, выполнению проектной работы, познакомиться с требованиями, предъявляемыми к оформлению и публичному представлению результатов своего труда, а также приобрести практические навыки работы в области аналитики данных и машинного обучения.

Уровень освоения ДОП

Дополнительная общеразвивающая программа «Введение в Data Science» рассчитана на общекультурный уровень освоения программы.

Объем и срок освоение программы

Общее количество учебных часов, запланированных на весь период обучения: 72 академических часа. Срок реализации программы 36 недель.

Цель и задачи ДОП

Цель программы:

Формирование у обучающихся начальных знаний и навыков в области анализа данных и визуализации, а также понимания важности данных в современном мире.

Задачи программы:

Обучающие (предметные):

- Формировать знания о понятиях «данные», «анализ данных», «машинное обучение».
- Научить основам работы с языком программирования Python и библиотеками для анализа данных (Pandas, NumPy, Matplotlib).
- Привить навыки работы с реальными наборами данных, обработки и визуализации данных.
- Развить навыки критического мышления и анализа информации.
- Способствовать формированию проектного мышления через выполнение практических заданий и проектов.

Развивающие (метапредметные):

- сформировать навыки логического мышления, умения работать по предложенным инструкциям, программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развить внимательность, аккуратность и изобретательность при работе с техническими устройствами, разработке приложений и выполнении учебных проектов;
- сформировать творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, грамотно работать с критикой и извлекать из неё пользу, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.

Воспитательные (личностные):

- способствовать формированию профессионального самоопределения в IT-сфере;
- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций и устройств;
- развить стремление к получению качественного законченного результата в проектной деятельности;
- сформировать навыки проектного мышления, работы в команде, сохранение дисциплины, эффективно распределять обязанности при реализации проекта, требующего от участников знаний и умений из различных направлений.

Планируемые результаты освоения программы

Личностные:

- Развитие интереса к наукам о данных и IT-технологиям.
- Сформированы навыки проектного мышления, работы в команде, сохранение дисциплины, эффективно распределять обязанности при реализации проекта, требующего от участников знаний и умений из различных направлений.

Метапредметные:

- сформированы навыки логического мышления, умения работать по предложенным инструкциям, программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развиты внимательность, аккуратность и изобретательность при работе с техническими устройствами, разработке приложений и выполнении учебных проектов;
- сформированы творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, грамотно работать с критикой и извлекать из неё пользу, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.

Предметные:

- Знания основ Data Science и работы с инструментами для анализа данных.
- Навыки визуализации данных и работы с библиотеками Python.
- Умение применять полученные знания для решения реальных задач.

Теоретическая подготовка включает: владение теоретическими знаниями (по основным разделам учебного плана программы), системой понятий; владение специальной терминологией; знание правил и алгоритмов деятельности.

Практическая подготовка ребенка включает: практические умения и навыки, предусмотренные программой; владение специальным оборудованием и оснащением; творческие навыки.

Организационно-педагогические условия реализации программы

Язык реализации программы – образовательная деятельность по дополнительной общеразвивающей программе «Введение в Data Science» осуществляется на русском языке.

Форма обучения: очная.

Особенности реализации программы: реализация программы проходит в течение всего учебного года, включая каникулярное время, с использованием электронного обучения и дистанционных образовательных технологий.

Особенности организации образовательного процесса: образовательный процесс по реализации ДОП «Введение в Data Science» строится на принципах научности, доступности, наглядности, систематичности и последовательности, связи теории с практикой, сознательности и активности обучения, индивидуальном подходе в обучении с учётом психолого-педагогических особенностей каждого обучающегося.

Условия набора в коллектив: набор в объединение осуществляется по желанию при отсутствии медицинских противопоказаний.

Условия формирования групп: группы формируются разновозрастные.

Количество детей в группе: 12 человек.

Режим занятий: 1 раз в неделю по 2 часа – 72 академических часа в год.

Формы организации занятий: занятия проводятся по группам, программой предусматриваются как аудиторные, так и внеаудиторные формы организации занятий.

Формы проведения занятий: основной формой организации деятельности является учебное занятие традиционной формы.

Формы организации деятельности учащихся на занятии: фронтальная, индивидуальная, групповая.

Материально-техническое оснащение ДОП:

Материально-техническое оснащение ДОП:

- проектор и экран для демонстрации учебного материала;
- персональные компьютеры для обучающихся;
- Python с установленными библиотеками Pandas, NumPy, Matplotlib, TensorFlow, Keras, PyTorch.

II. УЧЕБНЫЙ ПЛАН

№ п/п	Название раздела, темы	Кол-во часов			Формы контроля /аттестации
		всего	теория	практика	
1.	Введение в машинное обучение	8	2	6	Беседа, входной мониторинг.
2.	Основы машинного обучения	16	6	10	Реализация и защита проектов.
3.	Машинное обучение для больших данных	24	8	16	Реализация и защита проектов.
4.	Итоговый проект	24	4	20	Реализация и защита проектов.
ИТОГО		72	20	52	

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

III. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график реализации дополнительной
общеразвивающей программы «Введение в Data Science»
на 2024 - 2025 учебный год

Педагог: Авдеев Александр Сергеевич

Год обучения	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий
1 год	01.09.24	31.05.25	36	36	72	1 раз в неделю по 2 часа

III. РАБОЧАЯ ПРОГРАММА

Содержание программы

Модуль 1: Введение в машинное обучение (10 часов)

1.1 Базовый Python повторение (2 часа)

- Основы синтаксиса Python: переменные, типы данных, операции.
- Управляющие конструкции: циклы, условные операторы.
- Основные структуры данных: списки, кортежи, множества, словари.
- Введение в функции: определение функций, аргументы и параметры.
- Работа с файлами: чтение и запись данных.

1.2 Предобработка данных (2 часа)

- Импорт данных из различных источников: CSV, Excel, базы данных.
- Обработка пропущенных данных: методы заполнения и удаления.
- Кодирование категориальных признаков: one-hot encoding, label encoding.
- Нормализация и стандартизация данных.
- Обнаружение и обработка выбросов.

1.3 Исследовательский анализ данных (2 часа)

- Визуализация данных с использованием Matplotlib и Seaborn.
- Расчет основных статистических показателей: среднее, медиана, мода.
- Построение графиков распределения данных, коробчатых диаграмм и тепловых карт.
- Взаимосвязь между признаками: корреляция, ковариация.
- Создание отчетов и дашбордов для презентации данных.

1.4 Статистический анализ данных (2 часа)

- Введение в статистику: дисперсия, стандартное отклонение, вероятность.
- Гипотезы и тестирование гипотез: t-тест, chi-square тест.
- Анализ взаимосвязей между признаками: регрессия, корреляционный анализ.
- Оценка и интерпретация статистических моделей.
- Применение статистических методов в реальных задачах.

1.5 Итоговый проект по первому модулю (2 часа)

- Постановка задачи: выбор реального набора данных.
- Предобработка и анализ данных.
- Визуализация и статистический анализ.
- Создание отчета и презентации результатов.

Модуль 2: Основы машинного обучения (16 часов)

2.1 Линейные модели в машинном обучении (4 часа)

- Введение в линейную регрессию: простая и множественная линейная регрессия.
- Методы регуляризации: Lasso, Ridge.
- Логистическая регрессия: основы и применение.
- Оценка моделей: коэффициент детерминации (R^2), MSE, RMSE.
- Практическое применение линейных моделей на реальных данных.

2.2 Обучение с учителем и оценка качества моделей (8 часов)

- Классификация: введение в алгоритмы классификации (Decision Trees, KNN, SVM).

- Регрессия: полиномиальная регрессия, поддерживающие векторы.
- Метрики качества моделей: accuracy, precision, recall, F1-score.
- Подбор гиперпараметров: Grid Search, Random Search.
- Кросс-валидация и разделение данных на тренировочные и тестовые выборки.
- Введение в ансамблевые методы: Bagging, Random Forest, Boosting.

2.3 Итоговый проект по второму модулю (4 часа)

- Постановка задачи: выбор задачи классификации или регрессии.
- Выбор и обучение модели.
- Подбор гиперпараметров и оценка качества модели.
- Презентация результатов и выводы.

Модуль 3: Машинное обучение для больших объёмов данных (24 часа)

3.1 Машинное обучение для тестов (6 часов)

- Введение в распределённое вычисление: Spark, Hadoop.
- Работа с большими данными: основы работы с HDFS, обработка данных с PySpark.
- Обучение моделей на больших данных: использование MLlib.
- Оптимизация производительности моделей при работе с большими данными.

3.2 Компьютерное зрение (14 часов)

- Основы компьютерного зрения: работа с изображениями.
- Введение в Convolutional Neural Networks (CNN).
- Применение CNN для задач классификации изображений.
- Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.

3.3 Обучение без учителя (4 занятия)

- Введение в кластеризацию: k-means, DBSCAN, hierarchical clustering.
- Уменьшение размерности: PCA, t-SNE, UMAP.
- Применение обучения без учителя для сегментации данных.
- Практическое применение кластеризации на реальных данных.

Модуль 4: Выпускной проект (24 часа)

4.1 Постановка задачи: выбор задачи на основе реального или предоставленного набора данных.

4.2 Выбор подходящей модели (или моделей).

4.3 Предобработка данных и создание необходимых признаков.

4.4 Обучение и оценка модели.

4.5 Презентация итоговых результатов: отчет и защита проекта.

V. МЕТОДИЧЕСКИЕ и ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Методические материалы

УМК программы состоит из трех компонентов:

1. Учебные и методические пособия для педагога и учащихся.
2. Система средств обучения.
3. Система средств контроля результативности обучения.

Список литературы

1. Григорий Добин Основы искусственного интеллекта в примерах на Python. Самоучитель 2-е изд., перераб. И доп. -СПб.: БВХ-Петербург, 2024 - 448с.: ил.Бродская М. Доверенный ИИ: начало пути / М. Бродская // Vis journal. - 2023. - № 3. - С. 16-20.
2. Джоши Прадик Искусственный Интеллект с примерами на Python.: Пер. с англ - Спб.: ООО "Диалектика", 2019 -448с. - Парал. Тит англЕвсеенко С. М. Этапы развития технологий искусственного интеллекта и уточнение терминологии / С. М. Евсеенко // Инновации. - 2021. - № 4. - С. 39-48.
3. Рашид Тарик Создаём нейронную сеть.: Пер. с англ. - СПб.: ООО "Альфа-книга", 2017. 272 с. : ил.

5.2. Оценочные материалы

Для отслеживания результативности образовательной деятельности по программе проводятся: входной, текущий, промежуточный и итоговый.

Входной контроль – оценка стартового уровня образовательных возможностей учащихся при поступлении в объединение, ранее не занимающихся по данной дополнительной общеобразовательной общеразвивающей программе. Входная диагностика проводится в сентябре с целью выявления первоначального уровня знаний и умений.

Входной контроль проводится в форме тестирования.

Текущий контроль – оценка уровня и качества освоения тем программы и личностных качеств учащихся осуществляется на занятиях в течение всего учебного года для отслеживания уровня освоения учебного материала программы и развития ИТ-компетенций. Текущий контроль осуществляется посредством наблюдения и беседы.

Промежуточный контроль – оценка уровня и качества освоения учащимися дополнительной общеобразовательной общеразвивающей программы по итогам изучения раздела.

Итоговый контроль – оценка и качество освоения учащимися дополнительной общеобразовательной общеразвивающей программы по завершению учебного года. Итоговый контроль проходит в форме защиты итоговых проектных работ.

Формы фиксации результатов:

- диагностическая карта определения результатов по программе «Искусственный интеллект. Начальный уровень» (Приложение №1);
- анкета для родителей «Отношение родительской общественности к качеству образовательных услуг и степень удовлетворенности образовательным процессом в объединении»;
- фотографии защиты проектов.

Утверждаю
 Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Календарно-тематический план

№ п/п	Тема занятия	Кол-во часов	Дата занятия	
			план	факт
1.	Инструкция по технике безопасности. Переменные, типы данных, операции, условные операторы, циклы.	1	1 неделя	
2.	Основные структуры данных: списки, кортежи, множества, словари. Функции. Классы	1	1 неделя	
3.	Импорт данных из различных источников: CSV, Excel, базы данных. Обработка пропущенных данных: методы заполнения и удаления.	1	2 неделя	
4.	Кодирование категориальных признаков: one-hot encoding, label encoding. Нормализация и стандартизация данных. Обнаружение и обработка выбросов.	1	2 неделя	
5.	Визуализация данных с использованием Matplotlib и Seaborn. Расчет основных статистических показателей: среднее, медиана, мода. Построение графиков распределения данных, коробчатых диаграмм и тепловых карт.	1	3 неделя	
6.	Взаимосвязь между признаками: корреляция, ковариация. Создание отчетов и дашбордов для презентации данных.	1	3 неделя	
7.	Введение в статистику: дисперсия, стандартное отклонение, вероятность. Гипотезы и тестирование гипотез: t-тест, chi-square тест.	1	4 неделя	
8.	Анализ взаимосвязей между признаками: регрессия, корреляционный анализ. Оценка и интерпретация статистических моделей. Применение статистических методов в реальных задачах.	1	4 неделя	
9.	Постановка задачи: выбор реального набора данных. Предобработка и анализ данных.	1	5 неделя	
10.	Визуализация и статистический анализ. Создание отчета и презентации результатов.	1	5 неделя	
11.	Введение в линейную регрессию: простая и множественная линейная регрессия. Методы регуляризации: Lasso, Ridge.	1	6 неделя	
12.	Логистическая регрессия: основы и применение. Оценка моделей: коэффициент детерминации (R^2), MSE, RMSE. Практическое применение линейных моделей на реальных данных.	1	6 неделя	
13.	Классификация: введение в алгоритмы классификации (Decision Trees, KNN, SVM).	1	7 неделя	
14.	Регрессия: полиномиальная регрессия, поддерживающие векторы.	1	7 неделя	
15.	Метрики качества моделей: accuracy, precision, recall, F1-score.	1	8 неделя	
16.	Подбор гиперпараметров: Grid Search, Random Search.	1	8 неделя	
17.	Кросс-валидация и разделение данных на тренировочные и тестовые выборки.	1	9 неделя	

18.	Кросс-валидация и разделение данных на тренировочные и тестовые выборки.	1	9 неделя	
19.	Введение в ансамблевые методы: Bagging, Random Forest, Boosting.	1	10 неделя	
20.	Введение в ансамблевые методы: Bagging, Random Forest, Boosting.	1	10 неделя	
21.	Итоговый проект по второму модулю. Постановка задачи: выбор задачи классификации или регрессии.	1	11 неделя	
22.	Выбор и обучение модели.	1	11 неделя	
23.	Подбор гиперпараметров и оценка качества модели.	1	12 неделя	
24.	Презентация результатов и выводы.	1	12 неделя	
25.	Введение в распределённое вычисление: Spark, Hadoop.	1	13 неделя	
26.	Введение в распределённое вычисление: Spark, Hadoop.	1	13 неделя	
27.	Работа с большими данными: основы работы с HDFS, обработка данных с PySpark.	1	14 неделя	
28.	Обучение моделей на больших данных: использование MLlib.	1	14 неделя	
29.	Оптимизация производительности моделей при работе с большими данными.	1	15 неделя	
30.	Оптимизация производительности моделей при работе с большими данными.	1	15 неделя	
31.	Основы компьютерного зрения: работа с изображениями.	1	16 неделя	
32.	Основы компьютерного зрения: работа с изображениями.	1	16 неделя	
33.	Введение в Convolutional Neural Networks (CNN).	1	17 неделя	
34.	Введение в Convolutional Neural Networks (CNN).	1	17 неделя	
35.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	18 неделя	
36.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	18 неделя	
37.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	19 неделя	
38.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	19 неделя	
39.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	20 неделя	
40.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	20 неделя	
41.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	21 неделя	
42.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	21 неделя	
43.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	22 неделя	
44.	Использование библиотек для работы с изображениями: OpenCV, TensorFlow, Keras.	1	22 неделя	
45.	Введение в кластеризацию: k-means, DBSCAN, hierarchical clustering.	1	23 неделя	
46.	Уменьшение размерности: PCA, t-SNE, UMAP.	1	23 неделя	
47.	Применение обучения без учителя для сегментации данных.	1	24 неделя	
48.	Практическое применение кластеризации на реальных данных.	1	24 неделя	

49.	Постановка задачи: выбор задачи на основе реального или предоставленного набора данных.	1	25 неделя	
50.	Постановка задачи: выбор задачи на основе реального или предоставленного набора данных.	1	25 неделя	
51.	Выбор подходящей модели (или моделей).	1	26 неделя	
52.	Выбор подходящей модели (или моделей).	1	26 неделя	
53.	Выбор подходящей модели (или моделей).	1	27 неделя	
54.	Выбор подходящей модели (или моделей).	1	27 неделя	
55.	Предобработка данных и создание необходимых признаков.	1	28 неделя	
56.	Предобработка данных и создание необходимых признаков.	1	28 неделя	
57.	Предобработка данных и создание необходимых признаков.	1	29 неделя	
58.	Предобработка данных и создание необходимых признаков.	1	29 неделя	
59.	Предобработка данных и создание необходимых признаков.	1	30 неделя	
60.	Предобработка данных и создание необходимых признаков.	1	30 неделя	
61.	Предобработка данных и создание необходимых признаков.	1	31 неделя	
62.	Предобработка данных и создание необходимых признаков.	1	31 неделя	
63.	Предобработка данных и создание необходимых признаков.	1	32 неделя	
64.	Обучение и оценка модели.	1	32 неделя	
65.	Обучение и оценка модели.	1	33 неделя	
66.	Обучение и оценка модели.	1	33 неделя	
67.	Обучение и оценка модели.	1	34 неделя	
68.	Обучение и оценка модели.	1	34 неделя	
69.	Обучение и оценка модели.	1	35 неделя	
70.	Обучение и оценка модели.	1	35 неделя	
71.	Презентация итоговых результатов: отчет и защита проекта.	1	36 неделя	
72.	Презентация итоговых результатов: отчет и защита проекта.	1	36 неделя	

Диагностическая карта определения результатов по программе

за учебный год _____

Группа: _____

Педагог: _____

Даты проведения: вход: _____

1 полугодие: _____

2 полугодие _____

№	Фамилия	Входной контроль	Промежуточный контроль	Итоговый контроль
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				
11.				
12.				

Уровень освоения обучающимися программы

Параметры результативности	Входной контроль (количество / %)	Промежуточный контроль (количество / %)	Итоговый контроль (количество / %)
Высокий уровень			
Средний уровень			
Начальный уровень			

Входной контроль
по программе дополнительного образования
«Введение в Data Science»

ФИО _____
Группа _____

Выполните тестовое задание, записав ответы

1. Какой тип данных в Python будет у переменной `a`, если выполнить следующий код?

`a = 5 / 2`

Ответ: _____

2. Какой результат будет после выполнения следующего выражения?

`5 // 2`

Ответ: _____

3. Какое ключевое слово используется для определения функции в Python?

Ответ: _____

4. Что произойдет, если попытаться изменить элемент в строке? Пример:

`s = "hello"`
`s[0] = "H"`

Ответ: _____

5. Чем отличаются списки (`list`) и кортежи (`tuple`) в Python?

Ответ: _____

6. Что делает следующая строка кода?

`numbers = [1, 2, 3]`
`print(numbers[::-1])`

Ответ: _____

7. Какая ошибка возникает, если делить число на ноль в Python?

Ответ: _____

8. Каков результат следующего кода?

```
a = [1, 2, 3]
b = a
b.append(4)
print(a)
```

Ответ: _____

9. Какой результат выполнения этого кода?

```
def func(a, b=[]):
    b.append(a)
    return b
print(func(1))
print(func(2))
```

Ответ: _____

10. Чем отличается оператор `is` от оператора `==` в Python?

Ответ: _____

11. Какой результат будет после выполнения следующего кода?

```
my_list = [1, 2, 3, 4, 5]
new_list = [x * 2 for x in my_list]
print(new_list)
```

Ответ: _____

12. Какое значение переменной `x` после выполнения следующего кода?

```
x = 10
def update(x):
    x += 5
    return x
update(x)
```

Ответ: _____

13. Что такое "генератор" (generator) в Python?

Ответ: _____

14. Какое ключевое слово используется для обработки исключений в Python?

Ответ: _____

15. Какой результат будет после выполнения следующего кода?

```
a = {1, 2, 3}
b = {3, 4, 5}
print(a & b)
```

Ответ: _____

16. Что означает ключевое слово `yield` в Python и где оно используется?

Ответ: _____

17. Как создать пустое множество (set) в Python?

Ответ: _____

18. Каков результат работы этого фрагмента кода?

```
a = [1, 2, 3]
print(a[3:])
```

Ответ: _____

19. Что делает декоратор в Python и как его применить к функции?

Ответ: _____

20. Какая ошибка возникнет при выполнении следующего кода?

```
print("Hello")
raise ValueError("Ошибка!")
```

Ответ: _____

Ответы к входному тесту

1. Тип данных: float.
2. Результат: 2.
3. Ключевое слово для функций: def.
4. Результат: Возникнет ошибка `TypeError`, так как строки неизменяемы.
5. Отличие списков и кортежей: Списки изменяемы, кортежи — нет.
6. Результат: Выводит список в обратном порядке: [3, 2, 1].
7. Ошибка при делении на ноль: `ZeroDivisionError`.
8. Результат: `[1, 2, 3, 4]` — так как `b` и `a` ссылаются на один и тот же объект.
9. Результат: `[1]` и `[1, 2]` — функция использует один и тот же список `b` по умолчанию.
10. Отличие `is` и `==`: `is` проверяет идентичность объектов (ссылаются ли они на одну область памяти), `==` проверяет равенство значений.
11. Результат: `[2, 4, 6, 8, 10]`.
12. Значение переменной `x`: `10` — так как внутри функции переменная `x` локальная.
13. Генератор: Это функция, которая возвращает объект-генератор с помощью `yield` и генерирует последовательность значений по запросу.
14. Ключевое слово для обработки исключений: `try-except`.
15. Результат: `{3}` — пересечение множеств.
16. Ключевое слово `yield`: Оно используется для создания генераторов и возвращает значение без завершения функции.
17. Пустое множество: `set()`.
18. Результат: `[]` — пустой список.
19. Декоратор: Это функция, которая изменяет или расширяет поведение другой функции, не изменяя её код. Применяется через `@decorator_name`.
20. Ошибка: `ValueError` будет вызвана с сообщением "Ошибка!".

Промежуточный контроль
по программе дополнительного образования
«Искусственный интеллект. Начальный уровень»

Задание: создайте творческий проект нейросети делающей предсказание по исходным данным.

Таблица 1.

№	Критерии	Баллы
1.	Составлено техническое задание для проекта	2 балла
2.	Присутствует грамотно составленное авторское описание проекта в формате .doc	3 балла
3.	Код программы написан в соответствии с конвенцией. Нейросеть обучена и точность прогноза на тестовом наборе данных не ниже 95%	5 баллов
4.	Креативность и оригинальность проекта	5 баллов
	Полнота раскрытия темы	5 баллов
		20 баллов

Предлагаемые наборы данных:

Kaggle Datasets

[UCI Machine Learning Repository](#)

Google Dataset Search

[data.gov.ru](#)

EU Open Data Portal

data.gov

различные волны лонгитюдных исследований ВШЭ

самостоятельно собранные данные

Критерии оценивания
промежуточного контроля
по программе дополнительного образования
«Искусственный интеллект. Начальный уровень»

По итогам выполнения группового проекта возможно набрать 20 баллов.

Высокий уровень освоения программы курса - 16 – 20 баллов.

Средний уровень освоения программы курса - 11 – 15 баллов.

Низкий уровень освоения программы курса – 10 и менее баллов.

Итоговый контроль
по программе дополнительного образования
«Искусственный интеллект. Начальный уровень»

Задание: создайте учебную генеративную нейросеть в соответствии с приведёнными ниже критериями (см. таблицу 1).

Таблица 1.

№	Критерии	Баллы
1.	Присутствует грамотно составленное авторское описание задач проекта в формате .doc	5 баллов
2.	Нейросеть справляется с выполнением поставленных задач	5 баллов
3.	Логически выстроенный сценарий событий	5 баллов
4.	Точность ответа на поставленные вопросы	5 баллов
5.	Креативность и оригинальность проекта	5 баллов
		25 баллов

Критерии оценивания
итогового контроля
по программе дополнительного образования
«Искусственный интеллект. Начальный уровень»

Высокий уровень освоения программы курса - 20 – 25 баллов.

Средний уровень освоения программы курса - 13 – 19 баллов.

Низкий уровень освоения программы курса – 12 и менее баллов.

