

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)**

Институт информационных технологий и радиоэлектроники



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«Программирование на Python»

направление подготовки / специальность
09.04.01 «Информатика и вычислительная техника»

направленность (профиль) подготовки
Инженерия искусственного интеллекта

г. Владимир
2022

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Программирование на Python» является освоение современного языка программирования «Python», использование его для решения конкретных задач и создания приложений. Задачи дисциплины «Программирование на Python» - изучение тем: структуры данных, управление потоком, циклы, итерация, поколение, функция, модули, ООП, обработка ошибок, работа с файлами.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Программирование на Python» относится к обязательной части учебного плана.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции <i>(код, содержание индикатора)</i>	Результаты обучения по дисциплине	
ОПК-4. Способен применять на практике новые научные принципы и методы исследований	ОПК-4.1. Знать: новые научные принципы и методы исследований. ОПК-4.2. Уметь: применять на практике новые научные принципы и методы исследований. ОПК-4.3. Иметь навыки: применения новых научных принципов и методов исследования для решения профессиональных задач.	Знает: новые научные принципы и методы исследований. Умеет: применять на практике новые научные принципы и методы исследований. Имеет навыки: применения новых научных принципов и методов исследования для решения профессиональных задач.	Тестовые вопросы и задания, задания для контрольной работы, задания для самостоятельной работы, вопросы к экзамену
ПК-2. Способен выбирать, разрабатывать и проводить экспериментальную проверку работоспособности и программных компонентов систем искусственного интеллекта по обеспечению требуемых критериев эффективности и качества	ПК-2.1. Знать: ПК-2.1.1. основные критерии эффективности и качества функционирования системы искусственного интеллекта: точность, релевантность, достоверность, целостность, быстрота решения задач, надежность, защищенность функционирования систем искусственного интеллекта ПК-2.1.2. методы, языки и программные средства разработки программных	Знает: основные критерии эффективности и качества функционирования системы искусственного интеллекта: точность, релевантность, достоверность, целостность, быстрота решения задач, надежность, защищенность функционирования систем искусственного интеллекта; методы, языки и программные средства разработки программных	Тестовые вопросы и задания, задания для контрольной работы, задания для самостоятельной работы, вопросы к экзамену

функционирования	<p>компонентов систем искусственного интеллекта</p> <p>ПК-2.2. Уметь: ПК-2.2.1. выбирать, адаптировать, разрабатывать и интегрировать программные компоненты систем искусственного интеллекта с учетом основных критериев эффективности и качества функционирования</p> <p>ПК-2.3. Иметь навыки: ПК-2.3.1. экспериментальной проверки работоспособности программных компонентов систем искусственного интеллекта</p>	<p>компонентов систем искусственного интеллекта</p> <p>Умеет: выбирать, адаптировать, разрабатывать и интегрировать программные компоненты систем искусственного интеллекта с учетом основных критериев эффективности и качества функционирования</p> <p>Имеет навыки: экспериментальной проверки работоспособности программных компонентов систем искусственного интеллекта</p>	
------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов

Тематический план форма обучения – очная

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Контактная работа обучающихся с педагогическим работником				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	в форме практической подготовки		
1	Основы программирования на языке Python	1	1-6	6		6	3	36	Рейтинг-контроль №1
2	Инструменты разработчика на языке Python	1	7-12	6		6	3	36	Рейтинг-контроль №2
3	Прикладные сферы применения языка Python	1	13-18	6		6	3	36	Рейтинг-контроль №3
Всего за 1 семестр:				18		18		108	Экзамен, 36
Наличие в дисциплине КП/КР									
Итого по дисциплине				18		18		108	Экзамен, 36

Содержание лекционных занятий по дисциплине

1. Основы программирования на языке Python
Философия Python. Введение в программирование. Интерпретируемые языки программирования. Интерпретатор. IDE. Переменные, основные типы данных. Основы структур данных. Процедурное программирование. Понятие функции. Встроенная библиотека. Элементы функционального программирования. Основы объектно-ориентированного программирования (ООП)
2. Инструменты разработчика на языке Python
Библиотеки. Репозитории. Инструменты разработчика. Code review. Стили программирования
3. Прикладные сферы применения языка Python
Взаимодействие с другими программами. Построение информационных систем. Источники данных. Библиотеки для анализа данных в Python. Обзор. Интеграция с базами данных. Обзор. Веб-программирование на Python. Обзор.

Содержание лабораторных занятий по дисциплине

1. Элементы функционального программирования на Python
2. Объектно-ориентированное программирование на Python
3. Библиотеки и репозитории на Python
4. Инструменты разработчика Python
5. Проверка кода. Стили программирования
6. Работа с текстовыми файлами на Python
7. Библиотеки для анализа данных на Python
8. Интеграция баз данных в Python
9. Веб-программирование на Python

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

5.1. Текущий контроль успеваемости

Рейтинг-контроль №1

1. Философия Python. Введение в программирование.
2. Интерпретируемые языки программирования.
3. Интерпретатор. IDE
4. Переменные, основные типы данных
5. Основы структур данных
6. Процедурное программирование. Понятие функции.
7. Встроенная библиотека
8. Элементы функционального программирования
9. Основы объектно-ориентированного программирования (ООП)

Рейтинг-контроль №2

1. Библиотеки
2. Репозитории
3. Инструменты разработчика

4. Code review.
5. Стили программирования

Рейтинг-контроль №3

1. Взаимодействие с другими программами. Построение информационных систем
2. Источники данных
3. Библиотеки для анализа данных в Python. Обзор
4. Интеграция с базами данных. Обзор
5. Веб-программирование на Python. Обзор

5.2. Промежуточная аттестация по итогам освоения дисциплины (экзамен)

Примерный перечень вопросов к экзамену:

1. Функциональное программирование на Python.
2. Объектно-ориентированное программирование на Python.
3. Создание модулей и пакетов на Python.
4. Создание библиотек на Python.
5. Командная разработка на Python. Использование git.
6. Отладка программ на Python.
7. Рефакторинг кода Python.
8. Обзор кода на Python.
9. Среда разработки (IDE) на Python.
10. Работа с текстовыми файлами различных форматов на Python.
11. Работа с базами данных SQL на Python.
12. Работа с базой данных sqlite на Python.
13. Работа с базой данных MongoDB на Python.
14. Создание веб-приложений на Python.
15. Использование библиотеки Python Flask.
16. Использование библиотеки Python Django.
17. Многопоточное программирование на Python.
18. Библиотеки для многопоточного программирования на Python.
19. Библиотеки для взаимодействия с операционной системой на Python.
20. Библиотеки для анализа данных на Python.

5.3. Самостоятельная работа обучающегося

Самостоятельная работа обучающихся заключается в самостоятельном изучении отдельных тем, практической реализации заданий самостоятельных работ по этим темам, выполнении контрольной работы (теста). Контроль выполнения самостоятельной работы проводится при текущих контрольных мероприятиях и на промежуточной аттестации по итогам освоения дисциплины. Учебно-методическое обеспечение самостоятельной работы – основная литература [1-4], дополнительная литература [1].

Примерное содержание контрольной работы (тест):

1. Выберите библиотеки, которые позволяют вам работать с операционной системой:
 - numpy
 - flask
 - sys
 - os
 - keras
2. Выберите библиотеки для создания веб-приложений:
 - selenium
 - flask
 - scrapy

- django
 - requests
3. Выберите библиотеки для построения интеллектуальных систем:
 - openCV
 - pytorch
 - tornado
 - tensorflow
 4. Для каких целей файл с программным кодом разделен на логические модули:
 - чтобы упростить масштабирование и обслуживание программы
 - чтобы усложнить навигацию по программе
 - чтобы иметь больше файлов
 5. Чтобы python мог отличать обычный пакет от каталога, вам необходимо:
 - нужно назвать каталог особым образом
 - вам ничего не нужно, Python найдет все сам
 - вам нужно добавить `__init__.py` файл в каталог
 6. Какой файл содержит информацию, необходимую для установки пакета:
 - README.md
 - setup.py
 - main.py
 - MANIFEST.in
 7. Какая команда git отправляет изменения из локального репозитория в удаленный?
 - add
 - push
 - commit t
 8. Что такое разрешение конфликтов в git:
 - исправление ошибок при объединении ветвей (например, после выполнения pull)
 - исправлены ошибки при commit
 - исправлены ошибки при создании ветвей
 9. Что НЕ МОЖЕТ сделать отладчик?
 - Пошаговое выполнение кода
 - Просмотр стека вызовов функций
 - Автоматически проверять результат функции на разных значениях
 - Выполнение кода путем ввода или пропуска функций
 10. Какие подходы используются для тестирования программы?
 - Ручное тестирование
 - Тестирование с помощью функций
 - Автоматическое тестирование
 11. Какой компонент Python подходит как для написания тестов, так и для создания документации?
 - Docstring
 - Pytest
 - Pdb
 12. Профилирование кода заключается в поиске неэффективных фрагментов кода с использованием...?
 - синтаксический анализ кода
 - определение времени выполнения различных разделов кода
 - проверка результатов работы различных разделов кода
 13. Что может сделать среда IDE при рефакторинге кода?
 - Извлечение методов, переменных и констант из существующего кода
 - Извлечение классов из существующих сигнатур классов

- Извлечение строк комментариев из программного кода
14. Стандартом, описывающим правила форматирования кода Python, является:
- ISO 9001-2015
 - ГОСТ 19.506-79
 - MISRA
 - PEP8
15. Когда требуется рефакторинг:
- вы работаете над кодом в команде
 - вы изменяете и поддерживаете существующий код
 - вы работаете над кодом в одиночку
 - предоставьте первую рабочую версию кода и больше не редактируйте ее
 - срок службы проектов большой (годы, десятки лет)
 - код написан с нуля, а не отредактирован версией другого человека
 - версии программ меняются очень часто (недели, месяцы)

Примерное содержание СРС:

Самостоятельная работа №1 «Работа с базами данных на Python»

1. Проанализируйте готовые наборы данных Атласа в MongoDB(<https://developer.mongodb.com/article/atlas-sample-datasets>). Вы можете либо развернуть MongoDB локально, загрузив набор данных, либо использовать облако. Для решения проблем вам необходимо установить модуль pymongo.

Задачи анализа:

- Найдите количество записей в `sample_weatherdata`, где давление меньше 1000. Введите одно число в свой ответ.
- В разделе `sample_restaurants` найдите, сколько ресторанов в Бронксе имеют в своем названии слово “Еда” (например, F&B Foods Llc).
- Найдите минимальный и максимальный возраст клиентов в `sample_supplies`. В ответе напишите числа, разделенные запятыми, без пробелов. Пример: 28,64.

2. На порту 27017 хоста 127.0.0.1 запущен сервер MongoDB с базой данных `db`. Эта база данных содержит коллекцию пользователей, в которой пользователи хранятся в следующем виде: `{"name": "Kenneth", "surname": "Cruz", "age": 30, "rating": 5.0, "trip_count": 2478.0, "registered_at": "2010-04-25", "avg_trip_km": 6.99, "favorite_music": "Cream",}`. Реализуйте функцию `get_ages_sum`, которая возвращает сумму возрастов всех пользователей в данной коллекции. Для связи с сервером MongoDB вам необходимо использовать модуль pymongo.

Самостоятельная работа №2 «Создание веб-приложений на Python»

Используя фреймворк flask, разработайте веб-сайт с двумя страницами:

- страница, которая вернет текст «привет»
- страница со счетчиком, который возвращал бы число в виде количества посетителей, зашедших на эту страницу. Те. у первого посетителя есть номер 1, у второго - номер 2 и так далее.

Фонд оценочных материалов (ФОМ) для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ
		Наличие в электронном каталоге ЭБС
Основная литература		
1. Златопольский, Д. М. Основы программирования на языке Python : учебник / Златопольский Д. М. - Москва : ДМК Пресс, 2017. - 284 с. - ISBN 978-5-97060-552-3. - Текст : электронный. Режим доступа : по подписке.	2017	https://www.studentlibrary.ru/book/ISBN9785970605523.html
2. Хахаев, И. А. Практикум по алгоритмизации и программированию на Python / Хахаев И. А. - Москва : Национальный Открытый Университет "ИНТУИТ", 2016. - Текст : электронный. Режим доступа : по подписке.	2016	https://www.studentlibrary.ru/book/intuit_255.html
3. Ленц, М. Python. Непрерывная интеграция и доставка / Ленц М. , пер. с англ. А. Е. Мамонова, Д. А. Беликова. - Москва : ДМК Пресс, 2020. - 168 с. - ISBN 978-5-97060-797-8. - Текст : электронный. Режим доступа : по подписке.	2020	https://www.studentlibrary.ru/book/ISBN9785970607978.html
4. Маккинли, У. Python и анализ данных / Уэс Маккинли - Москва : ДМК Пресс, 2015. - 482 с. - ISBN 978-5-97060-315-4. - Текст : электронный. Режим доступа : по подписке	2015	https://www.studentlibrary.ru/book/ISBN9785970603154.html
Дополнительная литература		
1. Сузи, Р. А. Язык программирования Python / Сузи Р. А. - Москва : Национальный Открытый Университет "ИНТУИТ", 2016. (Основы информационных технологий) - ISBN 5-9556-0058-2. - Текст : электронный. Режим доступа : по подписке.	2016	https://www.studentlibrary.ru/book/ISBN5955600582.html

6.2. Периодические издания

1. Вестник компьютерных и информационных технологий ISSN 1810-7206.
2. Цифровая библиотека научно-технических изданий Института инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronic Engineers (IEEE)) на английском языке – <http://www.ieee.org/ieeexplore>

6.3. Интернет-ресурсы

1. Academic Search Ultimate EBSCO publishing – <http://search.ebscohost.com>
2. eBook Collections Springer Nature – <https://link.springer.com/>
3. Гугл Академия – <https://scholar.google.ru/>
4. Электронно-библиотечная система «Лань» – <https://e.lanbook.com/>
5. Университетская библиотека ONLINE – <https://biblioclub.ru/>
6. Электронно-библиотечная система "Библиокомплектатор" (IPRbooks) <http://www.bibliocomplectator.ru/available>
7. Электронные информационные ресурсы Российской государственной библиотеки <https://www.rsl.ru/>
8. Научная электронная библиотека «КиберЛенинка» <https://cyberleninka.ru/>
9. Портал российского образования www.edu.ru
10. Портал российских электронных библиотек www.elbib.ru

11. Научная электронная библиотека www.eLibrary.ru
12. Научная библиотека ВлГУ library.vlsu.ru
13. Учебный сайт кафедры ИСПИ ВлГУ <https://ispi.cdo.vlsu.ru>
14. Электронная библиотечная система ВлГУ <https://vlsu.bibliotech.ru/>
15. Web of Science Core Collection – <http://apps.webofknowledge.com/>
16. Онлайн-курс “Прикладное программирование на языке Python” – <https://openedu.ru/course/urfu/РУАР/>
17. Real Python Tutorials – <https://realpython.com/>

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий: занятий лекционного типа, занятий лабораторного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы. Лабораторные работы проводятся в компьютерном классе, оборудованном мультимедийным проектором с экраном и обеспеченным доступом в Интернет.

Перечень используемого лицензионного программного обеспечения:

- Операционная система Microsoft Windows 10
- Офисный пакет Microsoft Office 2016
- Бесплатно-распространяемое программное обеспечение (Python – <https://www.python.org/>, Веб - среда разработки для языка программирования Python: google colab - <https://colab.research.google.com/>, Anaconda solution - <https://www.anaconda.com/>).

Рабочую программу составил Куликов К.В. зав. каф. ВТиСУ
(ФИО, должность, подпись)

Рецензент
(представитель работодателя) _____ Генеральный директор ООО "Диаграмма" Протягов И.В.

Программа рассмотрена и одобрена на заседании кафедры ВТ и СУ
Протокол № 1 от 29 августа 2022 года
Заведующий кафедрой Куликов К.В. _____

Рабочая программа рассмотрена и одобрена
на заседании учебно-методической комиссии направления 09.04.01 информатика и
вычислительная техника
Протокол № 1 от 29 августа 2022 года
Председатель комиссии Куликов К.В. зав. каф. ВТиСУ _____

**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ**

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

в рабочую программу дисциплины

Программирование на Python

образовательной программы направления подготовки 09.04.01 «Информатика и вычислительная техника», направленность: *Инженерия искусственного интеллекта (магистратура)*

Номер изменения	Внесены изменения в части/разделы рабочей программы	Исполнитель ФИО	Основание (номер и дата протокола заседания кафедры)

Заведующий кафедрой _____ / _____