

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО
Заместитель директора

А.И. Рыбакова

	Рабочая программа дисциплины (модуля)
по дисциплине:	Современные средства разработки
по направлению:	Ядерные физика и технологии
профиль подготовки:	Ядерная физика, УТС и компьютерные методы в физике Физтех-школа физики и исследований им. Ландау центр дополнительного, дополнительного профессионального и онлайн-образования "Пуск"
курс:	1
квалификация:	магистр

Семестры, формы промежуточной аттестации:

- 1 (осенний) - Зачет
- 2 (весенний) - Зачет

Аудиторных часов: 90 всего, в том числе:

- лекции: 30 час.
- семинары: 60 час.
- лабораторные занятия: 0 час.

Самостоятельная работа: 180 час.

Всего часов: 270, всего зач. ед.: 6

Количество контрольных работ, заданий: 2

Программу составили:

Р.С. Кулиев, старший преподаватель
А.А. Чистов, ассистент
Е.А. Машевская, старший методист
Ж.И. Зубцова, канд. физ.-мат. наук, эксперт

Программа обсуждена на заседании центра дополнительного, дополнительного профессионального и онлайн-образования "Пуск" 28.01.2025

Аннотация

Дисциплина состоит из двух модулей:

1 модуль. Основы программирования на Python.

2 модуль. Разработка на платформе 1С:Предприятие.

По итогам обучения обучающийся будет способен формализовать и алгоритмизировать поставленную задачу, написать программный код с использованием языков программирования, оформить код в соответствии с установленными требованиями.

1. Цели и задачи

Цель дисциплины

- формирование/совершенствование компетенций студентов в области решения профессиональных задач по программированию с использованием языков Python и 1С, применения основ программирования, в том числе асинхронного, на Python, проектирования программного обеспечения с помощью встроенного языка 1С.

Задачи дисциплины

- научиться использовать базовые типы и конструкции языка программирования Python;
- сформировать умение работать со стандартными структурами данных в Python, писать функции на Python, применять функциональные особенности языка, работать с файлами с помощью языка Python;
- научиться применять механизмы наследования, создавать классы и работать с ними, обрабатывать исключения;
- сформировать умение искать и исправлять ошибки в программе на Python, тестировать программы на Python;
- научиться писать многопоточный код на Python, писать асинхронный код на Python, работать с сетью, создать своё серверное сетевое приложение;
- изучить основные принципы, подходы и механизмы разработки бизнес-приложений на платформе 1С:Предприятие;
- изучить возможности быстрой кросс-платформенной разработки на платформе 1С:Предприятие;
- изучить возможности создания веб и мобильных приложений на платформе 1С:Предприятие.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-3 Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной задачи	УК-3.1 Организует и координирует работу участников проекта, способствует конструктивному преодолению возникающих разногласий и конфликтов
	УК-3.4 Способен планировать командную работу, распределять поручения членам команды, организовать обсуждение разных идей и мнений
ОПК-2 Способен применять современные методы исследования, оценивать и представлять результаты выполненной работы	ОПК-2.1 Способен применять знания и навыки по использованию информационно-коммуникационных технологий для поиска и изучения научной литературы, применения прикладных программных продуктов

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

- принципы организации кода;
- понятия коллекций и функций в Python;
- назначение функций в языках программирования;
- сетевые термины (сокеты, клиент, сервер);
- синтаксис встроенного языка платформы 1С:Предприятие;
- возможности быстрой кросс-платформенной разработки на платформе 1С:Предприятие;
- базовые типы и конструкции Python;
- роль коллекций и функций в программировании на Python;
- принцип работы клиент-серверной архитектуры;
- возможности создания веб и мобильных приложений на платформе 1С:Предприятие;
- понятия классов и объектов в Python, понимает их взаимосвязь;
- понятие наследования;
- механизмы наследования и его роль в программировании на Python;
- библиотеки Python для обработки данных;
- где находится каталог библиотек;
- понятия тестирования и отладки;
- особенности работы с глобальным шлюзом GIL;
- несколько встроенных функций языка Python;
- понятие особых методов классов;
- роль особых классов в программировании на Python;
- особенности объектно-ориентированной модели в Python;
- механизм формирования исключений;
- принцип работы системы Git;
- роль тестирования в обеспечении качества ПО;
- виды тестирования;
- основные принципы, подходы и механизмы разработки бизнес-приложений на платформе 1С:Предприятие.

уметь:

- использовать базовые типы и конструкции Python для написания простых программ;
- сопоставить и выбрать необходимую структуру данных для конкретной практической задачи;
- составить иерархию классов и описать их методы для конкретной практической задачи;
- выделять в задаче на естественном языке необходимость применения многопоточности;
- выделять в задаче на естественном языке необходимость применения базовых конструкций языка;
- определить и спроектировать необходимые классы и методы классов для конкретной предметной области;
- написать простой код на Python;
- написать собственную функцию на языке Python;
- использовать магические методы в написании собственных программ на Python;
- писать многопоточный код на Python;
- создать программу по описанию задачи на естественном языке с использованием коллекций языка Python;
- использовать конструкции для генерации исключений на Python;
- установить интерпретатор Python себе на компьютер;
- установить среду разработки PyCharm;
- читать и записывать данные из файла;
- устанавливать внешние библиотеки в Python;
- использовать Git для отслеживания истории изменений версий кода;
- протестировать код на Python;
- вручную создавать или устанавливать библиотеки;
- проанализировать код на Python и исправить ошибки в программе;
- использовать коллекции и функции для написания программ на Python;
- создавать классы и использовать методы классов;
- создавать программу для обработки исключений на Python в условиях конкретной практической задачи;
- создавать кросс-платформенные приложения на платформе 1С:Предприятие;
- вызывать и получать результат выполнения встроенных функций Python;
- создать программу по описанию задачи на естественном языке с использованием функций языка Python;
- написать программу на Python с применением библиотек для конкретной практической задачи;
- использовать основные функции и методы Python для написания собственных клиентских и серверных приложений;
- определить, требуется ли в приложении многопоточность;
- анализировать код статически;
- создавать автоматические тесты;
- измерять покрытие кода тестами;
- разрабатывать веб и мобильные приложения на платформе 1С:Предприятие;
- создавать формы клиентского интерфейса.

владеть:

- навыками разработки алгоритмов и программ, пригодных для практического применения;
- навыками освоения методик использования программных средств для решения практических задач;
- навыками написания программного кода с использованием языков программирования, оформления кода в соответствии с установленными требованиями.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа

1	Основы программирования на Python	15	30		90
2	Разработка на платформе 1С:Предприятие	15	30		90
Итого часов		30	60		180
Подготовка к экзамену		0 час.			
Общая трудоёмкость		270 час., 6 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 1 (Осенний)

1. Основы программирования на Python

1.1. Введение в программирование на Python

1.1.1. Введение в программирование на Python

Лекция

Введение в программирование на Python

О языке Python

Установка интерпретатора Python на Windows

Установка интерпретатора Python на Linux

Работа в IDE PyCharm. Первая программа

Введение в Python

Ввод и вывод данных

Примеры простейших программ

Итоги занятия

Практическая работа

Выполнение заданий по теме лекции.

Самостоятельная работа

Дополнительные материалы

Тест на самопроверку по уроку

Задания на программирование

1.1.2. Типы данных. Конструкции языка

Лекция

Числовые типы и операции над ними

Строковый тип данных

Логический тип данных

Условный оператор

Цикл while

Цикл for

Операторы continue и break

Практическая работа

Выполнение заданий по теме лекции

Самостоятельная работа

Дополнительные материалы

Тест на самопроверку по уроку

Задания на программирование

Тест на оценку

1.2. Структуры данных и функции

1.2.1. Коллекции

Лекция

Встроенные структуры данных

Множества. Операции с множествами

Строки. Индексация строк
Списки
Методы split() и join(). Списочные выражения
Кортежи
Словари
Примеры работы с множествами
Примеры работы со строками
Примеры работы со списками
Примеры работы с методами split() и join()
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
1.2.2. Функции. Работа с файлами
Лекция
Именные функции, инструкция def
Возврат значений из функции
Множественное присваивание, упаковка и распаковка аргументов
Аргументы по умолчанию и именованные аргументы
Инструкция pass(). Согласованность аргументов
Функция как объект. Функции высших порядков
Лямбда-функция
Принципы работы с файлами на Python
Разбор задач на работу с файлами
Правила записи кода PEP 8
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
Тест на оценку
1.3. Объектно-ориентированное программирование
1.3.1. Классы и объекты
Лекция
Введение в объектно-ориентированное программирование
Классы и экземпляры классов
Методы
Пример рефакторинга программы на ООП
Инкапсуляция
Полиморфизм
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
1.3.2. Наследование
Лекция
Наследование классов
Особенности объектной модели в Python
Элементы статической типизации. Абстрактные классы и протоколы

Множественное наследование
Проблемы, связанные с наследованием
Композиция классов
Практические рекомендации
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
Тест на оценку

1.4. Углубленный Python

1.4.1. Специальные методы классов. Механизм работы классов

Лекция
Специальные методы классов
Хеширование
Специальные атрибуты
Перегрузка операторов
Коллекции и итераторы
Контекстные менеджеры
Callable-объекты и декораторы
Построитель классов данных dataclass
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование

1.4.2. Работа с ошибками

Лекция
Обработка ошибок в программировании
Две основные стратегии обработки ошибок
Синтаксис обработки ошибок
Обработка исключений и производительность
Генерация исключений
Инструкция assert
Классы исключений
Создание пользовательских исключений
Практика работы с исключениями
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование

1.4.3. Установка внешних библиотек. Работа с Git

Лекция
Инструкция import
Модули стандартной библиотеки
Создание своего модуля на Python
Создание виртуального окружения
Установка внешних библиотек Python
Параллельная установка версий интерпретатора
Git. Работа с распределёнными системами управления версиями

Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Тест на самопроверку по уроку
Задания на программирование
Тест на оценку
1.5. Работа с сетью
1.5.1. Работа с сетью. Сокеты
Лекция
Сетевые протоколы
Сокеты, программа клиент-сервер
Таймауты и обработка сетевых ошибок
Одновременная обработка нескольких соединений
Простой HTTP-сервер
Основные библиотеки для работы с HTTP
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
1.6. Тестирование кода
1.6.1. Контроль качества программного кода
Лекция
Обеспечение качества и тестирование ПО
Инструменты статического анализа кода
Инструменты тестирования
Использование фикстур и мок-объектов в pytest
Покрывание кода тестами. Плагин pytest-cov
Практическая работа
Выполнение заданий по теме лекции
Самостоятельная работа
Дополнительные материалы
Тест на самопроверку по уроку
Задания на программирование
1.7. Бонусные материалы для самостоятельного изучения
1.7.1. Процессы и потоки
Подходы к параллельному исполнению кода
Создание потоков
Создание процессов
Синхронизация потоков
Очереди и поддержание пула потоков
Блокировка GIL в Python
Использование универсального интерфейса concurrent.futures
Итоги занятия
Тест на самопроверку по уроку
Тест на оценку
1.7.2. Асинхронное программирование
Асинхронное программирование как альтернатива потокам
Введение в генераторы
Генераторы как сопрограммы
Синтаксис async, await и awaitable-объекты
Модуль asyncio. основные функции
Модуль asyncio. работа с сетью

Модуль asynсio. работа с потоками и процессами
Асинхронная загрузка данных с помощью библиотеки httpх
Итоги занятия
Тест на самопроверку по уроку
Задания на программирование

Семестр: 2 (Весенний)

2. Разработка на платформе 1С:Предприятие

1. Введение.

Знакомство. Установка платформы. Обзор конфигуратора. Создание простого приложения

2. Константы. Формы. Подсистемы.

Создание констант. Создание форм. Вывод элементов на форму. Создание подсистемами

3. Модули. Клиент-серверная модель. Программирование.

Основы программирования на 1С. Разбор клиент-серверной модели.

4. Справочники. Формы объектов. Объектное чтение. Перечисления.

Лекция

Создание и назначение справочников. Создание и назначение форм справочников. Чтение из базы данных. Создание и назначение перечислений

Практическая работа

Работа с формами. Создание форм без использования модальности

Самостоятельная работа

Добавить удобный интерфейс пользователя системы

5. Документы. Регистры накопления. Регистры сведений.

Создание и назначение справочников. Создание и назначение регистров накопления. Создание и назначение регистров сведений.

6. Запросы.

Лекция

Введение в запросы. Синтаксис. Конструктор запросов

Практическая работа

Работа с запросами и извлечение данных из базы, работа со сложными запросами, соединения и объединения запросов, формирование и настройка отчетов, формирование отчетов с диаграммами

Самостоятельная работа

Добавить отчеты о продажах в разрезах товара, группы товаров, контактов, временных интервалов

Добавить построение диаграмм продаж, в том числе, в разрезе по отдельным менеджерам.

7. Ввод на основании. Печатные формы. Передача данных между формами.

Изучение и использование механизма ввода на основании. Создание печатных форм и их заполнение. Способы передачи данных между формами и их использование

8. Отчеты. Схема компоновки данных.

Создание отчетов и работа со схемой компоновки данных.

9. Динамические списки. Функциональные опции. Условное оформление.

Произвольные запросы в динамических списках. Условное оформление. Программное взаимодействие с динамическим списком.

10. Роли. Права доступа. Пользователи.

Создание ролей. Ограничение прав доступа. Создание пользователей.

11. Загрузка из Excel. Загрузка и хранение изображений.

Программная реализация загрузки из Excel. Программная загрузка изображений для хранения в базе.

12. Расширения. Выдача финального задания. Ответы на вопросы.

Создание расширений. Подготовка к проведению финальной работы. Ответы на вопросы по курсу и разбор финальной работы.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Система дистанционного обучения:

Обучающемуся необходимо наличие доступа в сеть интернет, компьютер.

Преподавателю курса необходимо наличие доступа администратора курса и оборудование для проведения дистанционных семинаров (вебинаров), качественный отказоустойчивый доступ в сеть интернет.

6. Перечень рекомендуемой литературы

Основная литература

Литература из средств кафедры:

1. Think Python [Электронный ресурс] – Режим доступа - <https://greenteapress.com/wp/think-python-2e/>
2. Automate the Boring Stuff with Python [Электронный ресурс] – Режим доступа - <https://automatetheboringstuff.com/>
3. Dive Into Python 3 [Электронный ресурс] – Режим доступа - <http://diveintopython3.problemsolving.io/>
4. Problem Solving with Algorithms and Data Structures using Python [Электронный ресурс] – Режим доступа - <https://runestone.academy/runestone/static/pythonds/index.html>
5. Swaroop Chitlur. A Byte of Python [Электронный ресурс] – Режим доступа - <https://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.02.pdf> – 2020.
6. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для вузов — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2021. — 210 с
7. Свейгарт Э. Автоматизация рутинных задач с помощью Python [Текст]: практическое руководство для начинающих – Диалектика, 2021. – 672 с.
8. Книга издательства O'Reilly (доступна онлайн, англ.): <https://docs.python-guide.org/>
9. Лутц, Марк. Изучаем Python: авторитетный курс объектно-ориентированного программирования / Марк Лутц; перевод с английского Ю. Н. Артеменко. – 5-е изд. – Москва: Диалектика; Санкт-Петербург: Диалектика, 2020.
10. Учебная статья: <https://docs.python.org/3/tutorial/errors.html>
11. 1С:Предприятие 8.3. Версия для обучения программированию, Фирма "1С".
12. Веб-сервис для учебного тестирования по платформе "1С:Предприятие 8" – <http://edu.1c.ru/dist-training>.
13. Интернет ресурс: <http://v8.1c.ru>
14. Комплект вопросов сертификационного экзамена на знание основных механизмов платформы "1С:Предприятие 8" с примерами решений, М.: Изд-во ООО "1С", 2021.
15. Методические материалы курса обучения "Знакомство с платформой "1С:Предприятие 8.3", М.: Изд-во ООО "1С", 2020.
16. Методические материалы курса обучения "Основные механизмы платформы "1С:Предприятие 8.3", М.: Изд-во ООО "1С", 2020.
17. Методические материалы курса обучения "Основы программирования в системе "1С:Предприятие 8", М.: Изд-во ООО "1С", 2020.
18. Рабочая тетрадь "Основные механизмы платформы "1С:Предприятие 8.3", М.: Изд-во ООО "1С", 2014.

Дополнительная литература

Дополнительная литература из средств кафедры:

1. Бэрри П. Изучаем программирование на Python [Текст] / П. Бэрри. – 2017.
2. 101 совет начинающим разработчикам в системе "1С:Предприятие 8", Хрусталева Е.Ю., М.: Изд-во ООО "1С-Паблишинг", 2015.
3. 1С:Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы, Радченко М.Г., Хрусталева Е.Ю., М.: Изд-во ООО "1С-Паблишинг", 2013.
4. Профессиональная разработка в системе "1С:Предприятие 8". Издание 2, Ажеронок В.А., Габец А.П., Гончаров Д.И., Козырев Д.В., Кухлевский Д.С., Островерх А.В., Радченко М.Г., Хрусталева Е.Ю., под редакцией Радченко М.Г., М.: Изд-во ООО "1С-Паблишинг", 2013.
5. Разработка интерфейса прикладных решений на платформе "1С:Предприятие 8", Ажеронок В.А., Островерх А.В., Радченко М.Г., Хрусталева Е.Ю., М.: Изд-во ООО "1С-Паблишинг", 2018.
6. Сборник задач по разработке на платформе 1С:Предприятие (1С:Enterprise), Чистов П.А., М.: Изд-во ООО "1С-Паблишинг", 2020.
7. Язык запросов "1С:Предприятие 8", Хрусталева Е.Ю., под ред. Радченко М.Г., М.: Изд-во ООО "1С-Паблишинг", 2013.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

1. Официальный сайт Python - <https://www.python.org/>
2. Введение в анализ данных с помощью Pandas - <https://habr.com/ru/post/196980/>
3. Среда разработки IDE PyCharm - <https://www.jetbrains.com/pycharm/>
4. PEP 8 – Style Guide for Python Code - <https://peps.python.org/pep-0008/>
5. Официальная документация (англ.): <https://docs.python.org/3/>
6. Документация по исключениям: <https://docs.python.org/3/library/exceptions.html>
7. Модель GitFlow: <https://nvie.com/posts/a-successful-git-branching-model/>
8. Современный подход к версионированию для CI/CD:
<https://www.atlassian.com/continuous-delivery/continuous-integration/trunk-based-development>

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Документация Postgres про сравнение строк -
<https://postgrespro.ru/docs/postgrespro/9.5/functions-matching>
Документация Postgres про другие функции работы со строками -
<https://postgrespro.ru/docs/postgrespro/9.5/functions-string>
Тестер регулярных выражений - <https://www.regextester.com>
Интерактивный учебник по SQL - <http://www.sql-tutorial.ru/ru/content.html>
Введение в анализ данных с помощью Pandas - <https://habr.com/ru/post/196980/>
Начало работы с Power BI -
<https://docs.microsoft.com/ru-ru/power-bi/fundamentals/desktop-getting-started>
Интернет ресурс: <http://v8.1c.ru>

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Самостоятельная работа подразделяется на аудиторную и внеаудиторную. Аудиторную самостоятельную работу составляют практические задания, которые выполняются слушателями во время учебных занятий, результаты ее выполнения проверяются и оцениваются преподавателем в учебном процессе.

Внеаудиторная самостоятельная работа включает формы: изучение дополнительной литературы, подготовка итоговых проектов по модулям, подготовка проекта.

Основными критериями качества организации самостоятельной работы служит наличие контроля результатов самостоятельной работы.

Основными современными формами организации самостоятельной работы являются творческие работы и работа с информационными компьютерными технологиями.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Ядерная физика и технологии
профиль подготовки:	Ядерная физика, УТС и компьютерные методы в физике Физтех-школа физики и исследований им. Ландау центр дополнительного, дополнительного профессионального и онлайн-образования "Пуск"
курс:	<u>1</u>
квалификация:	магистр

Семестры, формы промежуточной аттестации:

- 1 (осенний) - Зачет
- 2 (весенний) - Зачет

Разработчики:

Р.С. Кулиев, старший преподаватель
А.А. Чистов, ассистент
Е.А. Машевская, старший методист
Ж.И. Зубцова, канд. физ.-мат. наук, эксперт

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-3 Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной задачи	УК-3.1 Организует и координирует работу участников проекта, способствует конструктивному преодолению возникающих разногласий и конфликтов
	УК-3.4 Способен планировать командную работу, распределять поручения членам команды, организовать обсуждение разных идей и мнений
ОПК-2 Способен применять современные методы исследования, оценивать и представлять результаты выполненной работы	ОПК-2.1 Способен применять знания и навыки по использованию информационно-коммуникационных технологий для поиска и изучения научной литературы, применения прикладных программных продуктов

2. Показатели оценивания компетенций

В результате изучения дисциплины «Современные средства разработки» обучающийся должен:

знать:

- принципы организации кода;
- понятия коллекций и функций в Python;
- назначение функций в языках программирования;
- сетевые термины (сокеты, клиент, сервер);
- синтаксис встроенного языка платформы 1С:Предприятие;
- возможности быстрой кросс-платформенной разработки на платформе 1С:Предприятие;
- базовые типы и конструкции Python;
- роль коллекций и функций в программировании на Python;
- принцип работы клиент-серверной архитектуры;
- возможности создания веб и мобильных приложений на платформе 1С:Предприятие;
- понятия классов и объектов в Python, понимает их взаимосвязь;
- понятие наследования;
- механизмы наследования и его роль в программировании на Python;
- библиотеки Python для обработки данных;
- где находится каталог библиотек;
- понятия тестирования и отладки;
- особенности работы с глобальным шлюзом GIL;
- несколько встроенных функций языка Python;
- понятие особых методов классов;
- роль особых классов в программировании на Python;
- особенности объектно-ориентированной модели в Python;
- механизм формирования исключений;
- принцип работы системы Git;
- роль тестирования в обеспечении качества ПО;
- виды тестирования;
- основные принципы, подходы и механизмы разработки бизнес-приложений на платформе 1С:Предприятие.

уметь:

- использовать базовые типы и конструкции Python для написания простых программ;
- сопоставить и выбрать необходимую структуру данных для конкретной практической задачи;
- составить иерархию классов и описать их методы для конкретной практической задачи;
- выделять в задаче на естественном языке необходимость применения многопоточности;
- выделять в задаче на естественном языке необходимость применения базовых конструкций языка;
- определить и спроектировать необходимые классы и методы классов для конкретной предметной области;
- написать простой код на Python;
- написать собственную функцию на языке Python;
- использовать магические методы в написании собственных программ на Python;
- писать многопоточный код на Python;
- создать программу по описанию задачи на естественном языке с использованием коллекций языка Python;
- использовать конструкции для генерации исключений на Python;
- установить интерпретатор Python себе на компьютер;
- установить среду разработки PyCharm;
- читать и записывать данные из файла;
- устанавливать внешние библиотеки в Python;
- использовать Git для отслеживания истории изменений версий кода;
- протестировать код на Python;
- вручную создавать или устанавливать библиотеки;
- проанализировать код на Python и исправить ошибки в программе;
- использовать коллекции и функции для написания программ на Python;
- создавать классы и использовать методы классов;
- создавать программу для обработки исключений на Python в условиях конкретной практической задачи;
- создавать кросс-платформенные приложения на платформе 1С:Предприятие;
- вызывать и получать результат выполнения встроенных функций Python;
- создать программу по описанию задачи на естественном языке с использованием функций языка Python;
- написать программу на Python с применением библиотек для конкретной практической задачи;
- использовать основные функции и методы Python для написания собственных клиентских и серверных приложений;
- определить, требуется ли в приложении многопоточность;
- анализировать код статически;
- создавать автоматические тесты;
- измерять покрытие кода тестами;
- разрабатывать веб и мобильные приложения на платформе 1С:Предприятие;
- создавать формы клиентского интерфейса.

владеть:

- навыками разработки алгоритмов и программ, пригодных для практического применения;
- навыками освоения методик использования программных средств для решения практических задач;
- навыками написания программного кода с использованием языков программирования, оформления кода в соответствии с установленными требованиями.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Тестовые задания

1. Какая версия языка Python является актуальной в настоящее время?
 - Python 3.X
 - Python 2.6.X
 - Python 1.X
2. Что такое инструкция в языках программирования?
 - команда для выполнения какого-либо действия
 - документ с перечислением всех команд языка
 - данные, которые пользователь вводит с консоли в программу

- спецсимволы, используемые для написания кода

3. Какие элементы используют для формирования структуры программы в Python? Выберите все верные варианты.

- пробелы
- двоеточие
- табуляция
- фигурные скобки
- команды языка
- знак #

4. Людмила пишет программу на языке Python. Какую команду ей нужно записать на место пропуска, что вывести на экран сумму чисел a и b? Запишите ответ строчными буквами.

Код программы Людмилы.

a = 2

b = 4

Практические задания

Тема 1. Установка системы, варианты поставки программы. Пример быстрой разработки приложения «с нуля до отчетов».

Тема 2. Работа с регистром сведений. Работа с конструктором управляемых форм. Конструирование интерфейса на подсистемах.

Тема 3. Работа с документами.

Тема 4. Работа с формами. Создание форм без использования модальности.

Тема 5. Работа с запросами и извлечение данных из базы, работа со сложными запросами, соединения и объединения запросов, формирование и настройка отчетов, формирование отчетов с диаграммами.

Тема 6. Формирование планов обмена, организация выгрузки данных в табличные файлы, настройка интеграции с веб-сервисами, работа с внешними компонентами.

Тема 7. Работа с элементами системы взаимодействия, создание чат-ботов, контекстных обсуждений.

Тема 8. Пример создания простейшего мобильного приложения.

Тема 9. Настройка веб-сервера. Публикация конфигурации на веб-сервере. Применение технологии PWA.

Тема 10. Выполнение регламентных операций с информационной базой. Настройка пользователей системы и прав доступа.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

Задание на программирование

1. Приветствие

На вход программа получает строку — имя пользователя. Считайте её и выведите на экран фразу: «Привет, [имя]!».

Обратите внимание, что выводимая строка должна быть написана в точности так, как в задании. Если будет отличаться регистр букв или будут отсутствовать знаки препинания, программа засчитана не будет.

Ввод

Вывод

Валентин Привет, Валентин!

2. Вычисление значения предыдущего числа и следующего

На вход программа получает целое число a. Получите значения предыдущего числа и следующего. Выведите результат a-1, a, a+1 по одному числу в строку.

Ввод	Результат
5	4
	5
	6

3. Питомец

Егор разрабатывает приложение «Кошечки-собачки». Ему нужно считать данные о питомце, которые введены в консоль, и подтвердить их ввод путем вывода на экран.

На вход программа получает четыре параметра: имя питомца (name), вид животного (animal), порода (breed) и возраст (age).

Необходимо вывести введенную информацию в виде:

Ваш питомец: animal name

Порода питомца: breed

Возраст питомца: age

Ввод	Результат
------	-----------

Кузя	Ваш питомец: кот Кузя
------	-----------------------

кот	
-----	--

мейн-кун	Порода питомца: мейн-кун
----------	--------------------------

5	Возраст питомца: 5
---	--------------------

Итоговый проект

Разработать бизнес-приложение в соответствии с следующим сценарием:

1. Установить систему, создать конфигурацию, сделать простейшую записную книжку.
2. Расширить записную книжку, добавив возможность создания списка комментариев к контакту, а также загрузки фотографии контакта.
3. Создать простейшую систему автоматизации интернет-магазина.
4. Добавить возможность создания продаж контактам из записной книжки.
5. Добавить удобный интерфейс пользователя системы.
6. Добавить отчеты о продажах в разрезах товара, группы товаров, контактов, временных интервалов.
7. Добавить построение диаграмм продаж, в том числе, в разрезе по отдельным менеджерам.
8. Организовать выгрузку продаж в табличном формате с возможностью выбора столбцов в выгрузке.
9. Добавить возможность выставления цены в валюте (путем интеграции с ЦБ РФ для получения курсов валют).
10. Создать чат-бота для клиентов (для обсуждения заказов).
11. Добавить возможность вести контекстные обсуждения по заказам для пользователей системы.
12. Создать мобильное приложение для разработанной конфигурации.
13. Опубликовать свое приложение как веб-сервис. Добавить возможность установки как PWA.
14. Добавить пользователей системы. Реализовать возможность восстановления пароля пользователем.

Критерии оценивания

Максимальная сумма, которую можно набрать, успешно выполнив все контрольные мероприятия, составляет 100 баллов. Для получения положительной оценки «зачтено» необходимо набрать не менее 30 баллов.

Оценка «зачтено» выставляется студенту, если он показал всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка «не зачтено» выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач.

Составляющие процесса обучения, которые оцениваются в ходе обучения, и их вклад в зачет представлены ниже:

Модуль	Вклад в зачет, %
Оценка за модуль 1	30
Оценка за модуль 2	30
Зачет	40

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Во время проведения зачета обучающиеся могут пользоваться программой дисциплины.