

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
СТАВРОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ**

**УТВЕРЖДАЮ**

Директор/Декан  
факультета цифровых технологий  
Аникуев Сергей Викторович

«\_\_» \_\_\_\_\_ 20\_\_ г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (ОЦЕНОЧНЫХ МАТЕРИАЛОВ)**

**Б1.О.20 Основы программирования**

**09.03.02 Информационные системы и технологии**

Системы искусственного интеллекта

бакалавр

очная

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций ОП ВО и овладение следующими результатами обучения по дисциплине:

| Код и наименование компетенции  | Код и наименование индикатора достижения   | Перечень планируемых результатов обучения по дисциплине  |
|---|--|--|
| ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий | ОПК-6.1 Использует методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий | <b>знает</b><br>методы алгоритмизации, языки и технологии программирования, применимые в области информационных систем и технологий  |
|   |  | <b>умеет</b><br>использовать методы алгоритмизации, языки и технологии программирования для практического применения в сфере информационных систем                                       |
|   |  | <b>владеет навыками</b><br>методами алгоритмизации, языками и технологиями программирования для решения практических задач в области информационных систем и технологий                  |
| ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий | ОПК-6.2 Применяет методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий       | <b>знает</b><br>методы алгоритмизации, языки и технологии программирования, необходимые для решения профессиональных задач в области информационных систем и технологий                  |
|   |  | <b>умеет</b><br>применять методы алгоритмизации, языки и технологии программирования для решения профессиональных задач в сфере информационных систем и технологий                       |
|   |  | <b>владеет навыками</b><br>методами алгоритмизации, языками и технологиями программирования для эффективного решения профессиональных задач в области информационных систем и технологий |
| ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий | ОПК-6.3 Применяет методы программирования, отладки и тестирования прототипов программно-технических комплексов   | <b>знает</b><br>методы программирования, отладки и тестирования прототипов программно-технических комплексов   |
|   |  | <b>умеет</b><br>применять методы программирования, отладки и тестирования прототипов программно-технических комплексов   |
|   |  | <b>владеет навыками</b><br>методами программирования, отладки и тестирования прототипов программно-технических комплексов для решения профессиональных задач                             |

## 2. Перечень оценочных средств по дисциплине

| №    | Наименование раздела/темы                   | Семестр | Код индикаторов достижения компетенций | Оценочное средство проверки результатов достижения индикаторов компетенций |
|------|---|---------|--|--|
| 1.   | 1 раздел. Основы программирования           |         |  |  |
| 1.1. | Базовые конструкции Python                  | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.2. | Коллекции и работа с памятью                | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.3. | КТ 1  | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.4. | Функции и их особенности в Python           | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.5. | Объектно-ориентированное программирование   | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.6. | КТ 2  | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
| 1.7. | Библиотеки для получения и обработки данных | 1       | ОПК-6.1, ОПК-6.2, ОПК-6.3              | Задачи   |
|      | Промежуточная аттестация                    |         |  | Эк   |

## 3. Оценочные средства (оценочные материалы)

Примерный перечень оценочных средств для текущего контроля успеваемости и промежуточной аттестации

| № п/п            | Наименование оценочного средства | Краткая характеристика оценочного средства | Представление оценочного средства в фонде (Оценочные материалы) |
|------------------|----------------------------------|--|---|
| Текущий контроль |                                  |  |   |
|                  | Для оценки знаний                |  |   |

|   |                          |  |                                    |
|---|--------------------------|--|------------------------------------|
| 1 | Задачи                   | Задачи репродуктивного уровня, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и правильное использование специальных терминов и понятий, узнавание объектов изучения в рамках определенного раздела дисциплины; | Комплект задач минимального уровня |
|   | Для оценки умений        |  |                                    |
|   | Для оценки навыков       |  |                                    |
|   | Промежуточная аттестация |  |                                    |
| 2 | Экзамен                  | Средство контроля усвоения учебного материала и формирования компетенций, организованное в виде беседы по билетам с целью проверки степени и качества усвоения изучаемого материала, определить необходимость введения изменений в содержание и методы обучения.           | Комплект экзаменационных билетов   |

#### 4. Примерный фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю) "Основы программирования"

##### *Примерные оценочные материалы для текущего контроля успеваемости*

Знания

Задание 1

Что такое переменная в программировании?

- а) Постоянное значение, которое нельзя изменить
- б) Именованная область памяти для хранения данных
- в) Математическая переменная в уравнениях
- г) Тип данных для чисел с плавающей точкой

Ответ: б

Задание 2

Какие из перечисленных типов данных являются целочисленными? (выберите несколько вариантов)

- а) int
- б) float
- в) char
- г) double

Ответ: а, в

Задание 3

Что такое алгоритм?

- а) Язык программирования
- б) Среда разработки
- в) Конечная последовательность шагов для решения задачи
- г) Математическая формула

Ответ: в

#### Задание 4

Какие из этих операторов относятся к операторам сравнения? (выберите несколько вариантов)

- а) =
- б) ==
- в) !=
- г) +=

Ответ: б, в

#### Задание 5

Что выведет этот код?

- а) 2
- б) 2.5
- в) 3
- г) Ошибку

Ответ: б

#### Задание 6

Какой тип цикла гарантирует одно выполнение тела цикла?

- а) for
- б) while
- в) do-while
- г) foreach

Ответ: в

#### Задание 7

Что такое массив?

- а) Переменная для хранения одного значения
- б) Структура данных для хранения набора элементов одного типа
- в) Функция для сортировки данных
- г) Математическая матрица

Ответ: б

#### Задание 8

Какие из этих понятий относятся к ООП? (выберите несколько вариантов)

- а) Инкапсуляция
- б) Компиляция
- в) Наследование
- г) Полиморфизм

Ответ: а, в, г

#### Задание 9

Что такое функция в программировании?

- а) Математическая функция
- б) Именованный блок кода, который можно вызывать
- в) Тип данных
- г) Оператор цикла

Ответ: б

#### Задание 10

Какой оператор используется для выполнения действия при истинном условии?

- а) if
- б) for
- в) while
- г) switch

Ответ: а

#### Задание 11

Что такое компилятор?

- а) Программа для написания кода
- б) Программа, переводящая код в машинные команды
- в) Библиотека функций
- г) Отладчик программ

Ответ: б

Задание 12

Какие из этих структур данных являются линейными? (выберите несколько вариантов)

- а) Массив
- б) Стек
- в) Дерево
- г) Очередь

Ответ: а, б, г

Задание 13

Что такое рекурсия?

- а) Бесконечный цикл
- б) Вызов функцией самой себя
- в) Ошибка компиляции
- г) Тип данных

Ответ: б

Задание 14

Какой принцип ООП описывает сокрытие внутренней реализации?

- а) Наследование
- б) Инкапсуляция
- в) Полиморфизм
- г) Абстракция

Ответ: б

Задание 15

Что такое IDE?

- а) Язык программирования
- б) Интегрированная среда разработки
- в) База данных
- г) Операционная система

Ответ: б

Задание 16

Какие из этих типов данных являются вещественными? (выберите несколько вариантов)

- а) int
- б) float
- в) double
- г) char

Ответ: б, в

Задание 17

Что такое отладка (debugging)?

- а) Написание кода
- б) Процесс поиска и исправления ошибок
- в) Компиляция программы
- г) Тестирование программы

Ответ: б

Задание 18

Какой оператор цикла используется, когда известно количество итераций?

- а) while
- б) do-while
- в) for
- г) if

Ответ: в

Задание 19

Что такое логический тип данных?

- а) Может хранить только числа
- б) Может хранить true/false значения
- в) Может хранить текст
- г) Может хранить дробные числа

Ответ: б

Задание 20

Какие из этих операторов относятся к арифметическим? (выберите несколько вариантов)

а) +

б) &&

в) \*

г) ||

Ответ: а, в

Умения

Задание 1

Установите соответствие между типами данных и их описаниями:

Тип данных      Описание

1. int      А. Символьный тип

2. float    Б. Целочисленный тип

3. char    В. Логический тип

4. bool    Г. Число с плавающей точкой

Правильный ответ: 1-Б, 2-Г, 3-А, 4-В

Задание 2

Установите соответствие между операторами и их назначением:

Оператор      Назначение

1. =      А. Сравнение на равенство

2. ==    Б. Присваивание

3. !=    В. Логическое И

4. &&    Г. Сравнение на неравенство

Правильный ответ: 1-Б, 2-А, 3-Г, 4-В

Задание 3

Установите соответствие между видами циклов и их характеристиками:

Цикл      Характеристика

1. for      А. Проверяет условие после выполнения тела

2. while    Б. Используется когда известно количество итераций

3. do-while      В. Проверяет условие перед выполнением тела

Правильный ответ: 1-Б, 2-В, 3-А

Задание 4

Установите соответствие между концепциями ООП и их определениями:

Концепция      Определение

1. Инкапсуляция      А. Создание новых классов на основе существующих

2. Наследование      Б. Скрытие внутренней реализации

3. Полиморфизм      В. Разные реализации методов с одинаковым именем

4. Абстракция      Г. Выделение основных характеристик объекта

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 5

Установите соответствие между структурами данных и их описаниями:

Структура      Описание

1. Массив      А. Коллекция элементов с доступом по ключу

2. Стек    Б. Элементы располагаются последовательно в памяти

3. Очередь      В. LIFO (последним пришел - первым ушел)

4. Словарь      Г. FIFO (первым пришел - первым ушел)

Правильный ответ: 1-Б, 2-В, 3-Г, 4-А

Задание 6

Установите соответствие между типами ошибок и их примерами:

Тип ошибки      Пример

1. Синтаксическая      А. Деление на ноль во время выполнения

2. Логическая      Б. Отсутствие точки с запятой

3. Ошибка времени выполнения      В. Неправильная формула расчета

Правильный ответ: 1-Б, 2-В, 3-А

Задание 7

Установите соответствие между арифметическими операторами и их результатом:

| Выражение | Результат |
|-----------|-----------|
|-----------|-----------|

- |                  |       |
|------------------|-------|
| 1. $5 + 3 * 2$   | А. 11 |
| 2. $(5 + 3) * 2$ | Б. 16 |
| 3. $10 / 3$      | В. 3  |
| 4. $10 \% 3$     | Г. 1  |

Правильный ответ: 1-А, 2-Б, 3-В, 4-Г

Задание 8

Установите соответствие между понятиями и их описаниями:

| Понятие | Описание |
|---------|----------|
|---------|----------|

- |               |                                    |
|---------------|------------------------------------|
| 1. Переменная | А. Постоянное значение             |
| 2. Константа  | Б. Именованная область памяти      |
| 3. Функция    | В. Блок кода с именем              |
| 4. Массив     | Г. Коллекция элементов одного типа |

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 9

Установите соответствие между логическими операторами и их результатом:

| Выражение | Результат (true/false) |
|-----------|------------------------|
|-----------|------------------------|

- |                           |          |
|---------------------------|----------|
| 1. $true \&\& false$      | А. true  |
| 2. $true \parallel false$ | Б. false |
| 3. $!true$                | В. false |
| 4. $!false$               | Г. true  |

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 10

Установите соответствие между этапами разработки и их содержанием:

| Этап | Содержание |
|------|------------|
|------|------------|

- |                   |                           |
|-------------------|---------------------------|
| 1. Анализ         | А. Написание кода         |
| 2. Проектирование | Б. Определение требований |
| 3. Реализация     | В. Создание архитектуры   |
| 4. Тестирование   | Г. Поиск ошибок           |

Правильный ответ: 1-Б, 2-В, 3-А, 4-Г

Задание 11

Расставьте в правильном порядке приоритет выполнения операторов (от высшего к низшему):

- А. +, -
- Б. \*, /
- В. ()
- Г. =

Правильный порядок: В → Б → А → Г

Задание 12

Расставьте в правильном порядке этапы выполнения программы:

- А. Компиляция
- Б. Написание кода
- В. Выполнение
- Г. Линковка

Правильный порядок: Б → А → Г → В

Задание 13

Расставьте в правильном порядке жизненный цикл переменной:

- А. Использование
- Б. Объявление
- В. Инициализация
- Г. Уничтожение

Правильный порядок: Б → В → А → Г

Задание 14

Расставьте в правильном порядке выполнение операторов в выражении:

- А.  $3 * 4$
- Б.  $2 + 12$



В. 8 / 2

Г. 4 - 6

Выражение:  $8 / 2 + 3 * 4 - 6$

Правильный порядок:  $B \rightarrow A \rightarrow B \rightarrow G$

Задание 15

Расставьте в правильном порядке этапы работы цикла for:

А. Выполнение тела цикла

Б. Проверка условия

В. Изменение счетчика

Г. Инициализация счетчика

Правильный порядок:  $G \rightarrow B \rightarrow A \rightarrow B$

Задание 16

Расставьте в правильном порядке процесс вызова функции:

А. Возврат значения

Б. Вызов функции

В. Выполнение тела функции

Г. Передача параметров

Правильный порядок:  $B \rightarrow G \rightarrow B \rightarrow A$

Задание 17

Расставьте в правильном порядке процесс создания массива:

А. Выделение памяти

Б. Объявление массива

В. Инициализация элементов

Г. Определение размера

Правильный порядок:  $B \rightarrow G \rightarrow A \rightarrow B$

Задание 18

Расставьте в правильном порядке этапы обработки исключения:

А. Выполнение блока catch

Б. Генерация исключения

В. Нормальное продолжение выполнения

Г. Поиск обработчика

Правильный порядок:  $B \rightarrow G \rightarrow A \rightarrow B$

Задание 19

Расставьте в правильном порядке процесс создания объекта в ООП:

А. Вызов конструктора

Б. Объявление переменной

В. Выделение памяти

Г. Инициализация полей

Правильный порядок:  $B \rightarrow B \rightarrow A \rightarrow G$

Задание 20

Расставьте в правильном порядке процесс работы со стеком:

А. Добавление элемента (push)

Б. Проверка на пустоту

В. Удаление элемента (pop)

Г. Чтение верхнего элемента (peek)

Правильный порядок:  $B \rightarrow A \rightarrow G \rightarrow B$

Навыки

Задание 1

Верно ли утверждение: В языке Java все переменные должны быть объявлены с указанием их типа данных.

Ответ: Верно

Задание 2

Верно ли утверждение: Оператор '==' и метод equals() в Java всегда работают одинаково для сравнения строк.

Ответ: Неверно

Задание 3

Верно ли утверждение: Массив в программировании может содержать элементы разных типов данных.

Ответ: Неверно

Задание 4

Верно ли утверждение: Цикл 'while' проверяет условие перед выполнением тела цикла, а цикл 'do-while' - после.

Ответ: Верно

Задание 5

Верно ли утверждение: Функция всегда должна возвращать значение.

Ответ: Неверно

Задание 6

Верно ли утверждение: В объектно-ориентированном программировании класс является шаблоном для создания объектов.

Ответ: Верно

Задание 7

Верно ли утверждение: Рекурсивная функция обязательно должна иметь базовый случай (условие выхода).

Ответ: Верно

Задание 8

Верно ли утверждение: Переменная, объявленная внутри функции, доступна в любой другой функции программы.

Ответ: Неверно

Задание 9

Верно ли утверждение: В двоичной системе счисления используются только цифры 0 и 1.

Ответ: Верно

Задание 10

Верно ли утверждение: Компилятор и интерпретатор - это одно и то же.

Ответ: Неверно

Задание 11

Как называется ошибка, когда программа пытается обратиться к элементу массива по индексу, который выходит за его границы?

Ответ: Выход за границы массива

Задание 12

Сколько бит занимает один байт?

Ответ: 8

Задание 13

Как называется структура данных, работающая по принципу "последним пришел - первым ушел" (LIFO)?

Ответ: Стек

Задание 14

Какой оператор в Java используется для получения остатка от деления?

Ответ: %

Задание 15

Как называется процесс поиска и исправления ошибок в программе?

Ответ: Отладка

Задание 16

Сколько значений может хранить переменная логического типа (boolean)?

Ответ: 2

Задание 17

Как называется ошибка, когда программа выполняется, но выдает неправильный результат из-за ошибки в алгоритме?

Ответ: Логическая ошибка

Задание 18

Какой тип данных используется для хранения одного символа в Java?

Ответ: char

Задание 19

Как называется специальный метод класса, который вызывается при создании нового объекта?

Ответ: Конструктор

Задание 20

Какой оператор цикла в Java гарантирует одно выполнение тела цикла?

Ответ: do-while

***Примерные оценочные материалы  
для проведения промежуточной аттестации (зачет, экзамен)  
по итогам освоения дисциплины (модуля)***

1. Язык программирования Python: описание и философия.
2. Исходный код на языке Python: кодировка, физические и логические строки, блоки кода
3. Выражения в языке Python.
4. Идентификаторы, пространства имен и области видимости
5. Управляющие конструкции: операторы выбора и цикла.
6. Обработка исключений
7. Функции в языке Python. Лямбда-выражения.
8. Встроенные типы: целочисленный, вещественный, комплексный, логический
9. Последовательности. Кортежи.
10. Последовательности. Списки. Срезы.
11. Последовательности. Словари.
12. Множества и операции над ними
13. Файлы и операции над ними
14. Стиль программирования: описание и назначение.
15. Модули и пакеты
16. Обзор стандартной библиотеки. Модуль sys
17. Обзор стандартной библиотеки. Модуль copy
18. Обзор стандартной библиотеки. Модуль os
19. Обзор стандартной библиотеки. Модуль math
20. Обзор стандартной библиотеки. Модуль random
21. Функции преобразования типов
22. Функции ввода-вывода
23. Функциональное программирование: определение и основные элементы
24. Виды параметров функций в Python'e. Параметры по умолчанию
25. Функции как параметры и как результат
26. Декораторы
27. Функции для обработки последовательностей: range, xrange, map, filter
28. Функции для обработки последовательностей: sum, reduce zip
29. Списковые включения
30. Генераторы
31. Генераторные выражения
32. Итераторы. Функции iter, enumerate, sorted.
33. Модуль itertools. Функции itertools.chain, itertools.repeat, itertools.count.
34. Объектно-ориентированное программирование в Python. Объявление класса
35. Объектно-ориентированное программирование в Python: атрибуты, свойства, сокрытие данных

***Темы письменных работ (эссе, рефераты, курсовые работы и др.)***

Список тем для рефератов по дисциплине "Основы программирования", разделенный по категориям сложности и направленности. Темы варьируются от базовых до более сложных, что позволяет выбрать вариант в зависимости от уровня подготовки.

**Категория 1: Базовые концепции и синтаксис**

Эти темы идеально подходят для демонстрации понимания фундаментальных принципов Python.

Сравнительный анализ типов данных в Python: изменяемые и неизменяемые объекты.

Раскрыть тему: Рассмотреть числа, строки, списки, кортежи, словари и множества. Объяснить

разницу между ними, последствия их использования (например, при передаче в функции) и практические примеры применения каждого.

Управляющие конструкции в Python: условные операторы и циклы.

Раскрыть тему: Подробно разобрать if, elif, else, циклы for и while. Показать их синтаксис, особенности (например, else у циклов) и привести примеры решения типовых задач (поиск элемента, подсчет суммы и т.д.).

Функции в Python: от простых до рекурсивных.

Раскрыть тему: Объяснить объявление и вызов функций, параметры (позиционные, именованные, по умолчанию), возвращаемые значения. Отдельно осветить тему рекурсии, ее плюсы и минусы, сравнить с итеративным подходом на примере (например, вычисление факториала, чисел Фибоначчи).

Работа с исключениями в Python: конструкция try-except-else-finally.

Раскрыть тему: Объяснить философию обработки ошибок в Python. Разобрать иерархию встроенных исключений, показать, как создавать собственные классы исключений, и как правильно использовать блоки try-except для написания устойчивого к ошибкам кода.

Категория 2: Углубленное изучение возможностей Python

Темы для тех, кто хочет показать понимание более сложных аспектов языка.

Объектно-ориентированное программирование (ООП) в Python: классы, наследование, инкапсуляция, полиморфизм.

Раскрыть тему: Классическая тема для реферата. Объяснить четыре основных концепции ООП на практических примерах. Показать, как они реализуются в Python (магические методы, доступ к атрибутам, переопределение методов).

Списковые включения (List Comprehensions), генераторы и итераторы в Python.

Раскрыть тему: Показать элегантность и эффективность списковых включений по сравнению с традиционными циклами. Объяснить концепцию генераторов (ключевое слово yield) и их преимущество в потреблении памяти при работе с большими данными.

Работа с файлами в Python: текстовые и бинарные режимы, менеджер контекста (with).

Раскрыть тему: Описать основные режимы открытия файлов (r, w, a, b). Подробно остановиться на использовании оператора with для гарантированного закрытия файла и корректного управления ресурсами.

Модули и пакеты в Python: создание, импорт и структура проекта.

Раскрыть тему: Объяснить, как организовать код в модули и пакеты для его повторного использования и поддержки. Разобрать способы импорта, переменную \_\_name\_\_ и структуру типичного Python-проекта (например, с использованием setup.py/pyproject.toml).

Категория 3: Прикладное применение и популярные библиотеки

Темы, демонстрирующие, как Python используется для решения реальных задач.

Введение в научные вычисления с библиотеками NumPy и SciPy.

Раскрыть тему: Рассказать о назначении этих библиотек, их основных объектах (многомерные массивы ndarray в NumPy) и преимуществах перед стандартными типами данных Python для математических операций.

Библиотека Pandas для анализа данных: Series и DataFrame.

Раскрыть тему: Описать два ключевых структуры данных в Pandas. Показать на примере, как можно загружать данные из CSV-файла, производить их фильтрацию, агрегацию и очистку.

Создание графиков и визуализация данных с помощью библиотек Matplotlib и Seaborn.

Раскрыть тему: Сравнить эти две библиотеки. Показать, как строить основные типы графиков (линейные, столбчатые, гистограммы, scatter plot) и настраивать их внешний вид для эффективного представления информации.

Веб-скрапинг на Python: библиотеки Requests и BeautifulSoup.

Раскрыть тему: Объяснить основы парсинга веб-страниц: получение HTML-кода с помощью requests и его последующий разбор с помощью BeautifulSoup для извлечения нужной информации. Важно затронуть этические и юридические аспекты скрапинга.

Категория 4: Обзорные и концептуальные темы

Темы для рефератов, требующие анализа и обобщения информации.

Сравнительный анализ Python с другими языками программирования (например, C++, Java, JavaScript).

Раскрыть тему: Сравнить синтаксис, типизацию (статическая vs динамическая),

производительность, области применения и философию языков. Выделить сильные и слабые стороны Python.

Области применения Python: от веб-разработки до искусственного интеллекта.

Раскрыть тему: Дать обзор основных сфер, где используется Python: веб-разработка (Django, Flask), Data Science (Pandas, Scikit-learn), машинное обучение (TensorFlow, PyTorch), автоматизация задач и научные исследования.

Инструменты разработки для Python: среды разработки (PyCharm, VS Code), управление зависимостями (pip, venv).

Раскрыть тему: Рассказать о том, как правильно организовать рабочее окружение: использование виртуальных сред (venv) для изоляции проектов, менеджер пакетов pip, и обзор популярных IDE и их преимуществ.