

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Кожедеров Александр Игоревич
Должность: Директор филиала ИнДИ (филиал) ФГБОУ ВО "ЮГУ"
Дата подписания: 16.07.2026 23:55:14
Уникальный программный ключ: 7f4522f81a862743c2711b37d9dd0f6adf40c4c8

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Югорский государственный университет»

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы программирования на Python

Направление подготовки (специальности): 21.03.01 *Нефтегазовое дело*

Профиль: *Эксплуатация и обслуживание объектов добычи нефти (ИнДИ)*

Форма обучения
Очно-заочная

Квалификация выпускника
Бакалавр

2026 год набора

Виды работ	Объём занятий по семестрам, час										Итого
	1	2	3	4	5	6	7	8	9	10	
Самостоятельная работа			84								84
Дистанционные лекции			10								10
Дистанционные практические занятия			14								14
Форма контроля			Зачёты								-
Итого:			108								108
з.е.			3								3

Ханты-Мансийск, 2026 год
(город)

Предисловие

1. Программа разработана в соответствии с требованиями Федерального закона от 27.12.2012 г. № 273-ФЗ «Об образовании в Российской Федерации», федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) *21.03.01 Нефтегазовое дело* утвержденного № 96 от 09.02.2018 года.

2. Разработчик(и):

Кандидат наук

ученая степень, ученое звание
(при наличии)

(подпись)

Г. В. Газя

(И. О. Фамилия)

3. Согласовано:

Руководитель
образовательной
программы по
направлению подготовки
21.03.01 Нефтегазовое
дело

(подпись)

Г. В. Газя

(И. О. Фамилия)

4. Утверждаю:

Руководитель
структурного
подразделения
Индустриальный институт
(г. Нефтеюганск)

(подпись)

А. П. Янукия

(И. О. Фамилия)

Документ подписан простой электронной подписью в
электронной информационно образовательной среде
Elios 2.0 ФГБОУ ВО «ЮГУ»

Идентификатор документа: 96482



Подписант



Газя Геннадий Владимирович



Янукия Арам Погосович

Дата подписания

18.05.2026 03:22:02

18.05.2026 09:09:43

1 Цель освоения дисциплины

Целью освоения дисциплины является формирование базовых знаний, умений и практических навыков разработки программ на языке Python для решения прикладных, инженерных и профессионально-ориентированных задач.

2 Место дисциплины в структуре ОПОП

Дисциплина относится к обязательной части блока ФТД учебного плана.

3 Формируемые компетенции обучающегося

Планируемые результаты освоения ОПОП (компетенции), достижение которых обеспечивает дисциплина		Планируемые результаты (соотнесенные с установленными индикаторами достижения компетенции)
код компетенции	наименование компетенции	
ОПК-1	Способен решать задачи, относящиеся к профессиональной деятельности, применяя методы моделирования, математического анализа, естественнонаучные и общинженерные знания	ОПК-1.2 З-1: Принципиальные особенности моделирования математических, физических и химических процессов, предназначенные для конкретных технологических процессов; принципы совершенствования производственных процессов с использованием экспериментальных данных и результатов моделирования ОПК-1.2 У-1: Систематизировать, анализировать и отбирать необходимую информацию для математического анализа; выбирать и применять соответствующие математические методы моделирования физических, химических и технологических процессов в нефтегазовой отрасли ОПК-1.1 В-1: Навыками разработки схемы лабораторных исследования; практическими приемами отбора образцов для лабораторного исследования; правильной эксплуатации основных приборов и оборудования современной лаборатории; обработки и интерпретирования результатов эксперимента. Навыками чтения и построения геологической графики, определения основных горных пород, интерпретации геолого-промысловой информации. Навыками решения прикладных

		задач гидравлики и гидромеханики, встречающихся в нефтегазовом деле; методами подземной гидромеханики для обеспечения рационального использования эксплуатируемых месторождений нефти и газа. Навыками использования основных общезначимых законов и принципов в важнейших практических приложениях; применения основных методов физико-математического анализа для решения естественнонаучных задач; навыками применения классических методов механики к анализу математических моделей формализованных материальных объектов. ОПК-1.2 В-1: Опытном анализе геолого-промысловой информации на непротиворечивость и достоверность; математического описания и анализа геологических процессов и явлений; навыками моделирования нефтегазовых залежей по геолого-геофизическим материалам
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 З-1: Знает основные математические методы решения задач, принципы математических рассуждений, математических доказательств и системного подхода УК-1.2 З-1: Знает возможности и принципы функционирования цифровых сервисов и технологий, используемых для работы с информацией УК-1.3 З-1: Знает основные различия между фактами, мнениями, интерпретациями и оценкам УК-1.4 З-1: Знает принципы сбора, отбора и обобщения информации УК-1.1 У-1: Умеет обосновывать выбор варианта решения и практически применять стандартные математические методы и

	<i>системный подход в решении поставленных задач</i> УК-1.2 У-1: <i>Умеет обосновывать выбор и использовать цифровые сервисы и технологии для безопасной и эффективной работы с информацией</i> УК-1.3 У-1: <i>Умеет формировать собственную позицию о фактах, мнениях, интерпретациях и оценках информации</i> УК-1.4 У-1: <i>Умеет критически оценивать полноту, адекватность и достоверность информации, необходимой для решения поставленных задач</i> УК-1.5 У-1: <i>Умеет производить постановку проблемы путем фиксации ее содержания, выявления субъекта проблемы, а также всех заинтересованных сторон в данной ситуации, а также осуществлять анализ ситуации в реальных социальных условиях для выявления актуальной, в т.ч. социально-значимой задачи/проблемы, требующей решения</i> УК-1.1 В-1: <i>Владеет навыком решения различных прикладных задач с использованием математических методов и системного подхода</i> УК-1.2 В-1: <i>Имеет практический опыт решения задач обработки информации с использованием различных цифровых сервисов и технологий, в т.ч. во взаимодействии с другими людьми в цифровой среде</i> УК-1.3 В-1: <i>Владеет навыками рассуждения и аргументации</i> УК-1.4 В-1: <i>Владеет навыками систематизации и синтеза информации, полученной из различных источников</i> УК-1.5 В-1: <i>Имеет опыт определения</i>
--	---

		<i>требований и ожиданий заинтересованных сторон с учетом социального контекста</i>
--	--	---

4 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетных единицы, 108 часов.

№ п/п	Тема	Трудоемкость по видам учебной работы, час					Код компетенции	Оценочные средства
		Занятия лекционного типа	Практические занятия	Лабораторные занятия	Консультации	Самостоятельная работа		
1	Введение в программирование и язык Python	2	2			16	ОПК-1; УК-1.	Тест; Ситуационные задачи.
2	Базовые конструкции программирования на Python	2	3			17	ОПК-1; УК-1.	Тест; Реферат; Ситуационные задачи.
3	Функции и модульное программирование	2	3			17	ОПК-1; УК-1.	Тест; Ситуационные задачи.
4	Работа с файлами и обработка данных	2	3			17	ОПК-1; УК-1.	Тест; Ситуационные задачи.
5	Основы прикладного программирования на Python	2	3			17	ОПК-1; УК-1.	Тест; Ситуационные задачи.
Итого		10	14			84	—	

5 Образовательные технологии, используемые при различных видах учебной работы

№ темы	Образовательная технология
1-5	Технология традиционного обучения
1-5	Дистанционные технологии

6 Методические материалы по освоению дисциплины

Электронная информационно - образовательная среда представлена личным кабинетом, расположенным по ссылке <https://itport.ugrasu.ru>, электронной библиотечной

системой <https://lib.ugrasu.ru>, электронным каталогом Научной библиотеки ЮГУ <https://irbis.ugrasu.ru> и системой дистанционного обучения.

Методические материалы для обучающихся представлены в электронном виде в системе Moodle по ссылке <http://eluniver.ugrasu.ru>.

Методические материалы для обучающихся из числа инвалидов и лиц с ОВЗ предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

6.1 Методические указания к занятиям лекционного типа

Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать его научно-педагогическому работнику на консультации, на практическом занятии.

6.2 Методические указания к практическим занятиям

Целью практических занятий является закрепление теоретических знаний и приобретение практических умений и навыков. Методические рекомендации по каждой практической работе имеют теоретическую часть, подготовленную отдельно, или указание на источник, необходимый для подготовки к соответствующему практическому занятию, с необходимыми для выполнения работы формулами, пояснениями, таблицами и графиками; алгоритм выполнения заданий. Практические задания сочетаются с теоретическими знаниями. Проведению практического занятия как правило предшествует самостоятельная работа обучающегося.

6.3 Методические указания к самостоятельной работе

В рамках самостоятельной работы обучающийся знакомится с рабочей программой, особое внимание должно уделяться целям и задачам, структуре и содержанию дисциплины. Анализируется конспект лекций, ведется подготовка ответов к контрольным вопросам, просматривается рекомендуемая литература, используются аудио-видеозаписи по заданной теме, решаются расчетно-графические задания, задачи по алгоритму и др.

7 Оценочные материалы для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины, учебно-методическое обеспечение самостоятельной работы обучающихся.

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплин (модулей). Для осуществления процедуры текущего контроля успеваемости обучающихся НПП создаются оценочные материалы (фонды оценочных средств), позволяющие оценить достижение запланированных результатов обучения и уровень сформированности компетенций.

Промежуточная аттестация обучающихся производится в дискретные временные интервалы НПП, обеспечивающими реализацию дисциплины в форме: зачёты.

Учебно-методическое обеспечение самостоятельной работы обучающихся предполагает предоставление студентам методических рекомендаций по изучению дисциплины, учитывающих особенности ее построения, освоения, преподавания и представлено как электронный учебно-методический комплект документов по дисциплине, размещено в системе управления обучением «Moodle» (сайт Университета

по ссылке <http://eluniver.ugrasu.ru>) и/или в других системах управления обучением электронной информационно-образовательной среды Университета.

Обучение и контроль обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья при необходимости осуществляется с использованием специальных методов обучения и дидактических материалов, составленных с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (обучающегося).

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

7.1 Технологическая карта дисциплины 3-й семестр

№ п/п	Название темы	Максимальное количество баллов
Обязательный уровень (текущая аттестация)		
1	Введение в программирование и язык Python	14
2	Базовые конструкции программирования на Python	14
3	Функции и модульное программирование	14
4	Работа с файлами и обработка данных	14
5	Основы прикладного программирования на Python	14
		70
Обязательный уровень (промежуточная аттестация)		
6	Зачёты	30
		30
	Итого	100
Дополнительный уровень		
7	Контрольная работа	15
		15

Шкала оценивания результатов по балльной системе (зачёты):

Зачтено с 50 по 100 баллов;

Не зачтено с 0 по 49 баллов.

7.2 Примерные тестовые задания

- Какой символ используется для комментариев в Python?
 - //
 - #
 - /*
 -
- Какой тип данных используется для хранения целых чисел в Python?
 - float
 - str
 - int
 - bool
- Какая функция используется для вывода информации на экран?
 - input()

- b. print()
 - c. write()
 - d. output()
4. Какой оператор используется для присваивания значения переменной?
 - a. ==
 - b. :=
 - c. =
 - d. !=
 5. Какой результат будет у выражения $5 + 2 * 3$?
 - a. 21
 - b. 11
 - c. 15
 - d. 13
 6. Какой тип данных используется для хранения текстовой информации?
 - a. str
 - b. int
 - c. bool
 - d. list
 7. Какая функция используется для ввода данных с клавиатуры?
 - a. scan()
 - b. read()
 - c. get()
 - d. input()
 8. Какой оператор используется для проверки равенства двух значений?
 - a. =
 - b. ==
 - c. !=
 - d. <=
 9. Какой цикл используется в Python для перебора элементов коллекции?
 - a. repeat
 - b. while
 - c. for
 - d. loop
 10. Как обозначается логическое значение «истина» в Python?
 - a. true
 - b. TRUE
 - c. True
 - d. yes
 11. Какой тип данных используется для хранения последовательности элементов?
 - a. float
 - b. list
 - c. int
 - d. bool
 12. Какой метод используется для добавления элемента в список?
 - a. add()
 - b. push()
 - c. insert()
 - d. append()
 13. Что делает функция len()?
 - a. вычисляет сумму элементов
 - b. определяет тип данных

- c. возвращает длину объекта
- d. сортирует данные
- 14. Какой оператор используется для логического «и»?
 - a. &&
 - b. and
 - c. &
 - d. +
- 15. Какой результат будет у выражения 10 // 3?
 - a. 3.33
 - b. 1
 - c. 3
 - d. 4
- 16. Какая конструкция используется для обработки исключений?
 - a. check-excerpt
 - b. if-else
 - c. try-except
 - d. do-catch
- 17. Какой символ используется для создания списка?
 - a. {}
 - b. ()
 - c. <>
 - d. []
- 18. Что такое функция в Python?
 - a. специальный тип данных
 - b. именованный блок кода для выполнения задачи
 - c. цикл обработки данных
 - d. комментарий в программе
- 19. Какой оператор используется для выполнения условия «иначе»?
 - a. elseif
 - b. else
 - c. another
 - d. other
- 20. Какой файл обычно имеет программа на Python?
 - a. .exe
 - b. .java
 - c. .py
 - d. .html

7.3 Примерные темы рефератов

1. История развития языка программирования Python
2. Основные возможности и сферы применения Python
3. Роль Python в современной разработке программного обеспечения
4. Преимущества и недостатки языка Python
5. Установка и настройка среды разработки Python
6. Структура программы на языке Python
7. Переменные и типы данных в Python
8. Арифметические операции и выражения в Python
9. Логические операции и условные конструкции в Python
10. Циклы и организация повторяющихся вычислений в Python
11. Работа со строками в Python

12. Списки и методы работы со списками в Python
13. Кортежи и словари в Python
14. Множества и операции над множествами в Python
15. Функции и их применение в Python
16. Рекурсия и рекурсивные алгоритмы в Python
17. Области видимости переменных в Python
18. Работа с модулями и библиотеками Python
19. Использование встроенных функций Python
20. Работа с файлами в Python
21. Обработка текстовой информации средствами Python
22. Обработка исключений и отладка программ в Python
23. Основы объектно-ориентированного программирования в Python
24. Классы и объекты в Python
25. Наследование и полиморфизм в Python
26. Основы работы с графическим интерфейсом в Python
27. Использование библиотеки Tkinter для создания интерфейсов
28. Автоматизация рутинных задач средствами Python
29. Python как инструмент анализа данных
30. Использование библиотеки NumPy в Python
31. Использование библиотеки Pandas для обработки данных
32. Основы визуализации данных в Python
33. Использование библиотеки Matplotlib для построения графиков
34. Применение Python в инженерных расчетах
35. Применение Python в веб-разработке
36. Использование Python для разработки чат-ботов
37. Основы работы с базами данных в Python
38. Использование SQL и Python для обработки данных
39. Основы тестирования программ на Python
40. Современные среды разработки для Python
41. Использование Python в искусственном интеллекте
42. Применение Python в машинном обучении
43. Python в области информационной безопасности
44. Применение Python в автоматизации сетевых процессов
45. Использование Python в образовательной деятельности
46. Роль Python в разработке научных проектов
47. Использование Python в нефтегазовой отрасли
48. Python как средство цифровизации производственных процессов
49. Перспективы развития языка программирования Python
50. Этические и профессиональные аспекты деятельности программиста Python

7.4 Примеры ситуационных задач

Практическая задача 1. Расчет показателей учебной успеваемости

Тема: Переменные, ввод данных, арифметические операции.

Цель работы:

сформировать навыки создания простых программ на Python с использованием переменных, ввода данных с клавиатуры и арифметических вычислений.

Задание:

Написать программу, которая запрашивает у пользователя оценки студента по пяти дисциплинам и рассчитывает средний балл.

Исходные данные:

Пользователь вводит 5 оценок по пятибалльной шкале.

Необходимо выполнить:

1. Создать переменные для хранения пяти оценок.
2. Организовать ввод данных с клавиатуры.
3. Рассчитать средний балл студента.
4. Вывести результат на экран.
5. Добавить текстовое пояснение к результату.

Пример результата работы программы:

```
Введите оценку по математике: 5
Введите оценку по информатике: 4
Введите оценку по физике: 5
Введите оценку по истории: 4
Введите оценку по экологии: 5
```

```
Средний балл студента: 4.6
```

Форма отчета:

В отчет включить текст задания, программный код, скриншот результата выполнения программы и краткий вывод.

Практическая задача 2. Определение категории температуры воздуха

Тема: Условные операторы.

Цель работы:

освоить применение условных конструкций `if`, `elif`, `else` для решения простых прикладных задач.

Задание:

Написать программу, которая по введенной температуре воздуха определяет погодную категорию.

Исходные данные:

Пользователь вводит температуру воздуха в градусах Цельсия.

Условия классификации:

- ниже -20°C — «сильный мороз»;
- от -20 до 0°C — «холодно»;
- от 1 до 15°C — «прохладно»;
- от 16 до 25°C — «комфортная температура»;
- выше 25°C — «жарко».

Необходимо выполнить:

1. Организовать ввод температуры.
2. Использовать условные операторы для выбора категории.
3. Вывести результат на экран.
4. Проверить работу программы на разных значениях температуры.

Пример результата работы программы:

Введите температуру воздуха: -25
Погодная категория: сильный мороз

Форма отчета:

В отчет включить код программы, 3 примера запуска программы с разными исходными данными и вывод по работе.

Практическая задача 3. Расчет суммы чисел в заданном диапазоне

Тема: Циклы в Python.

Цель работы:

сформировать навыки использования циклов для выполнения повторяющихся вычислений.

Задание:

Написать программу, которая рассчитывает сумму всех целых чисел от 1 до числа, введенного пользователем.

Исходные данные:

Пользователь вводит положительное целое число n .

Необходимо выполнить:

1. Организовать ввод числа n .
2. Использовать цикл `for` или `while`.

3. Рассчитать сумму чисел от 1 до n .
4. Вывести результат на экран.
5. Добавить проверку, что пользователь ввел положительное число.

Пример результата работы программы:

Введите положительное целое число: 10
Сумма чисел от 1 до 10 равна 55

Форма отчета:

В отчет включить текст задания, код программы, результат выполнения и краткое пояснение используемого цикла.

Практическая задача 4. Обработка списка расходов

Тема: Списки и базовая обработка данных.

Цель работы:

освоить работу со списками, вводом числовых данных и простейшими вычислениями по набору значений.

Задание:

Написать программу, которая анализирует расходы студента за неделю.

Исходные данные:

Пользователь вводит расходы за 7 дней недели.

Необходимо выполнить:

1. Создать список для хранения расходов.
2. Организовать ввод расходов за каждый день.
3. Рассчитать общую сумму расходов за неделю.
4. Найти максимальный и минимальный расход.
5. Рассчитать средний расход за день.
6. Вывести результаты на экран.

Пример результата работы программы:

Общие расходы за неделю: 5250 руб.
Максимальный расход за день: 1200 руб.
Минимальный расход за день: 400 руб.
Средний расход за день: 750 руб.

Форма отчета:

В отчет включить программный код, скриншот результата и вывод о возможностях использования списков в Python.

Практическая задача 5. Работа с текстовым файлом

Тема: Файлы и обработка текстовой информации.

Цель работы:

сформировать начальные навыки чтения и записи текстовых файлов средствами Python.

Задание:

Написать программу, которая создает текстовый файл с информацией о студенте, затем считывает этот файл и выводит его содержимое на экран.

Исходные данные:

Фамилия, имя, группа, направление подготовки и любимый язык программирования.

Необходимо выполнить:

1. Создать текстовый файл `student.txt`.
2. Записать в файл информацию о студенте.
3. Заккрыть файл после записи.
4. Открыть файл для чтения.
5. Считать данные из файла.
6. Вывести содержимое файла на экран.

Пример содержания файла:

Фамилия: Иванов

Имя: Иван

Группа: Б-101

Направление подготовки: Информатика и вычислительная техника

Любимый язык программирования: Python

Форма отчета:

В отчет включить код программы, содержимое созданного файла, скриншот результата выполнения и краткий вывод.

7.5 Примерный список вопросов, задаваемых на зачете

1. Что такое язык программирования Python и в каких областях он применяется?
2. Какие основные преимущества языка Python?
3. Что такое алгоритм и программа?
4. Какие этапы включает процесс разработки программы?
5. Как выполняется установка и настройка Python?
6. Какие существуют среды разработки для Python?
7. Что представляет собой структура программы на Python?
8. Какие правила используются при создании имен переменных?
9. Какие основные типы данных существуют в Python?
10. Чем отличаются целочисленные и вещественные типы данных?
11. Что такое строковый тип данных и какие операции со строками существуют?

12. Какие арифметические операции поддерживаются в Python?
13. Какие логические операции используются в Python?
14. Что такое оператор присваивания?
15. Для чего используются функции input() и print()?
16. Какие условные конструкции используются в Python?
17. Как работает оператор if?
18. В чем особенности конструкции if-elif-else?
19. Какие циклы используются в Python?
20. В чем различие между циклами for и while?
21. Для чего используются операторы break и continue?
22. Что такое вложенные циклы?
23. Что представляет собой список в Python?
24. Какие основные операции выполняются со списками?
25. Чем список отличается от кортежа?
26. Что такое словарь в Python?
27. Какие методы используются для работы со словарями?
28. Что такое множество в Python?
29. Для чего применяются функции в Python?
30. Как выполняется создание пользовательской функции?
31. Что такое параметры и аргументы функции?
32. Что означает оператор return?
33. Что такое область видимости переменных?
34. Что такое рекурсия?
35. Для чего используются модули в Python?
36. Какие стандартные библиотеки Python вы знаете?
37. Как выполняется подключение модуля в Python?
38. Что такое обработка исключений?
39. Для чего используется конструкция try-except?
40. Как осуществляется работа с текстовыми файлами в Python?
41. Какие режимы открытия файлов существуют в Python?
42. Что такое объектно-ориентированное программирование?
43. Что такое класс и объект в Python?
44. Что такое методы класса?
45. Что представляет собой наследование в Python?
46. Для чего используются библиотеки NumPy и Pandas?
47. Какие возможности предоставляет Python для визуализации данных?
48. Как Python применяется в автоматизации задач?
49. Какие основные этапы тестирования программ существуют?
50. Каковы перспективы развития и применения Python в современных цифровых технологиях?

8 Материально-техническое и учебно-методическое обеспечение дисциплины

8.1 Перечень учебной литературы

Наименование печатных и (или) электронных учебных изданий, методические издания, периодические издания по всем входящим в реализуемую образовательную программу учебным предметам, курсам, дисциплинам (модулям) <i>в соответствии с рабочими программами дисциплин, модулей, практик</i>		Количество экземпляров в	Обеспеченность студентов учебной литературой (экземпляров на одного студента)
Электронные учебные издания, имеющиеся в	Чернышев, Станислав Андреевич. Основы программирования на Python : учебник для вузов / С. А. Чернышев. - 2-е изд., пер. и доп. - Москва : Юрайт,	1	1

электронном каталоге электронно-библиотечной системы	2026. - 349 с.		
	Воробьев, Г. А. Основы программирования на Python : учебно-методическое пособие / Г. А. Воробьев. - Липецк : Липецкий ГПУ, 2022. - 89 с.	1	1
	Драпак, К. А. Основы программирования на Python : учебное пособие / К. А. Драпак, М. Ю. Ветлицын, А. М. Макаров. - Волгоград : ВолгГТУ, 2024. - 160 с.	1	1

8.2 Современные профессиональные базы данных, информационные справочные и электронно-библиотечные системы

№	Ссылка на информационный ресурс	Наименование ресурса в электронной форме	Доступность
Электронно-библиотечные системы			
1	https://urait.ru	Образовательная платформа Юрайт	Авторизованный доступ
2	http://www.iprbookshop.ru	ЭБС IPR SMART	Авторизованный доступ
3	http://znanium.com	ЭБС «Znaniy»	Авторизованный доступ
4	https://e.lanbook.com	ЭБС «Лань»	Авторизованный доступ
Информационные справочные системы			
5	http://garant.ugrasu.ru/	СПС Гарант	Авторизованный доступ
6	http://www.consultant.ru/	СПС КонсультантПлюс	Авторизованный доступ
Профессиональные базы данных			
7	https://ugrasu.dc-ipk.ru/docs/	Информационно-справочная система Техэксперт	Авторизованный
8	http://elibrary.ru	Научная электронная библиотека eLIBRARY.RU	Авторизованный доступ

8.3 Перечень лицензионного и свободно распространяемого программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине, в том числе отечественного производства

Python;
Система ГАРАНТ;

8.4 Материально-техническое обеспечение дисциплины

8.4.1 Учебная аудитория лекционного типа

компьютер/ноутбук, проектор, экран, учебная мебель, учебная доска

8.4.2 Учебная аудитория для проведения практических занятий
учебная мебель, учебная доска

8.4.3 Учебная аудитория для самостоятельной работы
учебная мебель, компьютеры с выходом в интернет и доступом к электронной информационно-образовательной среде

8.4.4 Компьютерный класс
учебная мебель, учебная доска, компьютеры с доступом в Интернет

