

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Петербургский государственный университет путей сообщения
Императора Александра I»
(ФГБОУ ВО ПГУПС)

Кафедра «Информационные и вычислительные системы»

РАБОЧАЯ ПРОГРАММА

дисциплины

**Б1.В.7 «ПРОГРАММИРОВАНИЕ
НА ЯЗЫКАХ ВЫСОКОГО УРОВНЯ (PYTHON)»**

для направления подготовки

09.03.01 «Информатика и вычислительная техника»

по профилю

**«Программное обеспечение вычислительной техники
и автоматизированных систем»**

Форма обучения – очная

Санкт-Петербург
2025

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа рассмотрена и утверждена на заседании кафедры
«Информационные и вычислительные системы»
Протокол №4 от « 23 » декабря 2024 г.

Заведующий кафедрой
«Информационные и вычислительные системы» _____ С.Г. Ермаков
« 23 » декабря 2024 г.

СОГЛАСОВАНО

Руководитель ОПОП ВО _____ С.Г. Ермаков
« 23 » декабря 2024 г.

1. Цели и задачи дисциплины

Рабочая программа дисциплины «Программирование на языках высокого уровня (Python) (Б1.В.7) (далее – дисциплина) составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.01 «Информатика и вычислительная техника» (далее – ФГОС ВО), утвержденного 19 сентября 2017г., приказ Минобрнауки России № 917.

Целью изучения дисциплины является формирование у обучающихся компетенций, необходимых для эффективного использования языка программирования Python в профессиональной деятельности, включая разработку программного обеспечения, проектирование баз данных и автоматизацию задач.

Для достижения поставленной цели решаются следующие задачи:

- изучение базовых конструкций языка Python и принципов их использования;
- развитие навыков алгоритмического мышления и умения разрабатывать эффективные алгоритмы;
- изучение методов работы с различными типами данных и структурами данных, а также принципов объектно-ориентированного программирования и их применения в Python;
- формирование навыков работы с основными библиотеками и инструментами Python, такими как NumPy, Pandas, Matplotlib;
- формирование навыков разработки, отладки и тестирования программного обеспечения;
- формирование навыков работы в команде и применения современных методологий разработки программного обеспечения;
- освоение методов проектирования программного обеспечения и баз данных;
- развитие навыков оценки и обоснования рекомендуемых решений;
- применение нормативно-правовой документации в профессиональной деятельности.

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в программе бакалавриата индикаторами достижения компетенций

Планируемыми результатами обучения по дисциплине является формирование у обучающихся компетенций (части компетенций). Сформированность компетенций (части компетенции) оценивается с помощью индикаторов достижения компетенций, приведенных в таблице 2.1.

Таблица 2.1.

Индикаторы достижения компетенций	Результаты обучения по дисциплине
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	
УК-2.1.1 Знает виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы,	Обучающийся <i>знает</i> : <ul style="list-style-type: none">– виды ресурсов и ограничений для решения профессиональных задач;– основные методы оценки разных способов решения задач;– действующее законодательство и правовые нормы, регулирующие профессиональную деятельность

регулирующие профессиональную деятельность.	
УК-2.2.1 Умеет проводить анализ поставленной цели и формулировать задачи, которые необходимо решить для ее достижения; анализировать альтернативные варианты для достижения намеченных результатов; использовать нормативно-правовую документацию в сфере профессиональной деятельности	Обучающийся <i>умеет</i> : <ul style="list-style-type: none"> – проводить анализ поставленной цели и формулировать задачи, которые необходимо решить для ее достижения; – анализировать альтернативные варианты для достижения намеченных результатов; – использовать нормативно-правовую документацию в сфере профессиональной деятельности
УК-2.3.1 Владеет методиками разработки цели и задач проекта; методами оценки потребности в ресурсах, продолжительности и стоимости проекта; навыками работы с нормативно-правовой документацией	Обучающийся <i>владеет</i> : <ul style="list-style-type: none"> – методиками разработки цели и задач проекта; – методами оценки потребности в ресурсах, продолжительности и стоимости проекта; – навыками работы с нормативно-правовой документацией
ПК-2. Способен разрабатывать технические спецификации на программные компоненты и их взаимодействие	
ПК-2.1.1 Знает методы и средства проектирования программного обеспечения	Обучающийся <i>знает</i> : <ul style="list-style-type: none"> – методы и средства проектирования программного обеспечения
ПК-2.1.2 Знает методы и средства проектирования баз данных	Обучающийся <i>знает</i> : <ul style="list-style-type: none"> – методы и средства проектирования баз данных
ПК-2.2.1 Умеет вырабатывать варианты реализации программного обеспечения	Обучающийся <i>умеет</i> : <ul style="list-style-type: none"> – вырабатывать варианты реализации программного обеспечения
ПК-2.2.2 Умеет проводить оценку и обоснование рекомендуемых решений	Обучающийся <i>умеет</i> : <ul style="list-style-type: none"> – проводить оценку и обоснование рекомендуемых решений
ПК-2.3.1 Имеет навыки разработки и согласование технических спецификаций на программные компоненты и их взаимодействие с архитектором программного обеспечения	Обучающийся <i>имеет навыки</i> : <ul style="list-style-type: none"> – разработки и согласования технических спецификаций на программные компоненты и их взаимодействие с архитектором программного обеспечения

3. Место дисциплины в структуре основной профессиональной образовательной программы

Дисциплина относится к вариативной части блока «Дисциплины (модули)».

4. Объем дисциплины и виды учебной работы

Таблица 4.1.

Вид учебной работы	Всего часов	Семестр
		3
Контактная работа (по видам учебных занятий) В том числе: – лекции (Л) – лабораторные работы (ЛР)	64 32 32	64 32 32
Самостоятельная работа (СРС)	44	44
Контроль	36	36
Форма контроля (промежуточной аттестации)	Э	Э
Общая трудоемкость: час / з.е.	144/4	144/4

5. Структура и содержание дисциплины

5.1. Разделы дисциплины и содержание рассматриваемых вопросов

Таблица 5.1.

№ п/п	Наименование раздела дисциплины	Содержание раздела	Индикаторы достижения компетенций
1	Основы языка Python	Лекция 1. Введение в язык программирования Python. История и особенности Python. Установка и настройка среды разработки. Основы синтаксиса Python. Выполнение простых программ. Лабораторная работа №1. Установка и настройка Python. Установка Python и среды разработки Выполнение простых программ. Самостоятельная работа Повторение лекционного материала, выполнение заданий к ЛР №1	УК-2.1.1 УК-2.2.1 УК-2.3.1 ПК-2.1.1, ПК-2.1.2, ПК-2.2.1, ПК-2.2.2, ПК-2.3.1
2	Базовые конструкции и управление потоком	Лекция 2. Базовые конструкции языка Python. Типы данных: числа, строки, списки, кортежи, словари, множества. Операторы: арифметические, логические, сравнения. Управляющие конструкции: if, for, while. Лабораторная работа №2. Основные конструкции языка Python. Работа с основными типами данных. Использование управляющих конструкций.	

		<u>Самостоятельная работа</u> Повторение лекционного материала, выполнение заданий к ЛР №2	
3	Функции и модули	<p><u>Лекция 3.</u> Функции и модули в Python. Определение и использование функций. Аргументы и возвращаемые значения. Работа с модулями и пакетами. Введение в стандартную библиотеку Python.</p> <p><u>Лабораторная работа №3.</u> Функции и модули. Создание и использование функций. Импорт и использование модулей.</p> <p><u>Самостоятельная работа</u> Повторение лекционного материала, выполнение заданий к ЛР №3.</p>	
4	Структуры данных и файловые операции	<p><u>Лекция 4.</u> Структуры данных в Python. Списки и их методы. Кортежи и их использование. Словари и множества. Вложенные структуры данных.</p> <p><u>Лекция 5.</u> Работа с файлами. Открытие, чтение и запись файлов. Работа с текстовыми и бинарными файлами. Исключения и их обработка.</p> <p><u>Лабораторная работа №4.</u> Работа со структурами данных. Операции со списками и кортежами. Использование словарей и множеств.</p> <p><u>Лабораторная работа №5.</u> Файловые операции. Открытие и чтение файлов. Запись данных в файлы. Обработка исключений.</p> <p><u>Самостоятельная работа</u> Повторение лекционного материала, выполнение заданий к ЛР №4, 5.</p>	
5	Объектно-ориентированное программирование (ООП)	<p><u>Лекция 6.</u> Основы ООП в Python. Основные концепции ООП: классы и объекты. Наследование, инкапсуляция и полиморфизм. Магические методы и перегрузка операторов.</p> <p><u>Лабораторная работа №6.</u> ООП в Python. Создание и использование классов и объектов. Реализация наследования и полиморфизма.</p> <p><u>Самостоятельная работа</u> Повторение лекционного материала, Выполнение заданий к ЛР №6.</p>	
6	Расширенные возможности языка Python	<u>Лекция 7.</u> Генераторы и итераторы. Определение и использование	

		<p>генераторов. Итераторы и протокол итерации.</p> <p>Лекция 8. Декораторы и контекстные менеджеры. Создание и применение декораторов. Встроенные декораторы. Контекстные менеджеры и их использование.</p> <p>Лабораторная работа №7. Генераторы и итераторы. Создание и использование генераторов и итераторов.</p> <p>Лабораторная работа №8. Декораторы и контекстные менеджеры. Создание декораторов и контекстных менеджеров.</p> <p>Самостоятельная работа. Повторение лекционного материала, выполнение заданий к ЛР №7, 8.</p>	
7	Работа с библиотеками и научными вычислениями	<p>Лекция 9. Стандартные библиотеки Python. Математические библиотеки: math, random. Работа с датой и временем: datetime. Регулярные выражения: re. Модуль collections.</p> <p>Лекция 10. Введение в библиотеки для научных вычислений. Библиотека NumPy: массивы и основные операции. Библиотека Pandas: работа с таблицами и данными. Основы визуализации данных с Matplotlib.</p> <p>Лабораторная работа №9. Стандартные библиотеки Python. Использование библиотек math, random, datetime, re, collections.</p> <p>Лабораторная работа №10. Научные вычисления с Python. Работа с массивами в NumPy. Анализ данных с Pandas. Визуализация данных с Matplotlib.</p> <p>Самостоятельная работа. Повторение лекционного материала, выполнение заданий к ЛР№9, 10.</p>	<p>УК-2.1.1 УК-2.2.1 УК-2.3.1 ПК-2.1.1, ПК-2.1.2, ПК-2.2.1, ПК-2.2.2, ПК-2.3.1</p>
8	Разработка, тестирование и автоматизация	<p>Лекция 11. Основы разработки программного обеспечения. Принципы разработки программного обеспечения. Инструменты отладки в Python. Тестирование кода: unittest, pytest. Применение систем контроля версий (Git).</p> <p>Лекция 12. Введение в веб-разработку с Python. Основы веб-программирования. Веб-фреймворки: Flask, Django. Создание простых веб-приложений.</p>	

		<p><u>Лекция 13.</u> Автоматизация задач с Python. Скрипты для автоматизации рутинных задач. Работа с файлами и папками. Автоматизация работы с веб-страницами (Selenium).</p> <p><u>Лекция 14.</u> Python для анализа данных. Основные методы анализа данных. Использование библиотеки Pandas для анализа данных. Визуализация данных с помощью Matplotlib и Seaborn.</p> <p><u>Лекция 15.</u> Организация проектной работы и работа в команде. Организация проектной работы. Использование систем контроля версий в команде. Методологии разработки ПО: Agile, Scrum.</p> <p><u>Лекция 16.</u> Примеры успешных проектов на Python. Обзор успешных проектов на Python. Анализ лучших практик и подходов.</p> <p><u>Лабораторная работа №11.</u> Разработка и тестирование программ. Написание и отладка программного кода. Написание тестов с unittest и pytest. Использование Git для контроля версий.</p> <p><u>Лабораторная работа №12.</u> Веб-разработка с Python. Создание простого веб-приложения с Flask. Введение в Django.</p> <p><u>Лабораторная работа №13.</u> Автоматизация задач. Написание скриптов для автоматизации задач. Автоматизация веб-страниц с Selenium.</p> <p><u>Лабораторная работа №14.</u> Анализ данных с Python. Анализ данных с использованием Pandas. Визуализация данных с Matplotlib и Seaborn.</p> <p><u>Лабораторная работа №15.</u> Проектная работа и работа в команде. Организация командной работы над проектом. Использование Git и других инструментов для совместной разработки.</p> <p><u>Лабораторная работа №16.</u> Примеры успешных проектов. Изучение и реализация элементов успешных проектов на Python.</p> <p><u>Самостоятельная работа.</u> Повторение лекционного материала, выполнение заданий к ЛР №11, 12, 13, 14, 15, 16.</p>	
--	--	---	--

5.2. Разделы дисциплины и виды занятий

Таблица 5.2.

№ п/п	Наименование раздела дисциплины	Л	ЛР	СРС	Всего
1	Основы языка Python	2	2	2	6
2	Базовые конструкции и управление потоком	2	2	4	8
3	Функции и модули	2	2	6	10
4	Структуры данных и файловые операции	4	4	4	12
5	Объектно-ориентированное программирование (ООП)	2	2	4	8
6	Расширенные возможности языка Python	4	4	4	12
7	Работа с библиотеками и научными вычислениями	4	4	6	14
8	Разработка, тестирование и автоматизация	12	12	14	38
	Итого	32	32	44	108
	Контроль	36			144

6. Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине

Оценочные материалы по дисциплине являются неотъемлемой частью рабочей программы и представлены отдельным документом, рассмотренным на заседании кафедры и утвержденным заведующим кафедрой.

7. Методические указания для обучающихся по освоению дисциплины

Порядок изучения дисциплины, следующий:

1. Освоение разделов дисциплины производится в порядке, приведенном в разделе 5 «Содержание и структура дисциплины». Обучающийся должен освоить все разделы дисциплины, используя методические материалы дисциплины, а также учебно-методическое обеспечение, приведенное в разделе 8 рабочей программы.

2. Для формирования компетенций обучающийся должен представить выполненные типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, предусмотренные текущим контролем успеваемости (см. оценочные материалы по дисциплине).

3. По итогам текущего контроля успеваемости по дисциплине, обучающийся должен пройти промежуточную аттестацию (см. оценочные материалы по дисциплине).

8. Описание материально-технического и учебно-методического обеспечения, необходимого для реализации программы магистратуры по дисциплине

8.1. Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой магистратуры, укомплектованные специализированной учебной мебелью и оснащенные оборудованием и техническими средствами обучения, служащими для представления учебной информации большой аудитории: стационарным или переносным экраном, маркерной доской и (или) меловой доской, стационарным или переносным мультимедийным проектором, персональными компьютерами.

Все помещения, используемые для проведения учебных занятий и самостоятельной работы, соответствуют действующим санитарным и противопожарным нормам и правилам.

Для проведения лабораторных работ используются дисплейные классы кафедры «Информационные и вычислительные системы». Классы оборудованы персональными компьютерами, включая сервер локальной сети для размещения методических материалов и результатов выполнения лабораторных работ. На компьютерах установлен комплект необходимого программного обеспечения, приведенного в п. 8.2.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

8.2. Университет обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства:

- операционная система Windows, MS Office, Антивирус Касперский;
- среда программирования Python (бесплатное, свободно распространяемое программное обеспечение; режим доступа <https://python.org/downloads/>)
- NetBeans IDE 8.2 (бесплатное, свободно распространяемое программное обеспечение; режим доступа <https://netbeans.org/downloads/>)

8.3. Обучающимся обеспечен доступ (удаленный доступ) к современным профессиональным базам данных:

- Научная электронная библиотека [eLIBRARY.RU](http://elibrary.ru) [Электронный ресурс] – Режим доступа: <https://www.elibrary.ru> (свободный доступ);
- Электронно-библиотечная система «Лань» [Электронный ресурс] – Режим доступа: <https://e.lanbook.com> (свободный доступ);
- Электронная библиотечная система «Юрайт» [Электронный ресурс]– Режим доступа: <https://urait.ru> (свободный доступ).

8.4. Обучающимся обеспечен доступ (удаленный доступ) к информационным справочным системам:

8.5. Перечень печатных и электронных изданий, используемых в образовательном процессе:

1. Рамальо Л. Python. К вершинам мастерства – ДМК Пресс, 2016.
2. Рейтц К, Шлюссер Т. [Автостопом по Python](#) – Издательский дом «Питер», 2017
3. Дауни А.Б. Основы Python – Издательский дом «Питер», 2020, - 312с.
4. Вандерплас Дж. Python для сложных задач.: Издательский дом «Питер», 2019, 572 с.
5. Мэтиз Э. Изучаем Python.: Издательский дом «Питер», 2019, - 492 с.

8.6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых в образовательном процессе:

1. Личный кабинет обучающегося и электронная информационно-образовательная среда [Электронный ресурс]. – Режим доступа: <http://sdo.pgups.ru/> (для доступа к полнотекстовым документам требуется авторизация);
2. Научно-техническая библиотека университета [Электронный ресурс]. – Режим доступа: <http://library.pgups.ru/> (свободный доступ).

Разработчик рабочей программы,
доцент

В.Е. Петров

« 23 » декабря 2024 г.