ЦЕНТР ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
МГТУ им. Н.Э. Баумана

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

Учебный план дополнительной профессиональной программы повышения квалификации:

«Системное программирование и работа с данными на языках Python и SQL с применением искусственного интеллекта»

№ п/п	Наименование темы	Описание темы	Всего, час	В том числе			
				Лекции	Практ. раб.	Сам. работа	Аттест.
Модуль 1 «Основы программирования и баз данных»							
1.1	Базовые понятия и определения	Прикладная задача – постановка и решение. Алгоритм – свойства и способы представления. Программа – подготовка и исполнение на компьютере. Данные – способы представления и обработки. Типы данных – назначение и роль в программе. Операнды и операторы – вычисление выражений. Основы программирования на Python. Интерактивный и пакетный режимы работы Python. Переменные и константы. Типы данных – изменяемые и неизменяемые. Работа с целыми числами и числами с плавающей точкой. Базовые операции работы со строками. Простые операции ввода-вывода. Преобразование типов. Разработка программ линейной структуры: ввод – обработка – вывод.	12	6	2	4	-
1.2	Разработка программ: модели и методологии	Методологии и языки программирования – обзор. Императивное и декларативное программирование.	10	4	4	2	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		Структурное и объектно-ориентированное. Функциональное и логическое. Структурное программирование. Базовые принципы: блочная структура кода – блоки и подпрограммы, типовые структуры управления – последовательность, ветвление, цикл, «нисходящая» разработка – пошаговая детализация программы. Структурное программирование на Python. Инструкции ветвления и цикла. Инструкции break, continue и pass. Функции. Структура программы. Наборы данных – кортежи, списки, словари, множества. Базовые принципы: абстрагирование, инкапсуляция, наследование, полиморфизм. Объектно-ориентированное программирование на Python. Объектная модель Python – встроенные типы как объекты. Объявление классов и создание объектов. Наследование классов. Полиморфизм и отделение интерфейса от реализации. Концепция MVC. Программирование типовых алгоритмов обработки наборов данных. Объектно-ориентированное программирование					
1.3	Работа с данными из внешних источников	Введение в Python, среды исполнения (IDE). Типы данных в Python. Базовые конструкции и структуры. Циклы и условия. Функции и классы. Библиотеки Numpy, Scipy для научных вычислений.	8	4	4	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		Библиотека Pandas, как стандарт исследования данных. Визуализация в Python и срезы данных. Библиотеки Matplotlib, seaborn, plotly. Работа с командной строкой. Linux, Windows. Решение практического кейса: «Визуализация данных с использованием основных библиотек языка программирования Python в среде Jupyter Notebook».					
1.4	Промежуточная аттестация	Тестирование	1	-	-	1	-
Модуль 2 «Программирование на языке Python. Уровень 1. Основы программирования»							
2.1	Язык программирования Python и его место среди других языков и систем программирования	Обзор существующих языков программирования. Основные принципы работы интерпретатора языка программирования Python. Язык программирования Python в различных операционных системах. Установка и запуск интерпретатора языка программирования Python. Установка и тестирование среды разработки языка программирования Python. Основные элементы программирования	5	3	1	1	-
2.2	Базовые конструкции языка программирования Python	Переменные и выражения. Знакомство и типами данных: числа, строки, списки, логический тип, None. Типы данных. Принцип динамической типизации. Оператор del. Структура программы. Блок Ветвления Базовая форма цикла Операторы break и continue. Оператор pass. Простой ввод и простой вывод.	4	3	1	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

2.3	Кортежи, списки, словари, множества	Кортеж. Основные операции с кортежем. Список. Основные операции со списком. Словарь. Основные операции со словарем. Цикл по итератору. Перебор (for)	7	3	2	2	-
2.4	Функции и коллекции	Создание функции. Вызов функции. Именованные и неименованные аргументы функций. Функции с переменным числом аргументов. Перебор (for). Функция как объект первого порядка. Понятие коллекции. Генераторное выражение. Генератор-функция. Генератор списков, генератор словарей, генератор множеств. Встроенные функции: map, zip, filter, reduce. Лябмда-функция.	5	3	2	-	-
2.5	Модули и пакеты	Создание модуля. Импорт пакета. Создание собственного пакета. Важнейшие стандартные пакеты. Подсистема pip. Установка стороннего модуля.	4	3	1	-	-
2.6	Работа с файловой системой	Работа с файлами и каталогами. Основные операции с файлами. Основные операции с путями к файлам. Рекурсивный обход каталога. Чтение файла. Запись в файл.	5	2	2	1	-
2.7	Исключения и обработка ошибок	Понятие об исключении. Выброс исключения. Перехват исключения. Стандартные исключения.	4	3	1	-	-
2.8	Регулярные выражения	Понятие о регулярном выражении. Синтаксис регулярных выражений. Применение регулярных выражений.	4	2	2	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

2.9	Элементы функционального программирования	Каррирование. Замыкание. Понятие о декораторе	6	3	2	1	-
2.10	Промежуточная аттестация	Тестирование	1	-	-	1	-
Модуль 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»							
3.1	Разработка программ с помощью ИИ (вайбкодинг) и новая культура программирования: от ИИ-чата к готовому решению	Понятие написания кода с помощью ИИ (вайбкодинга), его место в современной разработке и отличие от классического программирования. Обзор ИИ-чатов и отечественных инструментов, демонстрация перехода от идеи к первому варианту кода. Изучение материалов по базовым сценариям использования ИИ-чатов для разработки. Подготовка краткого описания задачи, которую можно автоматизировать.	4	2	-	2	-
3.2	От идеи к первому скрипту: создание полезной программы через ИИ-чат	Демонстрация создания простой программы через ИИ-чат: постановка задачи, генерация кода, запуск и первичная доработка. Разбор структуры скрипта, пользовательского ввода, вывода результата и сохранения данных. Подготовка варианта промпта для создания простого скрипта. Анализ полученного ИИ-кода по чек-листу: понятность, запуск, корректность результата, обработка типовых ошибок.	4	-	2	2	-
3.3	От ошибки к качеству: отладка, проверка и улучшение ИИ-сгенерированного кода	Разбор типовых ошибок ИИ-сгенерированного кода: неверные типы данных, отсутствующие файлы и ошибки структуры. Демонстрация цикла доработки.	4	-	2	2	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		Повторение типовых ошибок и способов их описания для ИИ-чата. Подготовка к тестированию по отладке кода, анализу трассировки ошибок и улучшению промптов.					
3.4	ИИ-чат как помощник программиста: автоматизация файлов, таблиц и отчётов	Демонстрация автоматизации работы с файлами и таблицами: чтение файлов, расчёт показателей, формирование текстового отчёта. Показано, как ИИ-чат помогает написать, объяснить и улучшить скрипт для обработки данных. Разбор примеров файловых задач: очистка данных, подсчёт итогов, группировка строк, сохранение результата. Подготовка практического задания по автоматизации небольшой таблицы.	4	-	2	2	-
3.5	Нейросеть внутри программы: подключение ГигаЧат	Переход от обычного нейросетевого чата к использованию нейросети внутри программы. Объяснение принципов программного интерфейса, безопасного хранения ключей доступа, отправки запроса в ГигаЧат/Яндекс GPT и получения ответа в заданной структуре. Изучение справочной документации по ГигаЧат и Яндекс GPT. Подготовка к тесту по различиям между ИИ-чатом, программного интерфейса и агентным сценарием.	4	2	-	2	-
3.6	Финальная сборка: мини ИИ-инструмент с элементами агентного подхода	Финальная демонстрация сборки мини ИИ-инструмента: постановка цели, генерация кода по частям, подключение обработки данных и	2	-	2	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		формирование результата. Рассматриваются элементы агентного подхода: цель, шаги, инструменты, проверка результата.					
3.7	Промежуточная аттестация	Тестирование	1	-	-	1	-
Модуль 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»							
4.1	Классы и объекты	Введение в Объектною-ориентированное программирование (ООП). Класс и экземпляр класса. Данные экземпляра, методы экземпляра и свойства экземпляра. Создание собственного класса. Инкапсуляция. Атрибуты класса. Разработка собственных классов. Определение нужных методов и свойств классов. Создание нескольких объектов.	6	4	1	1	-
4.2	Наследование	Роль наследования в ООП, понятие иерархии наследования. Принцип утиной типизации. Понятие базового класса и производного класса. Функция isinstance и ее применение. Создание производного класса. Расширение готовых классов.	5	3	1	1	-
4.3	Абстрактные классы и полиморфизм	Полиморфизм. Принцип DRY и WET. Понятие абстракции. Знакомство с абстрактными классами языка программирования Python. Подмена методов в производном классе. Полиморфные классы. Контейнерные типы. Библиотечные модули collections и collections.abc. Применение контейнерных типов. Имплементация новых	6	4	2	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		методов.					
4.4	Алгоритмы сортировки и поиска в языке программирования Python	Сложность алгоритмов, O(N)-нотация. Типы сортировки. Основные алгоритмы сортировки и поиска. Этапы разработки алгоритма. Разработка на языке программирования Python оптимальных алгоритмов поиска. Реализация на языке программирования Python алгоритма решателя Судоку. Решение ста сложнейших Судоку.	6	4	1	1	-
4.5	Алгоритмы поиска на графах	Введение в теорию графов, основные алгоритмы на графах. Теория графов. Представление графов в языке программирования Python. Поиск в ширину. Поиск в глубину. Имплементация графа в языке программирования Python. Нахождение кратчайшего пути проезда на примере графа станций московского метрополитена.	5	4	1	-	-
4.6	Записи и данных. Работа с данными системы SQLite	Разработка структуры данных. Сохранение данных. Список и запись. Записи и таблицы. Чтение и запись Понятие об объектно-реляционном соответствии. Хранение данных пользователей в СУБД SQLite.	5	3	1	1	-
4.7	Хранение данных вне программы	Понятие о структуре данных. Использование структуры для хранения данных. Хранение данных вне языка программирования Python. Сохранение и восстановление данных. Библиотечный модуль pickle и shelve. Сохранение данных графа	5	4	1	-	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		и использованием модуля pickle.					
4.8	Работа с данными JSON в языке программирования Python	Знакомство с форматом JSON. Работа с форматом JSON в языке программирования Python. Сериализация и десериализация файлов JSON в языке программирования Python. Хранение данных вне языка программирования Python. Составление графа московского метрополитена на основе файла JSON.	6	4	1	1	-
4.9	Промежуточная аттестация	Тестирование	1	-	-	1	-
Модуль 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»							
5.1	Введение. Постановка задачи	Анализ бизнес-требований, определение масштаба проблемы. Метрики оценки модели. Определение конечной цели проекта интеллектуального анализа данных. Элементы программирования. Переменные в языке SQL. Объявление переменных. Присвоение значения. Логические выражения и условные вычисления. Приоритет операторов.	4	2	-	2	-
5.2	Типы данных, функции, операторы и программные модули	Логические выражения и условные вычисления. Типы данных. Математические функции. Работа с календарём. Статистические функции. Работа с текстом. Представления, процедуры и пользовательские функции. Табличные выражения. Программирование и	11	1	6	4	-

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru

		динамические запросы. Работа с базой данных и представление данных. Использование функций различных видов.					
5.3	Структуры данных	Нормализованные базы данных. Хранилища и витрины данных. Объектные данные. Данные в формате XML. Работа с большими таблицами. Иерархии. Сети и графы. Работа со структурой баз данных. Работа с хранилищем и витриной данных (DWH, Data Marts). Эффективная работа с программой Management Studio (SSMS).	9	1	6	2	-
5.4	Методы анализа	Вычисления и поиск. Работа с упорядоченными сериями. Распределение. Сегментирование. Регрессия. Корреляция. Анализ связанных событий. Вероятностные модели. Другие методы анализа данных. Статистический анализ данных. Инструменты анализа данных. Использование функций на практике для построения отчетов.	9	1	6	2	-
5.5	Промежуточная аттестация	Тестирование	1	-	-	1	-
6	Итоговая аттестация	Зачет	2	-	-	-	2
	ИТОГО		**180**	**78**	**59**	**41**	**2**

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)



УТВЕРЖДАЮ
Проректор по учебной работе
МГТУ им. Н.Э. Баумана
_____ С.В. Альков
«24» 04 2026 г.

Дополнительное профессиональное образование

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

«Системное программирование и работа с данными на языках Python и SQL с применением искусственного интеллекта»

Регистрац. №06.05-11/569

Москва, 2026

Оглавление

1. ОБЩАЯ ХАРАКТЕРИСТИКА ДПП 3

1.1. Цель ДПП 3

1.2. Планируемые результаты обучения 3

1.3. Дополнительные характеристики ДПП 3

1.4. Перечень профессиональных компетенций в рамках имеющейся квалификации, качественное изменение которых осуществляется в результате обучения 4

1.5. Соответствие видов деятельности профессиональным компетенциям и их составляющих 4

2. УЧЕБНЫЙ ПЛАН ДПП 5

2.1. Категория слушателей ДПП 5

2.2. Общая трудоёмкость программы, аудиторная и самостоятельная работа 5

2.3. Форма обучения 5

2.4. Учебный план 5

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК 8

4. РАБОЧАЯ ПРОГРАММА ДПП 10

5. УСЛОВИЯ РЕАЛИЗАЦИИ ДПП 15

5.1. Организационные условия реализации ДПП 48

5.2. Педагогические условия реализации ДПП 48

5.3. Учебно-методическое обеспечение ДПП 48

5.4. Методические рекомендации 51

6. ФОРМЫ ИТОГОВОЙ АТТЕСТАЦИИ ДПП 53

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ИТОГОВОЙ АТТЕСТАЦИИ 54

7.1. Паспорт комплекта оценочных средств 54

7.2. Комплект оценочных средств 54

№ 06.05 – 11/569

1. ОБЩАЯ ХАРАКТЕРИСТИКА ДПП

Программа подготовлена на основе:

- Федерального закона от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказа Минобрнауки России от 24 марта 2025 г. N 266 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- Письма Минобрнауки РФ от 22.04.2015 N ВК-1032/06 «О направлении методических рекомендаций».

Реализация программы ДПП направлена на получение новой компетенции, необходимой для профессиональной деятельности.

1.1. Цель ДПП

Сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

1.2. Планируемые результаты обучения

Планируемые результаты обучения по ДПП:

- освоение профессиональных компетенций в процессе изучения перечисленных тем в учебном плане;
- успешное освоение программы повышения квалификации;
- успешное прохождение итоговой аттестации (зачет).

Обучающимся, успешно прошедшим обучение, выполнившим текущие контрольные задания и выдержавшим предусмотренный учебным планом зачет, выдается удостоверение о повышении квалификации по ДПП «Системное программирование и работа с данными на языках Python и SQL с применением искусственного интеллекта».

1.3. Дополнительные характеристики ДПП

Характеристики новой квалификации определены в приказе Минтруда России от 20 июля 2022 года N 424н «Об утверждении профессионального стандарта «Программист» (06.001).

Вид профессиональной деятельности:

- Разработка компьютерного программного обеспечения (Код 06.001).

Трудовые функции:

- Проектирование компьютерного программного обеспечения (D/03.6).

1.4. Перечень профессиональных компетенций в рамках имеющейся квалификации, качественное изменение которых осуществляется в результате обучения

Профессиональные компетенции базируются на основании Приказа Минобрнауки России от 19 сентября 2017 г. № 929 «Об утверждении федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника».

Перечень компетенций:

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения.

1.5. Соответствие видов деятельности профессиональным компетенциям и их составляющих

Профессиональные компетенции	Практический опыт	Умения	Знания
Проектирование компьютерного программного обеспечения (D/03.6)			
ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Проектирование структур данных	Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов	Методы и средства проектирования компьютерного программного обеспечения

2. УЧЕБНЫЙ ПЛАН ДПП

2.1. Категория слушателей ДПП

Имеющаяся квалификация (требования к слушателям) – к освоению ДПП допускаются лица, имеющие высшее образование.

2.2. Общая трудоёмкость программы, аудиторная и самостоятельная работа

Общая трудоемкость программы 180 академических часов, из них 137 академических часов аудиторной работы, 41 академический час самостоятельной работы и 2 часа итоговой аттестации.

2.3. Форма обучения

Форма обучения по ДПП – очная, с применением дистанционных образовательных технологий.

2.4. Учебный план

ДПП «Системное программирование и работа с данными на языках Python и SQL с применением искусственного интеллекта» реализуется пятью тематическими модулями.

№ п/п	Наименование темы, модуля	Форма контроля	Всего, час	В том числе			
				Лекции	Практ. занятия	Самост. работа	Итоговая аттестация
Модуль 1 «Основы программирования и баз данных»							
1.1	Базовые понятия и определения	Устный опрос	12	6	2	4	-
1.2	Разработка программ: модели и методологии	Устный опрос	10	4	4	2	-
1.3	Работа с данными из внешних источников	-	8	4	4	-	-
1.4	Промежуточная аттестация	Тест	1	-	-	1	-
Модуль 2 «Программирование на языке Python. Уровень 1. Основы программирования»							
2.1	Язык программирования Python и его место среди других языков и систем программирования	Устный опрос	5	3	1	1	-
2.2	Базовые конструкции языка программирования Python	Устный опрос	4	3	1	-	-
2.3	Кортежи, списки, словари, множества	Устный опрос	7	3	2	2	-
2.4	Функции и коллекции	Устный опрос	5	3	2	-	-
2.5	Модули и пакеты	Устный опрос	4	3	1	-	-

2.6	Работа с файловой системой	Устный опрос	5	2	2	1	-
2.7	Исключения и обработка ошибок	Устный опрос	4	3	1	-	-
2.8	Регулярные выражения	Устный опрос	4	2	2	-	-
2.9	Элементы функционального программирования	Устный опрос	6	3	2	1	-
2.10	Промежуточная аттестация	Устный опрос	1	-	-	1	-
Модуль 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»							
3.1	Разработка программ с помощью ИИ (вайбкодинг) и новая культура программирования: от ИИ-чата к готовому решению	Устный опрос	4	2	-	2	-
3.2	От идеи к первому скрипту: создание полезной программы через ИИ-чат	Практич. задание	4	-	2	2	-
3.3	От ошибки к качеству: отладка, проверка и улучшение ИИ-сгенерированного кода	Устный опрос	4	-	2	2	-
3.4	ИИ-чат как помощник программиста: автоматизация файлов, таблиц и отчётов	Практич. задание	4	-	2	2	-
3.5	Нейросеть внутри программы: подключение ГигаЧат	Устный опрос	4	2	-	2	-
3.6	Финальная сборка: мини ИИ-инструмент с элементами агентного подхода	-	2	-	2	-	-
3.7	Промежуточная аттестация	Устный опрос	1	-	-	1	-
Модуль 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»							
4.1	Классы и объекты	Устный опрос	6	4	1	1	-
4.2	Наследование	Устный опрос	5	3	1	1	-
4.3	Абстрактные классы и полиморфизм	Устный опрос	6	4	2	-	-
4.4	Алгоритмы сортировки и поиска в языке программирования Python	Устный опрос	6	4	1	1	-

4.5	Алгоритмы поиска на графах	Устный опрос	5	4	1	-	-
4.6	Записи данных. Работа с данными системы SQLite	Устный опрос	5	3	1	1	-
4.7	Хранение данных вне программы	Устный опрос	5	4	1	-	-
4.8	Работа с данными JSON в языке программирования Python	Устный опрос	6	4	1	1	-
4.9	Промежуточная аттестация	Устный опрос	1	-	-	1	-
Модуль 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»							
5.1	Введение. Постановка задачи	Устный опрос	4	2	-	2	-
5.2	Типы данных, функции, операторы и программные модули	Устный опрос	11	1	6	4	-
5.3	Структуры данных	Устный опрос	9	1	6	2	-
5.4	Методы анализа	Устный опрос	9	1	6	2	-
5.5	Промежуточная аттестация	Тест	1	-	-	1	-
6	Итоговая аттестация	Зачет	2	-	-	-	2
	ИТОГО		180	78	59	41	2

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

№ п/п	Наименование темы, модуля	1 день	2 день	3 день	4 день	5 день	6 день	7 день
1	Модуль 1 «Основы программирования и баз данных»							
2	Модуль 2 «Программирование на языке Python. Уровень 1. Основы программирования»							
3	Модуль 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»							
4	Модуль 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»							
5	Модуль 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»							
6	Итоговая аттестация							
№ п/п	Наименование темы, модуля	8 день	9 день	10 день	11 день	12 день	13 день	14 день
1	Модуль 1 «Основы программирования и баз данных»							
2	Модуль 2 «Программирование на языке Python. Уровень 1. Основы программирования»							
3	Модуль 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»							
4	Модуль 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»							
5	Модуль 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»							
6	Итоговая аттестация							

№ п/п	Наименование темы, модуля	15 день	16 день	17 день	18 день	19 день	20 день	21 день	22 день	23 день
1	Модуль 1 «Основы программирования и баз данных»									
2	Модуль 2 «Программирование на языке Python. Уровень 1. Основы программирования»									
3	Модуль 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»									
4	Модуль 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»									
5	Модуль 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»									
6	Итоговая аттестация									Зачет

Минимальный срок освоения ДПП – 23 дня.

4. РАБОЧАЯ ПРОГРАММА ДПП

4.1. Рабочая программа модуля 1 «Основы программирования и баз данных»

4.1.1. Цель изучения модуля: сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

4.1.2. Задачи изучения модуля: сформировать у слушателей базовые умения разработки алгоритмов и программ на языке Python, обработки данных из внешних источников, а также проектирования структур данных и баз данных для решения прикладных задач.

4.1.3. Планируемые результаты обучения

Процесс изучения раздела направлен на формирование следующих компетенций

Код компетенции	Перечень планируемых результатов обучения по модулю	Формы и методы обучения, способствующие формированию и развитию компетенции
ОПК-8	**Знать:** Методы и средства проектирования компьютерного программного обеспечения. **Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов. **Владеть:** Проектированием структур данных.	Формы обучения: Фронтальная. Методы обучения: лекции; практические занятия; самостоятельная работа.

4.1.4 Содержание курса

Тема 1.1. Базовые понятия и определения (12 часов)

Лекции (6 часов). Прикладная задача – постановка и решение. Алгоритм – свойства и способы представления. Программа – подготовка и исполнение на компьютере. Данные – способы представления и обработки. Типы данных – назначение и роль в программе. Операнды и операторы – вычисление выражений. Основы программирования на Python. Интерактивный и пакетный режимы работы Python. Переменные и константы. Типы данных – изменяемые и неизменяемые. Работа с целыми числами и числами с плавающей точкой. Базовые операции работы со строками. Простые операции ввода-вывода. Преобразование типов.

Практические занятия (2 часа). Разработка программ линейной структуры: ввод – обработка – вывод.

Самостоятельная работа (4 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Базовые понятия и определения	Программы линейной структуры	Проработка дополнительной литературы	Кумскова, И.А., Базы данных: учебник / И.А. Кумскова. – Москва: КноРус, 2022. – 400 с. – ISBN 978-5-406-09667-3	устный опрос

Тема 1.2. Разработка программ: модели и методологии (10 часов)

Лекции (4 часа). Методологии и языки программирования – обзор. Императивное и декларативное программирование. Структурное и объектно-ориентированное. Функциональное и логическое. Структурное программирование. Базовые принципы: блочная структура кода – блоки и подпрограммы, типовые структуры управления – последовательность, ветвление, цикл, «нисходящая» разработка – пошаговая детализация программы. Структурное программирование на Python. Инструкции ветвления и цикла. Инструкции break, continue и pass. Функции. Структура программы. Наборы данных – кортежи, списки, словари, множества. Базовые принципы: абстрагирование, инкапсуляция, наследование, полиморфизм. Объектно-ориентированное программирование на Python. Объектная модель Python – встроенные типы как объекты. Объявление классов и создание объектов. Наследование классов. Полиморфизм и отделение интерфейса от реализации. Концепция MVC.

Практические занятия (4 часа). Программирование типовых алгоритмов обработки наборов данных. Объектно-ориентированное программирование.

Самостоятельная работа (2 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Разработка программ: модели и методологии	Обработка информации, язык SQL	Проработка дополнительной литературы	Кумскова, И.А., Базы данных: учебник / И.А. Кумскова. – Москва: КноРус, 2022. – 400 с. – ISBN 978-5-406-09667-3	устный опрос

Тема 1.3. Работа с данными из внешних источников (8 часов)

Лекции (4 часа). Введение в Python, среды исполнения (IDE). Типы данных в Python. Базовые конструкции и структуры. Циклы и условия. Функции и классы. Библиотеки Numpy, Scipy для научных вычислений. Библиотека Pandas, как стандарт исследования данных. Визуализация в Python и срезы данных. Библиотеки Matplotlib, seaborn, plotly. Работа с командной строкой. Linux, Windows.

Практические занятия (4 часа). Решение практического кейса: «Визуализация данных с использованием основных библиотек языка программирования Python в среде Jupyter Notebook».

Тема 1.4. Промежуточная аттестация (1 час)

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Промежуточная аттестация	Основы программирования и баз данных	Проработка дополнительной литературы	Кумскова, И.А., Базы данных: учебник / И.А. Кумскова. – Москва: КноРус, 2022. – 400 с. – ISBN 978-5-406-09667-3	Тест

4.1.5. Оценочное средство для текущего контроля (примерные вопросы для устного опроса):

Тема 1.1.

1. Перечислите известные вам источники данных.
2. Какие типы данных вам известны?
3. Приведите структуру данных.

Тема 1.2

1. В чем заключается логика императивного программирования?
2. Приведите структуру программы.
3. Что такое полиморфизм?

Тема 1.4

Вопрос 1. Что представляют собой базы данных?

a) аппаратные средства для хранения данных

b) набор логически связанных данных

c) программные средства управления данными

d) компьютер с хранящимися на нем данными

Вопрос 2. Какие из приведенных стилей программирования поддерживает язык Python?

a) Процедурный

b) Объектно-ориентированный

c) Функциональный

d) Смешанный

Вопрос 3. Каким способом можно объявлять переменные в Python:

a) a=5

b) a=int (5)

c) int a=5

Вопрос 4. Какая функция отвечает за вывод на экран?

a) cout<<a

b) out (a)

c) print (a)

Вопрос 5. Какая функция отвечает за открытие файла?

a) file()

b) open()

c) open_file()

Вопрос 6. В каком из вариантов присутствует ошибка?

a) a=5

print ('a')

b) while True

print(a)

c) a=open("file.txt")

Вопрос 7. Что делает команда import

a) импортирует файл модуля

b) создает функцию

c) удаляет файл

Вопрос 8. Выберите вариант правильного удаления переменной a

a) del(a)

b) delete(a)

c) delete=a

Вопрос 9. Какое значение 1//2 вернет выражение в среде IDLE?

a) 0

b) 0.5

c) 0.50

Вопрос 10. Как называется встроенный в языке Python тип данных неупорядоченной коллекции из нуля или более пар ключ-значение?

a) dict

b) set

c) list

d) frozenset

4.2. Рабочая программа модуля 2 «Программирование на языке Python. Уровень 1. Основы программирования»

4.2.1. Цель изучения модуля: сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

4.2.2. Задачи изучения модуля:

- освоить базовые конструкции языка Python: типы данных, ветвление, циклы, ввод-вывод;
- научиться работать с кортежами, списками, словарями и множествами;
- сформировать навыки создания и вызова функций, использования генераторов и встроенных функций map, filter, zip, reduce;
- освоить работу с модулями, пакетами и установку сторонних библиотек через pip;
- сформировать навыки работы с файлами, каталогами и путями;
- освоить обработку исключений и применение регулярных выражений;
- изучить элементы функционального программирования: замыкания, каррирование, декораторы.

4.2.3. Планируемые результаты обучения

Процесс изучения раздела направлен на формирование следующих компетенций

Код компетенции	Перечень планируемых результатов обучения по модулю	Формы и методы обучения, способствующие формированию и развитию компетенции
ОПК-8	**Знать:** Методы и средства проектирования компьютерного программного обеспечения. **Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов. **Владеть:** Проектированием структур данных.	Формы обучения: Фронтальная. Методы обучения: лекции; практические занятия; самостоятельная работа.

4.2.4 Содержание курса

Тема 2.1. Язык программирования Python и его место среди других языков и систем программирования (5 часов)

Лекции (3 часа) Обзор существующих языков программирования. Основные принципы работы интерпретатора языка программирования Python. Язык программирования Python в различных операционных системах.

Практические занятия (1 час). Установка и запуск интерпретатора языка программирования Python. Установка и тестирование среды разработки языка программирования Python. Основные элементы программирования

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Язык программирования Python и его место среди других языков и систем программирования	Различные версии языка программирования Python	Проработка дополнительной литературы	Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	устный опрос

Тема 2.2. Базовые конструкции языка программирования Python (4 часа)

Лекции (3 часа). Переменные и выражения. Знакомство и типами данных: числа, строки, списки, логический тип, None. Типы данных. Принцип динамической типизации. Оператор del. Структура программы. Блок Ветвления Базовая форма цикла Операторы break и continue.

Практические занятия (1 час). Оператор pass. Простой ввод и простой вывод.

Тема 2.3. Кортежи, списки, словари, множества (7 часов)

Лекции (3 часа). Кортеж. Основные операции с кортежем. Список. Основные операции со списком. Словарь. Основные операции со словарем.

Практические занятия (2 часа). Цикл по итератору. Перебор (for).

Самостоятельная работа (2 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Кортежи, списки, словари, множества.	Множества. Основные операции с множеством. Срезы.	Проработка дополнительной литературы	Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	устный опрос

Тема 2.4. Функции и коллекции (5 часов)

Лекции (3 часа). Создание функции. Вызов функции. Именованные и неименованные аргументы функций. Функции с переменным числом аргументов. Перебор (for). Функция как объект первого порядка.

Практические занятия (2 часа). Понятие коллекции. Генераторное выражение. Генератор-функция. Генератор списков, генератор словарей, генератор множеств. Встроенные функции: map, zip, filter, reduce. Лямбда-функция.

Тема 2.5. Модули и пакеты (4 часа)

Лекции (3 часа). Создание модуля. Импорт пакета. Создание собственного пакета. Важнейшие стандартные пакеты

Практические занятия (1 час). Подсистема pip. Установка стороннего модуля.

Тема 2.6. Работа с файловой системой (5 часов)

Лекции (2 часа). Работа с файлами и каталогами. Основные операции с файлами. Основные операции с путями к файлам. Рекурсивный обход каталога.

Практические занятия (2 часа). Чтение файла. Запись в файл.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Работа с файловой системой	Обработка параметров командной строки	Проработка дополнительной литературы	Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	устный опрос

Тема 2.7. Исключения и обработка ошибок (4 часа)

Лекции (3 часа). Понятие об исключении. Выброс исключения. Перехват исключения.

Практические занятия (1 час). Стандартные исключения.

Тема 2.8. Регулярные выражения (4 часа)

Лекции (2 часа). Понятие о регулярном выражении. Синтаксис регулярных выражений.

Практические занятия (2 часа). Применение регулярных выражений

Тема 2.9. Элементы функционального программирования (6 часов)

Лекции (3 часа). Каррирование. Замыкание.

Практические занятия (2 часа). Понятие о декораторе.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Элементы функционального программирования	Применение регулярных выражений	Проработка дополнительной литературы	Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	устный опрос

Тема 2.10. Промежуточная аттестация (1 час)

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Промежуточная аттестация	Основы программирования на языке Python	Проработка дополнительной литературы	Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк :	устный опрос

			Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	

4.2.5. Оценочное средство для текущего контроля:

Тема 2.1. Язык программирования Python и его место среди других языков и систем программирования

1. Изменяемые и неизменяемые типы данных
2. Хеширование
3. Виды строк
4. Лямбда-выражения

Тема 2.2. Базовые конструкции языка программирования Python

1. Списки
2. Итераторы и генераторы

Тема 2.3. Кортежи, списки, словари, множества

Задача:

```
A0 = dict(zip(('a','b','c','d','e'),(1,2,3,4,5)))
A1 = range(10)
A2 = sorted([i for i in A1 if i in A0])
A3 = sorted([A0[s] for s in A0])
A4 = [i for i in A1 if i in A3]
A5 = {i:i*i for i in A1}
A6 = [[i,i*i] for i in A1]
```

Определите, что находится в каждой переменной, и сравните свои предположения с ответом.

```
A0 = {'a': 1, 'c': 3, 'b': 2, 'e': 5, 'd': 4}
A1 = range(0, 10)
A2 = []
A3 = [1, 2, 3, 4, 5]
A4 = [1, 2, 3, 4, 5]
A5 = {0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81}
A6 = [[0, 0], [1, 1], [2, 4], [3, 9], [4, 16], [5, 25], [6, 36], [7, 49], [8, 64], [9, 81]]
```

Тема 2.4. Функции и коллекции

1. Создание функции
2. Вызов функции
3. Именованные и неименованные аргументы функций
4. Функции с переменным числом аргументов
5. Перебор (for)
6. Функция как объект первого порядка

Тема 2.5. Модули и пакеты

1. Создание модуля
2. Импорт пакета
3. Создание собственного пакета
4. Важнейшие стандартные пакеты
5. Подсистема pip
6. Установка стороннего модуля

Тема 2.6. Работа с файловой системой

1. Работа с файлами и каталогами
2. Основные операции с файлами.
3. Основные операции с путями к файлам.
4. Рекурсивный обход каталога.
5. Обработка параметров командной строки

Тема 2.7. Исключения и обработка ошибок

1. Понятие об исключении
2. Выброс исключения
3. Перехват исключения
4. Стандартные исключения

Тема 2.8. Регулярные выражения

1. Понятие о регулярном выражении
2. Синтаксис регулярных выражений
3. Применение регулярных выражений

Тема 2.9. Элементы функционального программирования

1. Каррирование
2. Замыкание
3. Понятие о декораторе

Тема 2.10. Промежуточная аттестация

Билет №1

1. Принцип динамической типизации.
2. Основные операции с кортежем.
3. Приведите пример кода на использование перебора (for).
4. Приведите пример кода с использованием функции filter.
5. Словарь. Основные операции со словарем.

Билет №2

1. Именованные и неименованные аргументы функций.
2. Создание функции (привести пример кода).
3. Операторы break и continue.
4. Приведите пример кода на использование перебора (for).
5. Понятие коллекции.

Билет №3

1. Множества. Основные операции с множеством.
2. Именованные и неименованные аргументы функций.
3. Приведите пример кода на использование лямбда-функции.
4. Приведите пример кода на использование перебора (for).
5. Типы данных: числа, строки, списки, логический тип, None.

4.3. Рабочая программа модуля 3 «Разработка программ с помощью ИИ (вайбкодинг) на языке программирования Python: от идеи к готовому инструменту с элементами агентного подхода»

4.3.1. Цель изучения модуля: сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

4.3.2. Задачи изучения модуля:

1. Формирование представления о разработке программ с помощью ИИ (вайбкодинг) как современном подходе к разработке программ с применением генеративных нейросетей.
2. Изучение принципов эффективной работы с ИИ-чатами и ИИ-ассистентами для генерации, отладки и улучшения программного кода.
3. Освоение базовых приёмов создания прикладных скриптов и ИИ-инструментов для автоматизации работы с файлами, таблицами, текстами и отчётами.

4.3.3. Планируемые результаты обучения

Процесс изучения раздела направлен на формирование следующих компетенций

Код компетенции	Перечень планируемых результатов обучения по модулю	Формы и методы обучения, способствующие формированию и развитию компетенции
ОПК-8	**Знать:** Методы и средства проектирования компьютерного программного обеспечения. **Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов. **Владеть:** Проектированием структур данных.	Формы обучения: Фронтальная. Методы обучения: лекции; практические занятия; самостоятельная работа.

4.3.4. Содержание курса

Тема 3.1. Разработка программ с помощью ИИ (вайбкодинг) и новая культура программирования: от ИИ-чата к готовому решению (4 часа)

Лекции (2 часа). Понятие написания кода с помощью ИИ (вайбкодинга), его место в современной разработке и отличие от классического программирования. Обзор ИИ-чатов и отечественных инструментов, демонстрация перехода от идеи к первому варианту кода.

Самостоятельная работа (2 часа). Изучение материалов по базовым сценариям использования ИИ-чатов для разработки. Подготовка краткого описания задачи, которую можно автоматизировать.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Разработка программ с помощью ИИ (вайбкодинг) и новая культура программирования: от ИИ-чата к готовому решению	Что такое вайбкодинг? Определение и суть подхода	Проработка дополнительной литературы	Кобелев Н. Б. Искусственный интеллект и имитация функций человека : учебное пособие / Кобелев Н. Б. - М. : КУРС, 2026. - 111 с. : табл., граф. - Библиогр.: с. 109-110. - ISBN 978-5-907352-11-7	Устный опрос

Тема 3.2. От идеи к первому скрипту: создание полезной программы через ИИ-чат (4 часа)

Практические занятия (2 часа). Демонстрация создания простой программы через ИИ-чат: постановка задачи, генерация кода, запуск и первичная доработка. Разбор структуры скрипта, пользовательского ввода, вывода результата и сохранения данных.

Самостоятельная работа (2 часа). Подготовка варианта промпта для создания простого скрипта. Анализ полученного ИИ-кода по чек-листу: понятность, запуск, корректность результата, обработка типовых ошибок.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
От идеи к первому скрипту: создание полезной программы через ИИ-чат	Как правильно сформулировать задачу для ИИ-чата: что должно быть в промпте?	Проработка дополнительной литературы	Эртель В. Введение в искусственный интеллект / Эртель В. ; пер. с англ. Горман А. - М. : Эксмо, 2019. - 443 с. : рис. - (Библиотека Сбербанка. Искусственный интеллект). - Библиогр.: с. 419-436. - ISBN 978-5-04-108594-0	Практ. задание

Тема 3.3. От ошибки к качеству: отладка, проверка и улучшение ИИ-сгенерированного кода (4 часа)

Практические занятия (2 часа). Разбор типовых ошибок ИИ-сгенерированного кода: неверные типы данных, отсутствующие файлы и ошибки структуры. Демонстрация цикла доработки.

Самостоятельная работа (2 часа). Повторение типовых ошибок и способов их описания для ИИ-чата. Подготовка к тестированию по отладке кода, анализу трассировки ошибок и улучшению промптов.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
От ошибки к качеству: отладка, проверка и улучшение ИИ-сгенерированного кода	Цикл доработки кода с ИИ-чатом	Проработка дополнительной литературы	Искусственный интеллект и нейросетевое управление : учебное пособие / сост. Мамонова Т. Е. - Томский политехнический университет, 2020. - ISBN 978-5-4387-0921-3	Устный опрос

Тема 3.4. ИИ-чат как помощник программиста: автоматизация файлов, таблиц и отчётов (4 часа)

Практические занятия (2 часа). Демонстрация автоматизации работы с файлами и таблицами: чтение файлов, расчёт показателей, формирование текстового отчёта. Показано, как ИИ-чат помогает написать, объяснить и улучшить скрипт для обработки данных.

Самостоятельная работа (2 часа). Разбор примеров файловых задач: очистка данных, подсчёт итогов, группировка строк, сохранение результата. Подготовка практического задания по автоматизации небольшой таблицы.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
ИИ-чат как помощник программиста: автоматизация файлов, таблиц и отчётов	Роль ИИ-чата в автоматизации	Проработка дополнительной литературы	Эртель В. Введение в искусственный интеллект / Эртель В. ; пер. с англ. Горман А. - М. : Эксмо, 2019. - 443 с. : рис. - (Библиотека Сбербанка. Искусственный интеллект). - Библиогр.: с. 419-436. - ISBN 978-5-04-108594-0	Практ. задание

Тема 3.5. Нейросеть внутри программы: подключение ГигаЧат (4 часа)

Лекции (2 час). Переход от обычного нейросетевого чата к использованию нейросети внутри программы. Объяснение принципов программного интерфейса, безопасного хранения ключей доступа, отправки запроса в ГигаЧат/Яндекс GPT и получения ответа в заданной структуре.

Самостоятельная работа (2 часа). Изучение справочной документации по ГигаЧат и Яндекс GPT. Подготовка к тесту по различиям между ИИ-чатом, программным интерфейсом и агентным сценарием.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Нейросеть внутри программы: подключение ГигаЧат	Справочная документация по ГигаЧат и Яндекс GPT	Проработка дополнительной литературы	Кобелев Н. Б. Искусственный интеллект и имитация функций человека : учебное пособие / Кобелев Н. Б. - М. : КУРС, 2026. - 111 с. : табл., граф. - Библиогр.: с. 109-110. - ISBN 978-5-907352-11-7	Устный опрос

Тема 3.6. Финальная сборка: мини ИИ-инструмент с элементами агентного подхода (2 часа)

Практические занятия (2 часа). Финальная демонстрация сборки мини ИИ-инструмента: постановка цели, генерация кода по частям, подключение обработки данных

и формирование результата. Рассматриваются элементы агентного подхода: цель, шаги, инструменты, проверка результата.

Тема 3.7. Промежуточная аттестация (1 час)

Самостоятельная работа (1 час). Проработка материалов дополнительной литературы.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Промежуточная аттестация	Агентный подход в вайбкодинге	Проработка дополнительной литературы	Кобелев Н. Б. Искусственный интеллект и имитация функций человека : учебное пособие / Кобелев Н. Б. - М. : КУРС, 2026. - 111 с. : табл., граф. - Библиогр.: с. 109-110. - ISBN 978-5-907352-11-7	Устный опрос

4.3.5. Оценочное средство для текущего контроля:

Тема 3.1. Примерная формулировка устного вопроса:

1. Что такое вайбкодинг и чем он отличается от классической разработки?
2. Какие задачи можно решать с помощью ИИ-чата?
3. Почему ИИ-сгенерированный код необходимо проверять?
4. Чем отличаются ИИ-чат, программные интерфейсы и агентный сценарий?
5. Какие данные нельзя отправлять в нейросеть?

Тема 3.2. Примерная формулировка практического задания:

Сформулируйте промпт для создания простого скрипта и объясните, какие входные данные, результат и ограничения должны быть указаны в запросе.

Тема 3.3. Примерная формулировка устного вопроса:

1. Что показывает последняя строка трассировки ошибок?
2. Какая ошибка возникает при обращении к несуществующей переменной?
3. Что нужно передать ИИ-чату для исправления ошибки?
4. Почему запрос «код не работает» считается недостаточным?
5. Что означает улучшение структуры кода?

Тема 3.4. Примерная формулировка практического задания:

Опишите алгоритм скрипта для обработки табличных файлов: чтение данных, расчёт показателей, формирование и сохранение отчёта.

Тема 3.5. Примерная формулировка устного вопроса:

1. Что такое программный интерфейс?
2. Чем отличается ручная работа в ИИ-чате от подключения нейросети к программе?
3. Почему ключ программного интерфейса нельзя хранить в открытом коде?
4. Для чего нужен структурированный ответ модели?
5. Какие задачи можно решать с помощью ГигаЧат и Яндекс GPT в программном приложении?

Тема 3.7. Примерная формулировка устного вопроса:

1. Объяснение порядка перехода от идеи к рабочему программному сценарию на языке Питон с использованием нейросетевого чата.
2. Анализ примера трассировки ошибки и определение места сбоя в программном коде.
3. Разработка текстового запроса для автоматизации обработки таблицы заказов.
4. Описание безопасного сценария подключения нейросетевой модели к программе на языке Питон через программный интерфейс.
5. Представление структуры мини-инструмента на основе искусственного интеллекта и критериев проверки полученного результата.
6. Описание процесса уточнения и декомпозиции задачи при работе с ИИ-чатом для получения рабочего кода.
7. Разработка промпта для генерации скрипта, обрабатывающего несколько текстовых файлов и объединяющего их в единый отчёт.
8. Демонстрация применения агентного подхода: распределение подзадач между несколькими ИИ-агентами в рамках одного проекта.
9. Составление чек-листа для проверки кода, сгенерированного нейросетью, на соответствие техническому заданию.
10. Описание сценария использования ИИ-ассистента для рефакторинга и оптимизации уже написанного кода на Python.

4.4. Рабочая программа модуля 4 «Программирование на языке Python. Уровень 2. Алгоритмы, работа с данными»

4.4.1. Цель изучения модуля: сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

4.4.2. Задачи изучения модуля:

- освоить принципы создания классов, наследования и полиморфизма;
- изучить основные алгоритмы сортировки и поиска;
- сформировать умения оценивать сложность алгоритмов;
- научиться выбирать подходящий формат хранения данных для прикладной задачи.

4.4.3. Планируемые результаты обучения

Процесс изучения раздела направлен на формирование следующих компетенций

Код компетенции	Перечень планируемых результатов обучения по модулю	Формы и методы обучения, способствующие формированию и развитию компетенции
ОПК-8	**Знать:** Методы и средства проектирования компьютерного программного обеспечения. **Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов. **Владеть:** Проектированием структур данных.	Формы обучения: Фронтальная. Методы обучения: лекции; практические занятия; самостоятельная работа.

4.4.4. Содержание курса

Тема 4.1. Классы и объекты (6 часов).

Лекции (4 часа). Введение в объектно-ориентированное программирование (ООП). Класс и экземпляр класса. Данные экземпляра, методы экземпляра и свойства экземпляра. Создание собственного класса. Инкапсуляция. Атрибуты класса.

Практические занятия (1 час). Разработка собственных классов. Определение нужных методов и свойств классов. Создание нескольких объектов.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Классы и объекты	Чтение и изменение атрибута.	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2	устный опрос

Тема 4.2. Наследование (5 часов).

Лекции (3 часа). Роль наследования в ООП, понятие иерархии наследования. Принцип утиной типизации. Понятие базового класса и производного класса. Функция isinstance и ее применение. Создание производного класса.

Практические занятия (1 час). Расширение готовых классов.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Наследование	Применение экземпляров базового и производного класса.	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2	устный опрос

Тема 4.3. Абстрактные классы и полиморфизм (6 часов)

Лекции (4 часа). Полиморфизм. Принцип DRY и WET. Понятие абстракции. Знакомство с абстрактными классами языка программирования Python. Подмена методов в производном классе. Полиморфные классы. Контейнерные типы. Библиотечные модули collections и collections.abc. Применение контейнерных типов.

Практические занятия (2 часа). Имплементация новых методов.

Тема 4.4. Алгоритмы сортировки и поиска в языке программирования Python (6 часов)

Лекции (4 часа). Сложность алгоритмов, O(N)-нотация. Типы сортировки. Основные алгоритмы сортировки и поиска. Этапы разработки алгоритма. Разработка на языке программирования Python оптимальных алгоритмов поиска.

Практические занятия (1 час). Реализация алгоритма решения задачи Судоку на языке Python.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Алгоритмы сортировки и поиска в языке программирования Python	Оптимизация алгоритма	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2	устный опрос

Тема 4.5. Алгоритмы поиска на графах (5 часов)

Лекции (4 часа). Введение в теорию графов, основные алгоритмы на графах. Теория графов. Представление графов в языке программирования Python. Поиск в ширину. Поиск в глубину.

Практические занятия (1 час). Имплементация графа в языке программирования Python. Нахождение кратчайшего пути проезда на примере графа станций московского метрополитена.

Тема 4.6. Записи данных. Работа с данными системы SQLite (5 часов)

Лекции (3 часа). Разработка структуры данных. Сохранение данных. Список и запись. Записи и таблицы. Чтение и запись. Понятие об объектно-реляционном соответствии.

Практические занятия (1 час). Хранение данных пользователей в СУБД SQLite.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Записи данных. Работа с данными системы SQLite	Использование СУБД SQLite для хранения данных графа московского метрополитена	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2	устный опрос

Тема 4.7. Хранение данных вне программы (5 часов)

Лекции (4 часа). Понятие о структуре данных. Использование структуры для хранения данных. Хранение данных вне языка программирования Python. Сохранение и восстановление данных. Библиотечный модуль pickle и shelve.

Практические занятия (1 час). Сохранение данных графа и использованием модуля pickle.

Тема 4.8. Работа с данными JSON в языке программирования Python (6 часов)

Лекции (4 часа). Знакомство с форматом JSON. Работа с форматом JSON в языке программирования Python. Сериализация и десериализация файлов JSON в языке программирования Python. Хранение данных вне языка программирования Python.

Практические занятия (1 час). Составление графа московского метрополитена на основе файла JSON.

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Работа с данными JSON в языке программирования Python	Сохранение данных в файл данных JSON	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2	устный опрос

Тема 4.9. Промежуточная аттестация (1 час)

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Промежуточная аттестация	Программирование на языке Python. Алгоритмы, работа с данными	Проработка дополнительной литературы	Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397	устный опрос

4.4.5. Оценочное средство для текущего контроля (примерные вопросы для устного опроса):

Тема 4.1. Классы и объекты.

1. Как производится отладка программы в языке программирования Python? Можно ли выполнить Python-код пошагово?

2. Рассмотрите два подхода ниже для инициализации массива и массивов. В чём разница между этими подходами и почему вам следует использовать только один из них?

```
# Инициализация массива -- метод 1
...
x = [[1,2,3,4]] * 3
x
[[1, 2, 3, 4], [1, 2, 3, 4], [1, 2, 3, 4]]

# Инициализация массива -- метод 2
...
y = [[1,2,3,4] for _ in range(3)]
```

```
y
[[1, 2, 3, 4], [1, 2, 3, 4], [1, 2, 3, 4]]
```

Тема 4.2. Наследование

1. Что такое NumPy? Почему лучше использовать массивы NumPy вместо списков языка программирования Python?
2. Как можно создать массив NumPy в языке программирования Python?
3. Зачем использовать декораторы функций?

Тема 4.3. Абстрактные классы и полиморфизм

1. Что такое @classmethod, @staticmethod, @property?
2. Создайте декоратор, записываемый в лог.

Тема 4.4. Алгоритмы сортировки и поиска в Python

Посмотрите на небольшой декоратор, который можно использовать для записи названия любой функции и того, что она делает:

```
import logging

def log(func):
    """
    Логируем какая функция вызывается
    """

    def wrap_log(*args, **kwargs):
        name = func.__name__
        logger = logging.getLogger(name)
        logger.setLevel(logging.INFO)

        # Открываем файл логов для записи
        fh = logging.FileHandler("%s.log" % name)
        fmt = '%(asctime)s - %(name)s - %(levelname)s - %(message)s'
        formatter = logging.Formatter(fmt)
        fh.setFormatter(formatter)
        logger.addHandler(fh)

        logger.info("Вызов функции: %s" % name)
        result = func(*args, **kwargs)
        logger.info("Результат: %s" % result)
        return func
```

```
    return wrap_log


@log
def double_function(a):
    """
    Умножаем полученный параметр
    """
    return a*2


if __name__ == "__main__":
    value = double_function(2)
```

Тема 4.5. Алгоритмы поиска на графах

1. Введение в теорию графов, основные алгоритмы на графах.
2. Теория графов. Представление графов в языке программирования Python.
3. Поиск в ширину.
4. Поиск в глубину.

Тема 4.6. Записи данных. Работа с данными SQLite.

1. Разработка структуры данных.
2. Сохранение данных.
3. Список и запись. Записи и таблицы.
4. Чтение и запись. Понятие об объектно-реляционном соответствии.

Тема 4.7. Хранение данных вне программы

1. Понятие о структуре данных.
2. Использование структуры для хранения данных.
3. Хранение данных вне языка программирования Python.

Тема 4.8. Работа с данными JSON в Python

1. Знакомство с форматом JSON. Работа с форматом JSON в языке программирования Python.
2. Сериализация и десериализация файлов JSON в языке программирования Python.
3. Хранение данных вне языка программирования Python.
4. Сохранение данных в файл JSON.

Тема 4.9. Промежуточная аттестация

Билет №1

1. Класс и экземпляр класса. Атрибуты и методы экземпляра.
2. Сложность алгоритмов, O(N)-нотация.
3. Приведите пример кода создания производного класса.
4. Поиск в ширину: принцип работы.
5. Сериализация и десериализация JSON в Python.

Билет №2

1. Инкапсуляция. Атрибуты класса.
2. Основные алгоритмы сортировки.
3. Приведите пример кода абстрактного класса (модуль abc).
4. Представление графов в Python.
5. Модули pickle и shelve.

Билет №3

1. Наследование, утиная типизация, функция isinstance.
2. Полиморфизм. Принцип DRY.
3. Приведите пример кода записи данных в СУБД SQLite.
4. Поиск в глубину.

4.5. Рабочая программа модуля 5 «Язык программирования SQL. Уровень 1: создание запросов и анализ данных»

4.5.1. Цель изучения модуля: сформировать у обучающихся знания, навыки и умения в области разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

4.5.2. Задачи изучения модуля:

- освоить основные типы данных, операторы, функции и выражения языка SQL;
- научиться формулировать SQL-запросы для выборки, фильтрации и сортировки данных;
- освоить работу с таблицами, представлениями, временными таблицами и несколькими таблицами;
- изучить основы нормализации баз данных, хранилищ и витрин данных;
- сформировать навыки применения групповых, математических, статистических и текстовых функций;
- научиться использовать SQL для подготовки отчетов и анализа данных;
- познакомиться с методами статистического анализа, сегментирования, корреляции и регрессии.

4.5.3. Планируемые результаты обучения

Процесс изучения раздела направлен на формирование следующих компетенций

Код компетенции	Перечень планируемых результатов обучения по модулю	Формы и методы обучения, способствующие формированию и развитию компетенции
ОПК-8	**Знать:** Методы и средства проектирования компьютерного программного обеспечения. **Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов. **Владеть:** Проектированием структур данных.	Формы обучения: Фронтальная. Методы обучения: лекции; практические занятия; самостоятельная работа.

4.5.4. Содержание курса

Тема 5.1. Введение. Постановка задачи (4 часа)

Лекция (2 часа). Анализ бизнес-требований, определение масштаба проблемы. Метрики оценки модели. Определение конечной цели проекта интеллектуального анализа данных. Элементы программирования. Переменные в языке SQL. Объявление переменных. Присвоение значения. Логические выражения и условные вычисления. Приоритет операторов.

Самостоятельная работа (2 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Введение. Постановка задачи	Переменные в языке SQL	Проработка дополнительной литературы	Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.	Устный опрос

Тема 5.2. Типы данных, функции, операторы и программные модули (11 часов)

Лекция (1 час). Логические выражения и условные вычисления. Типы данных. Математические функции. Работа с календарём. Статистические функции. Работа с текстом. Представления, процедуры и пользовательские функции. Табличные выражения. Программирование и динамические запросы.

Практические занятия (6 часа). Работа с базой данных и представление данных. Использование функций различных видов.

Самостоятельная работа (4 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля

Типы данных, функции, операторы и программные модули	Операторы в языке SQL	Проработка дополнительной литературы	Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.	Устный опрос

Тема 5.3. Структуры данных (9 часов)

Лекция (1 час). Нормализованные базы данных. Хранилища и витрины данных. Объектные данные. Данные в формате XML. Работа с большими таблицами. Иерархии. Сети и графы.

Практические занятия (6 часов). Работа со структурой баз данных. Работа с хранилищем и витриной данных (DWH, Data Marts). Эффективная работа с программой Management Studio (SSMS).

Самостоятельная работа (2 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Структуры данных	Изменение структуры баз данных	Проработка дополнительной литературы	Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.	Устный опрос

Тема 5.4. Методы анализа (9 часов)

Лекция (1 час). Вычисления и поиск. Работа с упорядоченными сериями. Распределение. Сегментирование. Регрессия. Корреляция. Анализ связанных событий. Вероятностные модели. Другие методы анализа данных.

Практические занятия (6 часов). Статистический анализ данных. Инструменты анализа данных. Использование функций на практике для построения отчетов.

Самостоятельная работа (2 часа). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Методы анализа	Сложные методы анализа данных в языке SQL	Проработка дополнительной литературы	Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.	Устный опрос

Тема 5.5. Промежуточная аттестация (1 час)

Самостоятельная работа (1 час). Проработка материала дополнительной литературы по теме.

Наименование темы	Дидактические единицы, вынесенные на самостоятельное изучение	Формы самостоятельной работы	Учебно-методическое обеспечение	Форма контроля
Промежуточная аттестация	Создание запросов и анализ данных на языке SQL	Проработка дополнительной литературы	Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.	Тест

4.5.5. Оценочное средство для текущего контроля:

Тема 5.1. (примерные вопросы для устного опроса):

1. Логические выражения и условные вычисления.
2. Типы данных.
3. Математические функции.
4. Работа с календарём.
5. Статистические функции.
6. Работа с текстом.
7. Представления, процедуры и пользовательские функции.
8. Табличные выражения.
9. Программирование и динамические запросы.

Тема 5.2. (примерные вопросы для устного опроса):

1. Нормализованные базы данных.
2. Хранилища и витрины данных.
3. Объектные данные.
4. Данные в формате XML.
5. Работа с большими таблицами.
6. Иерархии.

7. Сети и графы.

Тема 5.3. (примерные вопросы для устного опроса):

1. Нормализованные базы данных.
2. Хранилища и витрины данных.
3. Объектные данные.
4. Данные в формате XML.
5. Работа с большими таблицами.
6. Иерархии.
7. Сети и графы.

Тема 5.4. (примерные вопросы для устного опроса):

1. Вычисления и поиск.
2. Работа с упорядоченными сериями.
3. Распределение.
4. Сегментирование.
5. Регрессия.
6. Корреляция.
7. Анализ связанных событий.
8. Вероятностные модели.
9. Методы анализа данных

Тема 5.5. (примерные вопросы теста):

1. Как удалить таблицу "my_table":
    a. DELETE "my_table"
    b. DELETE TABLE "my_table"
    c. DELETE FROM "my_table"
    d. DROP TABLE "my_table"

2. Требуется отсортировать записи по убыванию поля "field". Какая конструкция будет верна?
    a. ORDER BY "field" DESC
    b. SORT BY "field"
    c. ORDER BY "field"

d. SORT BY "field" DESC

3. Выберите верное утверждение относительно индекса:
    a. Индекс позволяет сэкономить место, занимаемое таблицей.
    b. Индекс позволяет ускорить выборку с тем полем, для которого он сделан.
    c. Индекс ускоряет абсолютно любые запросы с таблицей.
    d. Индекс ускоряет добавление записей в таблицу.

4. Для чего используется команда GRANT?
    a. Для очистки таблицы.
    b. Для управления пользователями.
    c. Для перезапуска сервера MySQL.
    d. Для очистки базы данных

5. Если выполнить следующий запрос: SELECT * FROM `my_table` WHERE 1=1 Что получится в результате?
    a. Синтаксическая ошибка (из-за *).
    b. Выведутся все записи из таблицы "my_table".
    c. Не выведется ни одной записи.
    d. Синтаксическая ошибка (из-за 1=1).

6. Что означает [a-z] при задании шаблона для LIKE?
    a. Он означет любое количество символов в диапазоне от a до z.
    b. Он означает любое количество символов НЕ из диапазона от a до z.
    c. Он означает наличие 1-го символа НЕ из диапазона от a до z.
    d. Он означает наличие 1-го символа в диапазоне от a до z.

7. Необходимо извлечь все записи, где значение поля "field" начинается с подстроки "abc". Какой запрос ей необходимо использовать?
    a. SELECT * FROM `my_table` WHERE `field` <> 'abc'
    b. SELECT * FROM `my_table` WHERE `field` LIKE 'abc%'
    c. SELECT * FROM `my_table` WHERE `field` LIKE '%abc%'
    d. SELECT * FROM `my_table` WHERE `field` STARTSWITH 'abc'.
8. Для чего применяются индексы?
    a. Для выборки из нескольких таблиц в одном запросе.

b. Для уменьшения места, занимаемого таблицей.

c. Для ускорения операций выборки.

d. Для восстановления после случайного изменения.

9. Выберите верное утверждение:

a. Приоритет AND и OR одинаковый.

b. Если использовать AND и OR в одном запросе, то возникнет ошибка.

c. Приоритет AND выше, чем OR.

d. Приоритет AND ниже, чем OR.

10. Как обновить запись сразу в двух колонках:

a. UPDATE `my_table` SET `field_1` = 'value_1' SET `field_2` = 'value_2'

b. UPDATE `my_table` SET `field_1` = 'value_1' AND `field_2` = 'value_2'

c. UPDATE `my_table` SET `field_1` = 'value_1', SET `field_2` = 'value_2'

d. UPDATE `my_table` SET `field_1` = 'value_1', `field_2` = 'value_2'.

5. УСЛОВИЯ РЕАЛИЗАЦИИ ДПП

5.1. Организационные условия реализации ДПП

Наименование аудитории	Вид занятия	Наименование оборудования, программного обеспечения
Аудитория для проведения лекций/семинаров	Лекции	ПК с доступом в Интернет и возможностью просмотра файлов в формате *.ppt, *.pptx, *.pdf, проектор/телевизор/монитор.
Аудитория для проведения лекций/семинаров	Практические занятия	ПК с доступом в Интернет и возможностью просмотра файлов в формате *.ppt, *.pptx, *.pdf, проектор/телевизор/монитор.
Аудитория для проведения лекций/семинаров	Самостоятельная работа	ПК с доступом в Интернет и возможностью просмотра файлов в формате *.html, *.doc, *.docx, *.pdf, *.djvu.
Аудитория для проведения лекций/семинаров	Итоговая аттестация	ПК с доступом в Интернет и возможностью просмотра файлов в формате *.html, *.doc, *.docx, *.pdf, *.djvu, лист бумаги формата А4, ручка.

5.2. Педагогические условия реализации ДПП

Реализация программы обеспечивается преподавательским составом, удовлетворяющим следующим условиям:

- наличие высшего профессионального образования, соответствующего профилю программы, из числа штатных преподавателей, или привлеченных на условиях почасовой оплаты труда;

- значительный опыт практической деятельности в соответствующей сфере из числа штатных преподавателей или привлеченных на условиях почасовой оплаты труда

5.3. Учебно-методическое обеспечение ДПП

Основная литература:

1. Кумскова, И.А., Базы данных: учебник / И.А. Кумскова. – Москва: КноРус, 2022. – 400 с. – ISBN 978-5-406-09667-3.
2. Полуэктова, Н.Р. Разработка веб-приложений: учебное пособие для вузов / Н.Р. Полуэктова. – 2-е изд. – Москва: Издательство Юрайт, 2024. – 204 с. – (Высшее образование). – ISBN 978-5-534-18645-1.
3. Гниденко, И.Г Технологии и методы программирования: учебное пособие для вузов / И.Г. Гниденко, Ф.Ф. Павлов, Д.Ю. Федоров. – 2-е изд., перераб. и доп. – Москва: Издательство Юрайт, 2024. – 248 с. – (Высшее образование). – ISBN 978-5-534-18130-2.

4. Кудрявцева, И.А. Программирование: теория типов: учебное пособие для вузов / И.А. Кудрявцева, М.В. Швецкий. – 2-е изд., перераб. и доп. – Москва: Издательство Юрайт, 2024. – 652 с. – (Высшее образование). – ISBN 978-5-534-11088-3.
5. Ковалевская, Е. В. Методы программирования : учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 c. — ISBN 978-5-374-00356-7.
6. Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2.
7. Кобелев Н. Б. Искусственный интеллект и имитация функций человека : учебное пособие / Кобелев Н. Б. - М. : КУРС, 2026. - 111 с. : табл., граф. - Библиогр.: с. 109-110. - ISBN 978-5-907352-11-7.
8. Эртель В. Введение в искусственный интеллект / Эртель В. ; пер. с англ. Горман А. - М. : Эксмо, 2019. - 443 с. : рис. - (Библиотека Сбербанка. Искусственный интеллект). - Библиогр.: с. 419-436. - ISBN 978-5-04-108594-0.
9. Еремин, А. Л. Информационная и цифровая гигиена : учебное пособие для вузов / А. Л. Еремин. — 2-е изд., стер. — Санкт-Петербург : Лань, 2024. — 92 с. — ISBN 978-5-507-47641-1.
10. Бессмертный, И. А. Системы искусственного интеллекта : учебник для вузов / И. А. Бессмертный. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 164 с. — (Высшее образование). — ISBN 978-5-534-18416-7.
11. Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.

Дополнительная литература:

1. Искусственный интеллект и нейросетевое управление : учебное пособие / сост. Мамонова Т. Е. - Томский политехнический университет, 2020. - ISBN 978-5-4387-0921-3.
2. Сузи, Р. А. Язык программирования Python : учебное пособие / Р. А. Сузи. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 350 с. — ISBN 978-5-4497-0705-5.

3. Кудряшев А.В., Светашков П.А. Введение в современные веб-технологии: учебное пособие / Кудряшев А.В., Светашков П.А. – Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. – ISBN 978-5-4497-2388-8.
4. Прешерн К. Язык С. Мастерство программирования. Принципы, практики и паттерны / пер. с англ. А. Н. Слинкина – М.: ДМК Пресс, 2023. – 300 с.
5. Дональд Кнут: Искусство программирования. Том 1. Основные алгоритмы/ пер. с англ. Тригуб С. Г., Гордиенко Ю. Г., Красиков И. В. – М.: Вильямс, 2019. – 720 с.
6. Бизли Д. М. Язык программирования Python: справочник: Пер. с англ/ Бизли Д. М. – Киев: ДиаСофт, 2000. - 326 с. - ISBN 966-7393-54-2.
7. Джеймс Р., Грофф П., Вайнберг Н., Оппель Э. Дж. SQL. Полное руководство. М: Вильяме, 2014. 960 с.
8. Коннолли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. М: Вильяме, 2017. 1440 с.
9. Визуальное программирование на основе библиотеки MFC : методические указания к лабораторным работам по курсу «Визуальное программирование» для студентов направления 09.03.02 Информационные системы и технологии / составители А. Я. Лахов, Р. Е. Борщиков. — Саратов : Вузовское образование, 2016. — 57 с. — ISBN 2227-8397.
10. Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397.
11. Промпт-инжиниринг для LLM. Искусство построения приложений на основе больших языковых моделей», авторы: Джон Берриман, Альберт Циглер. Книга рассказывает о науке и искусстве промпт-инжиниринга, помогает раскрыть потенциал больших языковых моделей (LLM).
12. Промпт-инжиниринг. Практическое руководство для освоения нейросетей с нуля. Том 1: основы», автор: Владимир Водолазский. Книга построена по принципу «от простого к сложному», содержит множество наглядных примеров и практических упражнений.
13. Промпт-инжиниринг для GenAI. Паттерны надёжных запросов для качественных результатов», авторы: Джеймс Феникс, Майк Тейлор. Книга посвящена искусству составления инструкций и подсказок для ИИ, которые позволяют получать качественные и точные результаты.

14. 150+ промптов для продвинутого вайб-кодинга», автор: Лэй Энстазия. Книга представляет собой практическое руководство для программистов, дизайнеров и инженеров, стремящихся раскрыть потенциал LLM в создании реальных приложений. В ней собраны профессиональные промпты и техники для разработки программ, сервисов и интерфейсов на популярных технологических стеках (LAMP, MEAN, MERN, JAMstack, .NET, Django, Spring Boot и других).
15. Полякова, Л. Н. Основы SQL : учебное пособие / Л. Н. Полякова. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 273 c. — ISBN 978-5-4497-0672-0.
16. Введение в модель данных SQL : учебное пособие / С. Кузнецов. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. - ISBN 5-9556-00028-0.
17. Крис, Фиайли SQL / Фиайли Крис ; перевод А. В. Хаванов. — 2-е изд. — Саратов: Профобразование, 2019. — 452 c. — ISBN 978-5-4488-0103-7.
18. Дьяков, И. А. Базы данных. Язык SQL : учебное пособие / И. А. Дьяков. — Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2012. — 81 c. — ISBN 2227-8397.
19. Стасышин, В. М. Практикум по языку SQL : учебное пособие / В. М. Стасышин, Т. Л. Стасышина. — Новосибирск : Новосибирский государственный технический университет, 2016. — 60 c. — ISBN 978-5-7782-2937-2.

5.4. Методические рекомендации

ДПП построена по тематическому принципу, каждый раздел представляет собой логически завершенный материал.

Преподавание программы основано на личностно-ориентированной технологии образования, сочетающей два равноправных аспекта этого процесса: обучение и учение. Личностно-ориентированный подход развивается при участии слушателей в активной работе на практических занятиях. Личностно-ориентированный подход направлен, в первую очередь, на развитие индивидуальных способностей обучающихся, создание условий для развития творческой активности слушателя и разработке инновационных идей, а также на развитие самостоятельности мышления при решении учебных задач разными способами, нахождение рационального варианта решения, сравнения и оценки нескольких вариантов их решения и т.п. Это способствует формированию приемов умственной деятельности по восприятию новой информации, ее запоминанию и

осознанию, созданию образов для сложных понятий и процессов, приобретению навыков поиска решений в условиях неопределенности.

Практические занятия проводятся для приобретения навыков решения практических задач в предметной области модуля. Задания, выполняемые на практических занятиях, выполняются с использованием активных и интерактивных методов обучения.

Самостоятельная работа слушателей предназначена для проработки дополнительной литературы. Результаты практических заданий слушателей учитываются на итоговой аттестации.

При изучении курса предусмотрены следующие методы организации и осуществления учебно-познавательной деятельности:

- объяснительно-иллюстративный метод;
- репродуктивный метод;
- частично-поисковый метод.

6. ФОРМЫ ИТОГОВОЙ АТТЕСТАЦИИ ДПП

Итоговая аттестация проводится в форме зачета для проверки сформированности компетенций, полученных в рамках ДПП. Результатом зачета служат правильные ответы на вопросы теста, состоящего из 40 (сорока) вопросов.

По результатам итоговой аттестации слушателю выставляется оценка «ЗАЧТЕНО/НЕ ЗАЧТЕНО»:

Оценка «ЗАЧТЕНО» выставляется слушателю, который:

- правильно ответил не менее чем на 75% вопросов теста;

- продемонстрировал необходимые систематизированные знания и достаточную степень владения принципами предметной области программы, понимание их особенностей и взаимосвязь между ними в течение всего срока обучения по ДПП.

Оценка «НЕ ЗАЧТЕНО» ставится слушателю, который:

- ответил правильно менее чем на 75% вопросов теста;

- имеет крайне слабые теоретические и практические знания, обнаруживает неспособность к построению самостоятельных заключений.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ИТОГОВОЙ АТТЕСТАЦИИ

7.1. Паспорт комплекта оценочных средств

Предметы оценивания	Объекты оценивания	Показатели оценки
ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Ответы на вопросы теста	Количество правильных ответов

7.2. Комплект оценочных средств

7.2.1. Темы для подготовки к зачету:

1. Основные понятия программирования: алгоритм, программа, данные, типы данных, операторы и выражения.
2. Основы языка программирования Python: переменные, константы, динамическая типизация и преобразование типов.
3. Ввод и вывод данных. Линейная структура программы.
4. Условные конструкции и циклы в Python. Операторы break, continue, pass.
5. Основные встроенные типы данных Python: числа, строки, логический тип и None.
6. Коллекции Python: списки, кортежи, словари и множества; основные операции и срезы.
7. Функции: создание, вызов, параметры, аргументы, возвращаемые значения и область видимости.
8. Итераторы, генераторы, генераторные выражения, функции map, filter, zip, reduce и лямбда-функции.
9. Модули, пакеты и установка сторонних библиотек с использованием pip.
10. Работа с файлами, каталогами и путями. Чтение, запись и обработка данных.
11. Исключения и обработка ошибок в Python.
12. Регулярные выражения и их применение для поиска и обработки текстовых данных.
13. Основы функционального программирования: замыкания, каррирование и декораторы.
14. Разработка программ с применением ИИ: понятие вайбкодинга, возможности и ограничения ИИ-чатов.
15. Постановка задачи для ИИ-чата и разработка эффективных промптов для генерации программного кода.
16. Проверка, отладка и улучшение кода, сгенерированного искусственным интеллектом.

17. Автоматизация обработки файлов, таблиц, текстов и формирования отчетов с помощью Python и ИИ.
18. Программные интерфейсы нейросетевых моделей, структурированные ответы и безопасное хранение ключей доступа.
19. Объектно-ориентированное программирование в Python: классы, объекты, атрибуты, методы, инкапсуляция, наследование и полиморфизм.
20. Алгоритмы и структуры данных: оценка сложности, сортировка и поиск, графы, SQLite, pickle, shelve и JSON.
21. Базы данных и таблицы.
22. СУБД SQL Server.
23. Основные принципы работы с программой Management Studio (SSMS).
24. Получение доступа к базам данных различных программ.
25. Извлечение, обработка и подготовка данных различных источников.
26. Проведение фильтрации данных по столбцам и строкам.
27. Использование встроенных функций и выражений.
28. Особенности работы с различными типами данных: числовыми, строковыми, дата и время.
29. Выполнение сортировки данных.
30. Работа с различными типами данных.
31. Осуществление простейших действий в одной таблице.
32. Использование групповых функций.
33. Фильтрация групп.
34. Использование временных таблиц.
35. Редактирование временных таблиц.
36. Функции в нескольких таблицах.
37. Операции с несколькими таблицами.
38. Использование нескольких таблиц в запросе.
39. Работа со сводными таблицами.
40. Использование функций для построения отчетов.
41. Операционные базы данных.
42. Структура хранилища и работа с ним.
43. Статистический анализ данных и инструменты анализа.
44. Реляционное деление нескольких таблиц

7.2.2. Примерные вопросы теста для проведения итоговой аттестации:

1. Что такое алгоритм?

а) набор случайных команд

б) конечная последовательность действий для решения задачи

в) файл с исходным кодом

г) база данных

2. Какой тип данных используется для хранения целых чисел в Python?

а) float

б) str

в) int

г) bool

3. Каков результат выражения 7 // 2?

а) 3

б) 3.5

в) 4

г) 1

4. Какая функция используется для вывода данных на экран?

а) input()

б) write()

в) print()

г) display_file()

5. Какая функция используется для получения данных от пользователя?

а) read()

б) input()

в) scan()

г) get()

6. Какой оператор используется для ветвления?

а) for

б) if

в) def

г) import

7. Какой цикл обычно используется для перебора элементов последовательности?

а) for

б) switch

в) case

г) select

8. Какой оператор завершает выполнение цикла?

а) continue

б) pass

в) break

г) stop

9. Какой тип данных представляет собой изменяемую упорядоченную коллекцию?

а) кортеж

б) список

в) множество

г) строка

10. Какой тип данных хранит пары «ключ — значение»?

а) list

б) tuple

в) dict

г) set

11. Как создать функцию в Python?

а) с помощью function

б) с помощью def

в) с помощью func

г) с помощью new

12. Что делает оператор return?

а) завершает программу Python

б) возвращает результат из функции

в) создает переменную

г) подключает модуль

13. Для чего используется оператор import?

а) для удаления файла

б) для подключения модуля или пакета

в) для создания класса

г) для запуска цикла

14. Какой режим открытия файла используется для чтения?

а) w

б) a

в) r

г) x

15. Для чего используется конструкция try...except?

а) для объявления класса

б) для обработки исключений

в) для сортировки списка

г) для подключения библиотеки

16. Что такое регулярное выражение?

а) разновидность базы данных

б) шаблон для поиска и обработки текста

в) способ объявления класса

г) формат хранения изображений

17. Что представляет собой лямбда-функция?

а) именованный класс

б) анонимная функция

в) модуль Python

г) обработчик исключений

18. Что такое декоратор?

а) функция, изменяющая или расширяющая поведение другой функции

б) тип данных

в) разновидность цикла

г) формат файла

19. Что такое класс в объектно-ориентированном программировании?

а) конкретное значение переменной

б) описание структуры и поведения объектов

в) файл с настройками

г) команда запуска программы

20. Что представляет собой экземпляр класса?

а) объект, созданный на основе класса

б) имя модуля

в) условный оператор

г) строковый литерал

21. Что такое наследование?

а) удаление объекта

б) создание нового класса на основе существующего

в) сортировка объектов

г) преобразование строки в число

22. Что означает полиморфизм?

а) возможность объектов разных классов использовать общий интерфейс

б) запрет создания объектов

в) обязательное использование одного типа данных

г) удаление методов класса

23. Что показывает нотация O(n)?

а) объем файла

б) оценку роста времени или ресурсов алгоритма при увеличении размера входных данных

в) количество классов

г) число строк программы

24. Какой алгоритм поиска используется для последовательного просмотра элементов?

а) линейный поиск

б) двоичный поиск

в) поиск в глубину

г) сортировка слиянием

25. Какой алгоритм поиска используется для обхода графа по уровням?

а) DFS

б) BFS

в) JSON

26. Для чего используется SQLite?

а) для создания графических интерфейсов

б) для работы с реляционными базами данных

в) для запуска виртуальных машин

г) для компиляции Python

27. Для чего используется формат JSON?

а) для структурированного хранения и обмена данными

б) для написания циклов

в) для создания классов

г) для запуска командной строки

28. Что такое сериализация?

а) удаление данных

б) преобразование объекта в формат, пригодный для хранения или передачи

в) сортировка списка

г) проверка синтаксиса

29. Почему ИИ-сгенерированный код необходимо проверять?

а) ИИ всегда создает только синтаксически неверный код

б) ИИ может допускать логические, синтаксические и архитектурные ошибки

в) проверка требуется только для программ без функций

г) проверка нужна только при работе с текстом

30. Как следует хранить ключ доступа к программному интерфейсу нейросетевой модели?

а) непосредственно в открытом исходном коде

б) в комментарии программы

в) в переменной окружения или защищенном хранилище

г) в имени файла

31. Как удалить таблицу "my_table":

а) DELETE "my_table"

б) DELETE TABLE "my_table"

в) DELETE FROM "my_table"

г) DROP TABLE "my_table"

32. Требуется отсортировать записи по убыванию поля "field". Какая конструкция будет верна?

а) ORDER BY "field" DESC

б) SORT BY "field"

в) ORDER BY "field"

г) SORT BY "field" DESC

33. Выберите верное утверждение относительно индекса:

а) Индекс позволяет сэкономить место, занимаемое таблицей.

б) Индекс позволяет ускорить выборку с тем полем, для которого он сделан.

в) Индекс ускоряет абсолютно любые запросы с таблицей.

г) Индекс ускоряет добавление записей в таблицу.

34. Для чего используется команда GRANT?

а) Для очистки таблицы.

б) Для управления пользователями.

в) Для перезапуска сервера MySQL.

г) Для очистки базы данных

35. Если выполнить следующий запрос: SELECT * FROM `my_table` WHERE 1=1 Что получится в результате?

а) Синтаксическая ошибка (из-за *).

б) Выведутся все записи из таблицы "my_table".

в) Не выведется ни одной записи.

г) Синтаксическая ошибка (из-за 1=1).

36. Что означает [a-z] при задании шаблона для LIKE?

а) Он означает любое количество символов в диапазоне от a до z.

б) Он означает любое количество символов НЕ из диапазона от a до z.

в) Он означает наличие 1-го символа НЕ из диапазона от a до z.

г) Он означает наличие 1-го символа в диапазоне от a до z.

37. Необходимо извлечь все записи, где значение поля "field" начинается с подстроки "abc". Какой запрос необходимо использовать?

а) SELECT * FROM `my_table` WHERE `field` <> 'abc'

б) SELECT * FROM `my_table` WHERE `field` LIKE 'abc%'

в) SELECT * FROM `my_table` WHERE `field` LIKE '%abc%'

г) SELECT * FROM `my_table` WHERE `field` STARTSWITH 'abc'.

38. Для чего применяются индексы?

а) Для выборки из нескольких таблиц в одном запросе.

б) Для уменьшения места, занимаемого таблицей.

в) Для ускорения операций выборки.

г) Для восстановления после случайного изменения.

39. Выберите верное утверждение:

а) Приоритет AND и OR одинаковый.

б) Если использовать AND и OR в одном запросе, то возникнет ошибка.

в) Приоритет AND выше, чем OR.

г) Приоритет AND ниже, чем OR.

40. Как обновить запись сразу в двух колонках:

а) UPDATE `my_table` SET `field_1` = 'value_1' SET `field_2` = 'value_2'

б) UPDATE `my_table` SET `field_1` = 'value_1' AND `field_2` = 'value_2'

в) UPDATE `my_table` SET `field_1` = 'value_1', SET `field_2` = 'value_2'

г) UPDATE `my_table` SET `field_1` = 'value_1', `field_2` = 'value_2'.

АВТОРЫ ПРОГРАММЫ:

Преподаватель ЦДО _____ Н.А. Степаненко

СОГЛАСОВАНО:

Начальник УСП _____ Т.А. Гузева

Директор
Центра дополнительного образования _____ М.В. Стоянова

№06.05-11/569