

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
*ФАКУЛЬТЕТ ИНФОРМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ*

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

### **Программирование на языке Python**

Кафедра информационных систем и технологии программирования

Образовательная программа  
**09.03.04 Программная инженерия**

Направленность (профиль) программы  
Разработка программно-информационных систем  
Программное обеспечение вычислительной техники  
и автоматизированных систем

Уровень высшего образования  
**Бакалавриат**

Форма обучения  
**Очная**


Статус дисциплины: входит в часть ОПОП, формируемую участниками  
образовательных отношений

Махачкала, 2022

Рабочая программа дисциплины «Программирование на языке Python» составлена в 2022 году в соответствии с требованиями ФГОС ВО – бакалавриат по направлению подготовки 09.03.04 Программная инженерия от «19» 09. 2017г. № 920.

Разработчик(и): кафедра информационных систем и технологий программирования, ст. преподаватель Мирзабеков Я.М., к.э.н. Гаджиев Н.К.

Рабочая программа дисциплины одобрена:  
на заседании кафедры ИСиТП от «01» марта 2022г., протокол № 8

Зав. кафедрой  Исмиханов З.Н.  
(подпись)

на заседании Методической комиссии факультета ИиИТ  
от «17» марта 2022г., протокол № 7

Председатель  Бакмаев А.Ш.  
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением «31» марта 2022г.

Начальник УМУ  Гасангаджиева А.Г.

## Аннотация рабочей программы дисциплины

Дисциплина «Программирование на языке Python» входит в часть, формируемую участниками образовательных отношений ОПОП бакалавриата по направлению 09.03.04 Программная инженерия

Дисциплина реализуется на Информатики и информационных технологий кафедрой Информационных систем и технологий программирования.

Содержание дисциплины охватывает круг вопросов, связанных с изучением современных технологий и методов программирования на языке Python, механизмов доступа к базам данных и работы с ними, приобретением практических навыков использования современных инструментальных средств для разработки, отладки и тестирования создаваемых прикладных программ.

Дисциплина нацелена на формирование следующих компетенций выпускника: профессиональных – ПК-2, ПК-6.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, лабораторные занятия и самостоятельная работа.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости в форме контрольной работы и промежуточный контроль в форме зачета.

Объем дисциплины 7 зачетных единиц, в том числе в академических часах по видам учебных занятий

### Объем дисциплины в очной форме

Семестр	Учебные занятия							СРС, в том числе экзамен	Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)
	в том числе:								
	всего	Контактная работа обучающихся с преподавателем					консультации		
		всего	Лекции	Лабораторные занятия	Практические занятия	КСР			
3	144	54	18	36			90	Зачет	
4	108	50	16	34			58	Зачет	
итого	252	104	34	70			148		

## 1. Цели освоения дисциплины

Целью освоения дисциплины «Программирование на языке Python» является знакомство с возможностями языка программирования Python.

## 2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Программирование на языке Python» входит в обязательную часть ОПОП бакалавриата по направлению 09.03.04 Программная инженерия.

Дисциплина «Программирование на языке Python» призвана содействовать знакомству студентов с языками и методами программирования и является курсом, для освоения которого необходимы теоретические знания и практические навыки, полученные по дисциплинам «Базы данных», «Объектно-ориентированное программирование», «Информатика и программирование», «Дискретная математика».

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения и процедура освоения).

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии с ОПОП)	Планируемые результаты обучения Процедура освоения	Процедура освоения
ПК-2 Способность разрабатывать и адаптировать прикладное программное обеспечение.	<b>ПК- 2.1.</b> Знает принципы организации проектирования и содержание этапов процесса разработки прикладных программ. <b>ПК- 2.2.</b> Умеет разрабатывать и отлаживать эффективные алгоритмы и программы с использованием современных технологий программирования. <b>ПК- 2.3.</b> Владеет навыками проектирования и разработки прикладного программного обеспечения с использованием современных технологий программирования	<b>Знать:</b> принципы разработки программного обеспечения, концепции и понятия объектно-ориентированного подхода к программированию, механизмы его реализации в языке программирования. <b>Уметь:</b> создавать приложения на различных языках программирования, использовать основные принципы объектно-ориентированного подхода при написании программ; проектировать и реализовывать программы со сложной иерархией классов и объектов. <b>Владеть:</b> навыками анализа поставленных задач, проектирования и разработки приложений, приемами разработки программных комплексов для решения прикладных	Конспектирование и проработка лекционного материала. Участие в лабораторных занятиях. Самостоятельная работа.

		задач, методами использования современных технологий программирования, тестирования и документирования программных комплексов.	
<p><b>ПК-6</b> Способность программировать приложения и создавать программные прототипы решения прикладных задач.</p>	<p><b>ПК- 6.1.</b> Знает основные сведения о методах и способах построения эффективных алгоритмов для решения прикладных задач</p> <p><b>ПК- 6.2.</b> Умеет создавать программные прототипы решения задач предметной области.</p> <p><b>ПК- 6.3.</b> Владеет практическими навыками разработки программных прототипов решения прикладных задач</p>	<p><b>Знать:</b> проблемы и процессы анализа предметной области программных решений современные подходы анализа предметной области программных решений.</p> <p><b>Уметь:</b> разрабатывать программные приложения для предметной области; производить анализ сложности алгоритма и находить пути упрощения полученных алгоритмов</p> <p><b>Владеть:</b> практическими навыками использования языков программирования для создания программные прототипов решения прикладных задач; основные и наиболее популярные программные продукты, позволяющие проектировать и разрабатывать алгоритмы.</p>	<p>Конспектирование и проработка лекционного материала. Участие в лабораторных занятиях. Самостоятельная работа.</p>

#### 4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 7 зачетных единиц, 252 академических часа.

4.2. Структура дисциплины.

##### 4.2.1. Структура дисциплины в очной форме

№ п/п	Разделы и темы дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости и промежуточной аттестации
			Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа	
Модуль 1.							
1	Python. Первые шаги.	3	2		4	8	Прием лабораторных и практических работ
2	Управляющие инструкции.	3	2		6	14	Прием лабораторных и практических работ
	<i>Итого по модулю 1:</i>		4		10	22	Контрольная работа
Модуль 2.							
1	Коллекции в Python.	3	4		10	22	Прием лабораторных и практических работ
	<i>Итого по модулю 2:</i>		4		10	22	Контрольная работа
Модуль 3.							
1	Функции в языке Python.		4				Прием лабораторных и практических работ
2	Библиотеки Python	3	4		10	18	Прием лабораторных и практических работ
	<i>Итого по модулю 3:</i>		8		10	18	Контрольная работа
Модуль 4.							
1	Работа с файлами в Python.	3	2		6	28	Прием лабораторных и практических работ
	<i>Итого по модулю 4:</i>		2		6	28	Зачет
	<b>Итого за 3 сем</b>		<b>18</b>		<b>36</b>	<b>90</b>	
Модуль 1.							
1	Основы ООП в Python.	4	2		4	10	Прием лабораторных и практических работ
2	Графический интерфейс программ в Python.	4	4		6	10	Прием лабораторных и практических работ

	<i>Итого по модулю 1:</i>		6		10	20	Контрольная работа
	Модуль 2.						
1	Работа с БД SQLite в Python	4	2		6	8	Прием лабораторных и практических работ
2	Создание простейших игр в Python	4	4		8	8	Прием лабораторных и практических работ
	<i>Итого по модулю 2:</i>		6		14	16	Контрольная работа
	Модуль 3.						
	Создание web-приложений в Python	4	4		10	22	Прием лабораторных и практических работ
	<i>Итого по модулю 3:</i>		4		10	22	Зачет
	<b>Итого на 4 сем</b>		<b>16</b>		<b>34</b>	<b>58</b>	
	<b>ИТОГО:</b>		<b>34</b>		<b>70</b>	<b>148</b>	

### 4.3. Содержание дисциплины, структурированное по темам (разделам).

#### 4.3.1 Содержание лекционных занятий по дисциплине

#### 3 семестр

#### Первый модуль

##### Тема 1. Python. Первые шаги.

Понятие «алгоритм» и «программа». Знакомство с Python. Вывод информации на экран. Ввод данных в программу.

Синтаксис языка Python. Структура программы. Переменные величины. Инструкция присваивания.

##### Тема 2. Управляющие инструкции.

Условный оператор if-elif-else. Функция range(). Операторы цикла for и while. Операторы break и continue. Обработка исключительных ситуаций.

#### Второй модуль

##### Тема 1. Коллекции в Python.

Объекты типа set. Операции над множеством. Операции над двумя множествами. Сравнение множеств.

Строка как коллекция. Индексация в строках. Перебор элементов строки. Хранение текстов в памяти компьютера. Срезы строк.

Списки. Создание списков. Индексация в списках. Добавление элемента в список. Перебор элементов списка. Срезы списков. Удаление элементов. Списки и массивы. Методы split и join. Списочные выражения. Использование списочных выражений в аргументах методов split и join. Считывание значений, введенных одной строкой. Вложенные списки.

Двумерные вложенные списки (матрицы). Создание двумерного списка. Перебор элементов двумерного списка. Вывод списка на экран. Матрицы. Кортежи. Присваивание кортежей. Преобразования между коллекциями. Знакомство со словарями. Создание словаря. Обращение к элементу словаря. Добавление и удаление элементов. Проверка наличия элемента в словаре. Нестроковые ключи. Методы словарей. Допустимые типы ключей

## **Третий модуль**

### **Тема 1. Функции в языке Python.**

Создание функции. Значения аргументов по умолчанию. Функция как аргумент. Рекурсия. Лямбда-функции. Локальные и глобальные переменные. Вложенные функции. Функция как результат функции.

### **Тема 2. Библиотеки Python**

Библиотеки как наследие. Репозиторий PyPI. Встроенные модули. Модуль math. Модуль random. Модуль datetime. Модуль pprint. Обработка изображений. Растровые изображения. PIL. Установка библиотек. Модельный пример. Создание изображений и рисование. Работа с документами. Создание текстового документа. Работа с документом как с шаблоном. Создание презентаций. Создание таблиц Excel

## **Четвертый модуль**

### **Тема 1. Работа с файлами в Python.**

Общие сведения о файлах. Перевод строки. Файловый путь. Относительные и абсолютные пути. Кодировки файлов. Типичные операции с файлами. Чтение файла. Запись в файл. Закрытие файлов.

## **4 семестр Первый модуль**

### **Тема 1. Основы ООП в Python.**

Основные понятия. Создание классов. Методы классов. Инициализация экземпляров класса. Полиморфизм. Проверка типа объекта. Иерархия классов. Механизм наследования.

### **Тема 2. Графический интерфейс программ в Python.**

Графический интерфейс. Установка и настройка. Первые шаги в PyQt. Кто отправил сигнал. Открытие других форм. Установка QtDesigner и первый запуск. Подключение дизайна к программе. Размещение виджетов.



## **Второй модуль**

### **Тема 1. Работа с БД SQLite в Python**

Введение в базы данных. Основы SQL. Работаем с SQLite базой данных из Python. Возможности PyQT по работе с базами данных. Модификация из Python.

### **Тема 2. Создание простейших игр в Python.**

Установка pygame. Hello, Pygame. Рисование. Цвета. Система координат. Surface, Rect. Функции рисования. Время в PyGame. События. События по таймеру. Холст (Surface). Что такое клетчатое поле? Создание класса. Реакция поля на события мыши. Изображения. Спрайты. Наследование спрайтов. Пересечение по прямоугольнику и окружности. Пересечение по маске.

## **Третий модуль**

### **Тема 1. Создание web-приложений в Python**

Инструменты для разработки веб-приложений. Первая страница на Flask. Статический контент. Передача параметров в адресной строке. Обработка форм. Шаблоны. Условия в шаблонах. Циклы в шаблонах. Создание переменных в шаблоне. Наследование шаблонов. Знакомство с Flask-WTF. Базы данных и ORM. Установка. Постановка задачи для первого примера. Начальный каркас приложения. Таблица с пользователями. Таблица с новостями. Взаимодействие с базой данных. Небольшая шпаргалка по filter. Отображение публичных записей. Регистрация пользователей.

## **4.3.2 Содержание лабораторно-практических занятий по дисциплине**

### **3 семестр**

#### **Первый модуль**

**Тема 1. Python. Первые шаги.**

**Тема 2. Управляющие инструкции.**

#### **Второй модуль**

**Тема 1. Коллекции в Python.**

#### **Третий модуль**

**Тема 1. Функции в языке Python.**

**Тема 2. Библиотеки Python**

## Четвертый модуль

### Тема 1. Работа с файлами в Python.

#### 4 семестр Первый модуль

### Тема 1. Основы ООП в Python.

### Тема 2. Графический интерфейс программ в Python.

#### Второй модуль

### Тема 1. Работа с БД SQLite в Python

### Тема 2. Создание простейших игр в Python.

#### Третий модуль

### Тема 1. Создание web-приложений в Python

## 5. Образовательные технологии

В процессе преподавания дисциплины «Языки и методы программирования» применяются следующие образовательные технологии: лекционно-зачетная система обучения, технология развития критического мышления, изложение теоретических положений в ходе лекционных занятий с применением современного интерактивного презентационного оборудования, проведение групповых дискуссий. Основная литература, презентации и конспекты лекций предоставляются студентам в электронном виде.

## 6. Учебно-методическое обеспечение самостоятельной работы студентов.

Самостоятельная работа студентов складывается из:

- проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к отчетам по лабораторным работам;
- подготовки к сдаче промежуточных форм контроля.

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
---	----------------------------	--------------	---------------------------------

1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.2, 8, 9 данного документа
2.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.2, 8, 9 данного документа
3.	Подготовка к отчетам по практическим работам.	Проверка выполнения работ, опрос по теме работы.	См. разделы 7.2, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю.	См. разделы 7.2, 8, 9 данного документа

## **7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

### **7.1. Типовые контрольные задания**

#### **Вопросы к зачету**

1. Переменные и типы данных языка Python. Преобразование типов
2. Ввод-вывод данных, оператор присваивания
3. Условный оператор
4. Функция range()
5. Операторы цикла
6. Функции
7. Лямбда-функции
8. Функции с переменным числом аргументов
9. Списки
10. Вложенные списки
11. Методы split и join. Списочные выражения
12. Срезы (строки, списки)
13. Кортежи
14. Множества
15. Словари
16. Файлы
17. Библиотеки Python. Модули math и random
18. Классы в Python. Наследование классов.
19. Создание приложений с графическим интерфейсом. Модуль PyQt5.

20. БД SQLite в Python
21. Модуль PyGame. Создание игр.
22. Модуль Flask. Создание web-приложений.

7.2. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий - 10 баллов,
- участие на практических занятиях - 20 баллов,
- выполнение лабораторных заданий - 30 баллов,
- выполнение домашних (аудиторных) контрольных работ - 40 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 40 баллов,
- письменная контрольная работа - 60 баллов.

## **8. Учебно-методическое обеспечение дисциплины.**

а) основная литература:

1. Сузи Р.А. Язык программирования Python: курс / Р.А. Сузи. - 2-е изд., испр. - Москва: Интернет-Университет Информационных Технологий, 2007. - 327 с. - (Основы информационных технологий). - ISBN 978-5-9556-0109-0; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233288>
2. Северенс, Ч. Введение в программирование на Python / Ч. Северенс. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 231 с.: схем., ил.; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429184>

б) дополнительная литература:

1. Буйначев, С.К. Основы программирования на языке Python: учебное пособие / С.К. Буйначев, Н.Ю. Боклаг; Министерство образования и науки Российской Федерации, Уральский федеральный университет им. первого Президента России Б. Н. Ельцина. - Екатеринбург: Издательство Уральского университета, 2014. - 92 с.: табл., ил. - Библиогр. в кн. - ISBN 978-5-7996-1198-9; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=275962>
2. Sweigart, A. Разработка компьютерных игр на языке Python / A. Sweigart. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 505 с.: ил.; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429009>

## **9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.**

1) eLIBRARY.RU [Электронный ресурс]: электронная библиотека / Науч. электрон. б-ка. — Москва, 1999 —. Режим доступа:

<http://elibrary.ru/defaultx.asp> – Яз. рус., англ.

2) Moodle [Электронный ресурс]: система виртуального обучения: [база данных] / Даг. гос. ун-т. – Махачкала, г. – Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. – URL: <http://edu.dgu.ru/>

3) Электронный каталог НБ ДГУ [Электронный ресурс]: база данных содержит сведения о всех видах лит, поступающих в фонд НБ ДГУ/Дагестанский гос. ун-т. – Махачкала, 2010 – Режим доступа: <http://elib.dgu.ru>, свободный.

4) <https://pythontutor.ru/> - Бесплатный курс по программированию с нуля.

## **10. Методические указания для обучающихся по освоению дисциплины.**

При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.**

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение: Microsoft Windows, Python 3.10, Pycharm 2021 Community. Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с необходимым программным обеспечением.