### УЧЕБНОЕ ПОСОБИЕ ПО PYTHON для образовательных центров и самостоятельного обучения

### 

### 

### АВТОР: Мусоев Мухаммадчон

### ──────────────────────────────────────────────────

### ОСНОВЫ ПРОГРАММИРОВАНИЯ • Переменные и типы данных • Условия и циклы • Функции и модули • Работа со строками и коллекциями

### 

### 

### 

### 

ЧТО ВНУТРИ?

**📚 12 уроков с поурочным планированием**

**💻 Практические задания для каждого занятия**

**🎯 Мини-проекты после каждого урока**

**📱 Подходит для работы со смартфона**

**🏆 Сертификат об окончании курса**

ДЛЯ КОГО ЭТОТ УЧЕБНИК?  
• **Ученики образовательных центров  
• Студенты начинающих курсов  
• Самостоятельные обучающиеся**

ОТЗЫВЫ:  
**"Лучший учебник для начала пути в программировании!"**

### 

### 

### 

### 

### 

### 

### 

### 

### 

## 

## **📖 ОГЛАВЛЕНИЕ**

### **Введение**

1. Для кого этот учебник
2. Как работать с учебником
3. Установка Python и настройка среды

### **Модуль 1: Основы программирования (Недели 1-4)**

* Урок 1: Первые шаги — Среда и вывод данных
* Урок 2: Данные — Переменные и типы
* Урок 3: Математика в коде
* Урок 4: Интерактив — Ввод данных от пользователя

### **Модуль 2: Работа с текстом и логикой (Недели 5-7)**

* Урок 5: Обработка строк
* Урок 6: Логика — Операторы if/else
* Урок 7: Разветвление — elif и вложенные условия

### **Модуль 3: Коллекции данных (Недели 8-11)**

* Урок 8: Списки (List) — основы
* Урок 9: Цикл for и итерация
* Урок 10: Цикл while и управление потоком
* Урок 11: Словари (Dictionary)

### **Модуль 4: Функции и модульность (Неделя 12)**

* Урок 12: Функции и модули

### **Приложения**

* Шпаргалка по синтаксису
* Ответы к заданиям
* Глоссарий терминов

## 

## 

## 

## **📝 ВВЕДЕНИЕ**

### **Для кого этот учебник?**

Этот учебник создан для **абсолютных новичков** в программировании. Если вы:

* Никогда не писали код
* Хотите научиться программировать с нуля
* Ищете практический подход к обучению
* Предпочитаете учиться шаг за шагом

...то эта книга для вас!

### **Что вы узнаете?**

За 12 уроков вы освоите основы Python — одного из самых популярных языков программирования. Вы научитесь:

✅ Работать с данными разных типов

✅ Создавать интерактивные программы

✅ Принимать решения в коде

✅ Работать с коллекциями данных

✅ Писать собственные функции

### **Как работать с учебником?**

Формат каждого урока:

* **Цель урока** — что вы узнаете
* **Теория** — объяснение простыми словами
* **Практика** — задания для закрепления
* **Мини-проект** — применение знаний на практике
* **Дополнительный проект** — для тех, кто хочет большего

**Важные правила:**

* **🚫 Не копируйте код!** Набирайте его вручную — это лучше для запоминания.
* **💻 Практикуйтесь ежедневно** — даже 15-20 минут в день дают результат.
* **❓ Не бойтесь ошибок** — они лучшие учителя.
* **🔄 Повторяйте пройденное** — возвращайтесь к предыдущим урокам.

### **Что вам понадобится?**

* **Компьютер или смартфон** — можно заниматься даже с телефона!
* **Доступ в интернет** — для работы с онлайн-редактором кода.
* **Браузер** — Chrome, Firefox, Safari или другой.
* **Терпение и желание учиться** — самое главное!

## **🛠 УСТАНОВКА PYTHON И НАСТРОЙКА СРЕДЫ**

### **Вариант 1: Установка Python на компьютер**

Для программирования на языке Python необходимо установить бесплат- ную среду программирования у себя на компьютере, которую можно загру- зить по адресу: ***https://***[***www.python.org/downloads/.***](http://www.python.org/downloads/)

Вместе с Python на компьютер установится программа IDLE – среда разра- ботки для написания Python-программ.

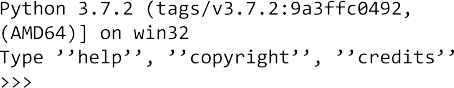
Для того чтобы записывать, сохранять и выполнять Python-команды, необ- ходимо сделать следующие шаги:

### **Шаг. Запустить IDLE**

Откроется окно консоли, в котором вы будете видеть результат вашей программы.



**1**

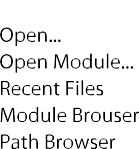
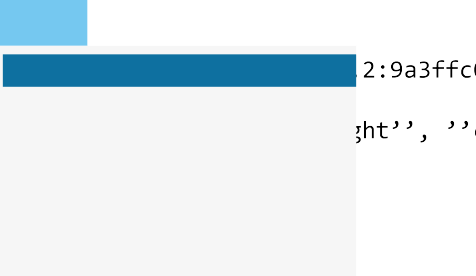


**2**



**Шаг. Создать новый файл**

Для записи новой программы необ- ходимо создать отдельный файл. Для этого в меню **File** выберите **New File**:



### **Шаг. Ввести программу**



**3**

В открывшемся окне введите вашу программу, например:

**print** (**'Salаm'**)

### **Шаг. Сохранить программу**



**4**

В меню **File** выберите **Save As**, задайте новое имя для вашего файла и на- жмите **Save**. Мы рекомендуем вам предварительно создать папку My scripts и сохранять туда все свои программы.

**Шаг. Запустить программу**



**5**



В меню **Run** выберите **Run Module** (либо можно просто нажать бы- струю клавишу F5)

Готово! В самой консоли вы должны увидеть сообщение:

>>>

Salаm!

### **Вариант 3: Использование смартфона**

1. Установите приложение **Pydroid 3** (Android) или **Pythonista** (iOS).
2. Откройте приложение и создайте новый файл.
3. Начните писать код!

## **🎯 СТРУКТУРА КУРСА**

|  |  |  |
| --- | --- | --- |
| **Неделя** | **Тема** | **Основные Навыки** |
| **1-3** | **Основы синтаксиса** | Переменные, типы данных, базовые операции |
| **4-7** | **Логика и взаимодействие** | Условия, ввод данных, обработка строк |
| **8-11** | **Коллекции и циклы** | Списки, словари, циклы for/while |
| **12** | **Функции** | Создание собственных функций |

# 

# 

# 

# **Модуль 1: Основы программирования**

# **УРОК 1: ПЕРВЫЕ ШАГИ - ЗНАКОМСТВО С PYTHON**

## **🎯 Цели урока:**

* Узнать, что такое Python и где он используется
* Настроить среду для работы
* Написать первую программу
* Понять основы синтаксиса

## **📚 Теория**

### **1. Что такое Python?**

Python — это язык программирования, который используют:

* YouTube — для обработки видео
* Google — для поисковых алгоритмов
* NASA — для научных расчетов
* Instagram — для веб-приложения

💡 **Почему Python идеален для тебя?**

* Простой и понятный синтаксис: Ты пишешь почти как на английском. print("Привет!") — это как сказать компьютеру: "Скажи 'Привет!'".
* Можно писать код даже на телефоне: Ты не ждёшь, пока купишь компьютер. Начни прямо сейчас!
* Большое сообщество: Ты не один. Миллионы людей учат Python, и ты найдёшь ответ на любой вопрос.

**💡 Почему это важно?**

Представь, что ты хочешь, чтобы твой смартфон напоминал тебе пить воду каждый час.  
Без кода ты должен вручную ставить будильник 24 раза в день.  
С Python ты напишешь одну строчку кода, и он будет делать это автоматически.  
print("Пей воду!") — это твой первый инструмент для автоматизации. Ты не просто пишешь текст — ты учишь машину делать работу за тебя. Это сила!

### **2. Где писать код?**

|  |  |  |
| --- | --- | --- |
| Платформа | Преимущества | Недостатки |
| Replit (онлайн) | Не требует установки, работает везде | Нужен интернет |
| IDLE (компьютер) | Работает офлайн, стандартная среда | Простой интерфейс |
| Pydroid (телефон) | Удобно для мобильных устройств | Маленький экран |

🎨 Визуальная схема: Типы данных в Python

|  |
| --- |
| ТИПЫ ДАННЫХ — ЭТО КАК ЯЩИКИ С НАКЛЕЙКАМИ  ┌─────────────┐ ┌─────────────┐ ┌─────────────┐  │ ЧИСЛА │ │ ТЕКСТ │ │ ЛОГИЧЕСКИЕ │  │ │ │ │ │ │  │ int │ │ str │ │ bool │  │ (целые) │ │ (любой текст)│ │ (Да/Нет) │  │ 25, -10 │ │ "Привет" │ │ True, False │  └───────┬─────┘ └───────┬─────┘ └───────┬─────┘  │ │ │  └───────┬───────────┴───────────┬───────┘  │ │  Ты кладёшь в ящик: Ты кладёшь в ящик:  25 яблок (число) "Привет!" (текст) |

**❓ Часто задаваемые вопросы (перед началом):**

* "Почему print(Привет) не работает, а print('Привет') работает?"  
  *Ответ:* Потому что Python думает, что Привет — это имя переменной, которую ты должен был создать. А Привет без кавычек — это просто слово, которое нужно вывести как текст. Кавычки говорят компьютеру: "Это не переменная, а просто текст!"
* "Зачем нужны комментарии? Код и так понятен!"  
  *Ответ:* Через неделю ты забудешь, что делал. Через месяц — твой учитель. Комментарии — это твои личные заметки, которые помогут тебе и другим понять, *почему* ты написал этот код. Это как закладка в книге.
* "Что будет, если я напишу pritn вместо print?"  
  *Ответ:* Компьютер скажет: "Я не знаю такой команды!" — и выведет ошибку. Это нормально! Ошибки — это твой лучший учитель.

## **💻 Практика**

### **Задание 1.1: Первая программа**

**Твоя задача:** Напиши код, который выводит твоё имя и фразу "Я начал изучать Python!"

**ВАЖНО:** Не копируй! Набирай его вручную.

|  |
| --- |
| python print("Я начал изучать Python!") print("Это мой первый код") |

### **🔬 Эксперимент!**

### Попробуй убрать кавычки: print(Я начал изучать Python!). Что выведет программа?

### *Запиши, что ты увидел, и почему так произошло.*

### **Задание 1.2: Эксперименты с выводом**

|  |
| --- |
| python print("Текст в двойных кавычках") print('Текст в одинарных кавычках') print("Комбинация 'разных' кавычек") |

💡 Совет: Попробуй использовать одинарные кавычки внутри строки, которая обернута в двойные, и наоборот. Это часто бывает полезно!

### **Задание 1.3: Комментарии**

|  |
| --- |
| python # Это комментарий - компьютер его игнорирует print("А это код") # Комментарий после кода  """ Многострочный комментарий для больших объяснений """ |

**Твоя задача:** Добавь комментарий к своей первой программе (из задания 1.1), который объясняет, *зачем* ты её написал.

Например: # Это моя первая программа! Я горжусь собой!

## **🚀 МИНИ-ПРОЕКТ**

"Визитная карточка"

|  |
| --- |
| python # ВИЗИТНАЯ КАРТОЧКА print("=" \* 30) print(" МОЯ ВИЗИТНАЯ КАРТОЧКА") print("=" \* 30) print("👤 Имя: Алексей") print("🎂 Возраст: 16 лет") print("🏠 Город: Москва") print("💻 Хобби: программирование") print("🐍 Изучаю: Python") print("=" \* 30) |

Результат:

|  |
| --- |
| text ==============================  МОЯ ВИЗИТНАЯ КАРТОЧКА ============================== 👤 Имя: Алексей 🎂 Возраст: 16 лет 🏠 Город: Москва 💻 Хобби: программирование 🐍 Изучаю: Python ============================== |

🎯 **Твоя задача:**

Замени "Алексей", "16", "Москва", "программирование" и "Python" на свои данные!

*Сейчас ты не просто копируешь чужой код — ты создаёшь свой первый цифровой продукт.*

Запусти код. Посмотри на результат. Это твоя визитка!

*Скопируй её в блокнот или отправь другу — ты уже программист!*

## **🔥 ДОПОЛНИТЕЛЬНЫЙ ПРОЕКТ**

"Генератор мотивационных цитат"

|  |
| --- |
| python # ГЕНЕРАТОР МОТИВАЦИОННЫХ ЦИТАТ print("🌟 ГЕНЕРАТОР МОТИВАЦИОННЫХ ЦИТАТ 🌟") print()  # Вывод нескольких цитат print("1. Каждый программист когда-то был новичком!") print("2. Ошибки -- это ступеньки к успеху!") print("3. Сегодня ты учишь Python, завтра -- создаешь технологии!") print("4. Код -- это суперсилка 21 века!") print("5. Ты на правильном пути! 🚀")  print() print("Верь в себя! 💪") |

🤔 **ВОПРОСЫ НА РАЗМЫШЛЕНИЕ**

1. Что будет, если написать print(Привет) без кавычек?  
   *Подсказка: Попробуй! Запиши ошибку, которую ты увидишь.*
2. Почему важно использовать одинаковые кавычки в начале и конце строки?  
   *Подсказка: Попробуй написать print("Привет').*
3. Зачем нужны комментарии в коде?  
   *Подсказка: Представь, что ты через месяц открыл этот код. Что тебе нужно, чтобы понять, что ты делал?*

## **❌ ЧАСТЫЕ ОШИБКИ**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ошибка** | **Неправильно** | **Правильно** | **Почему так?** |
| Нет кавычек | print(Привет) | print("Привет") | Компьютер ищет переменную "Привет", а не текст. |
| Разные кавычки | print("Привет') | print("Привет") | Кавычки должны быть парными. Это как открыть и закрыть дверь. |
| Опечатка | pritn("Текст") | print("Текст") | Компьютер не умеет читать мысли. Он видит "pritn" как несуществующую команду. |

## **📋 ШПАРГАЛКА**

|  |
| --- |
| *python # Вывод текста print("Любой текст в кавычках")  # Комментарии # Это комментарий - компьютер его игнорирует print("Код") # Комментарий после кода  # Разные кавычки print("Двойные") print('Одинарные')  # Спецсимволы print("Первая строка\nВторая строка") # Перевод строки* |

## **🏠 ДОМАШНЕЕ ЗАДАНИЕ**

### **Обязательная часть (для всех):**

1. Создайте программу, которая выводит ваше имя и 3 факта о себе
2. Напишите код, который выводит расписание ваших уроков на завтра

### **Для продвинутых (со звездочкой ⭐):**

Создайте "ASCII-арт" — рисунок из символов, например:

|  |
| --- |
| text  /\\_/\  ( o.o )  > ^ < |

## **🌍 PYTHON В РЕАЛЬНОЙ ЖИЗНИ**

Где используется Python:

* 🤖 Искусственный интеллект — распознавание лиц, голосовые помощники
* 🌐 Веб-сайты — Instagram, Pinterest, Spotify
* 📊 Анализ данных — прогнозы, статистика, графики
* 🎮 Игры — Minecraft, Civilization IV
* 📱 Мобильные приложения — Uber, Netflix

**🌟 История успеха:**

*Малика, 14 лет, из Худжанда, закончила этот курс за 3 недели. Она создала свой первый калькулятор чаевых, а потом — программу для расчета оценок в классе. Теперь она помогает учителю математики! "Я не думала, что смогу это сделать", — говорит Мария.*Ты тоже можешь!

**Интересный факт: Python назван не в честь змеи, а в честь комедийного шоу "Монти Пайтон"!**

## **✅ ТЕСТ ДЛЯ САМОПРОВЕРКИ**

1. Какой символ используется для однострочного комментария?
2. Что выведет код: print("5" + "5")?
3. Почему Python хорош для начинающих?

# **🎉 Поздравляем! Вы завершили первый урок!**

# **Ты сделал то, что большинство людей боятся попробовать. Ты написал код. Ты создал свою визитку. Ты почувствовал, что можешь управлять машиной.**

# **Твой путь в Python: [Новичок] → [Знаю, как вывести текст] → [Храню данные в переменных] → [Принимаю решения с if] → [Работаю со списками и циклами] → [Создаю функции] → [\*\*Программист!\*\*]**

# **Ты прошел 1/12 пути!**

# **Скоро ты узнаешь, как хранить эти слова в "умных коробках" — переменных.**

# **Ты уже не просто ученик. Ты — создатель.**

# 

# **УРОК 2: ПЕРЕМЕННЫЕ И ТИПЫ ДАННЫХ - ФУНДАМЕНТ ПРОГРАММИРОВАНИЯ**

## **🎯 Цели урока:**

* Научиться хранить данные в переменных
* Понять основные типы данных в Python
* Освоить работу с числами и текстом
* Создать первую интерактивную программу

## **📚 Теория**

### **1. Что такое переменная?**

Переменная — это как умная коробка с наклейкой:

* Имя переменной = надпись на коробке
* Значение = то, что лежит внутри
* Тип данных = что именно лежит в коробке (книга, яблоко, игрушка)

|  |
| --- |
| *python # Создание переменных имя\_ученика = "Анна" # Коробка с надписью "имя\_ученика" возраст = 16 # В ней лежит число 16 средний\_балл = 4.75 # Дробное число любит\_программирование = True # Логическое значение* |

### **💡 Почему это важно?**

### **Представьте, что вы пишете программу для школы в Душанбе, которая считает средний балл ученика. Без переменных вы бы каждый раз писали:**

|  |
| --- |
| **print("Средний балл: (5 + 4 + 5 + 4) / 4 = 4.5")** |

### **А с переменными — просто:**

|  |
| --- |
| **print("Средний балл:", средний\_балл)** |

### **Это как использовать имя ученика вместо "ребёнок №1" — понятно, удобно и профессионально!**

### **2. Таблица типов данных в Python**

|  |  |  |  |
| --- | --- | --- | --- |
| Тип данных | Пример | Описание | Использование |
| int | 25, -10, 0 | Целые числа | Возраст, количество, счёт |
| float | 3.14, -2.5, 0.0 | Дробные числа | Цены, рост, средний балл |
| str | "Привет", 'A' | Текст | Имена, сообщения, адреса |
| bool | True, False | Логический | Да/нет, включено/выключено |

### **3. Правила именования переменных**

|  |
| --- |
| *python # ✅ ПРАВИЛЬНО имя = "Мария" возраст\_ученика = 15 count\_students = 25 is\_active = True  # ❌ НЕПРАВИЛЬНО 1имя = "Иван" # Не может начинаться с цифры имя ученика = "Петр" # Не может содержать пробелы print = "текст" # Нельзя использовать ключевые слова* |

|  |  |  |
| --- | --- | --- |
| ✅ ПРАВИЛЬНО | ❌ НЕПРАВИЛЬНО | Почему? |
| имя\_ученика = "Рахмон" | 1имя = "Исмоил" | Не может начинаться с цифры |
| возраст\_ученика = 16 | имя ученика = "Алишер" | Нет пробелов в имени |
| средний\_балл = 4.8 | print = "Отлично!" | print — ключевое слово Python |
| is\_active = True | класс = 9 | Можно, но лучше class\_number (чтобы не путать с ключевым словом class) |

**💬 Совет: В Таджикистане часто используют русские имена и названия. Это нормально!**

Можно писать: имя = "Рахмон", город = "Душанбе", школа = "СШ №1 им. А. Рудаки" — всё на кириллице работает в Python!

## **🎨 Визуальная схема: Как работают переменные**

|  |
| --- |
| text ПАМЯТЬ КОМПЬЮТЕРА ┌─────────────┬─────────────┐ │ Имя │ Значение │ Тип │ ├─────────────┼─────────────┼────────┤ │ name │ "Анна" │ str │ │ age │ 16 │ int │ │ height │ 1.65 │ float │ │ is\_student │ True │ bool │ └─────────────┴─────────────┴────────┘ |

## **💻 Практика**

### **Задание 2.1: Создание профиля пользователя**

|  |
| --- |
| python # Создаем переменные для профиля user\_name = "Дмитрий" user\_age = 17 user\_height = 1.78 favorite\_subject = "информатика" has\_pet = True  print("=== ПРОФИЛЬ ===") print("Имя:", user\_name) print("Возраст:", user\_age) print("Рост:", user\_height) print("Любимый предмет:", favorite\_subject) print("Есть питомец:", has\_pet) |

### **📌 Ваше задание: Замените данные на свои! Например: имя\_ученика = "Алишер" город = "Худжанд" школа = "Гимназия №5" любимый\_предмет = "математика"**

### **Запустите код. Это — ваш профиль!**

### **Задание 2.2: Изменение значений**

|  |
| --- |
| python # Переменные можно менять! score = 0 print("Начальный счёт:", score)  score = 10 # Меняем значение print("Новый счёт:", score)  score = score + 5 # Увеличиваем на 5 print("После бонуса:", score) |

### **🔍 Эксперимент:**

### **Попробуйте изменить счет\_баллов = счет\_баллов + 5 на счет\_баллов += 5 — результат тот же! Это сокращённая запись.**

### **Задание 2.3: Работа с разными типами**

|  |
| --- |
| python # Преобразование типов number\_text = "100" number = int(number\_text) # Преобразуем строку в число result = number + 50  print("Исходная строка:", number\_text) print("После преобразования:", number) print("Результат сложения:", result) |

## **🚀 МИНИ-ПРОЕКТ**

"Умный школьный дневник"

|  |
| --- |
| python # УМНЫЙ ШКОЛЬНЫЙ ДНЕВНИК print("📚 УМНЫЙ ШКОЛЬНЫЙ ДНЕВНИК 📚") print("=" \* 35)  # Данные ученика student\_name = "София" class\_number = 10 school = "Гимназия №1"  # Оценки по предметам math\_grade = 5 physics\_grade = 4 history\_grade = 5 english\_grade = 4  # Расчет среднего балла average\_grade = (math\_grade + physics\_grade + history\_grade + english\_grade) / 4  # Вывод информации print("Ученик:", student\_name) print("Класс:", class\_number) print("Школа:", school) print() print("📊 УСПЕВАЕМОСТЬ:") print("Математика:", math\_grade) print("Физика:", physics\_grade) print("История:", history\_grade) print("Английский:", english\_grade) print() print("⭐ СРЕДНИЙ БАЛЛ:", average\_grade) print("=" \* 35) |

Результат:

|  |
| --- |
| text 📚 УМНЫЙ ШКОЛЬНЫЙ ДНЕВНИК 📚 =================================== Ученик: София Класс: 10 Школа: Гимназия №1  📊 УСПЕВАЕМОСТЬ: Математика: 5 Физика: 4 История: 5 Английский: 4  ⭐ СРЕДНИЙ БАЛЛ: 4.5 =================================== |

**✅ Ваша задача:**  
Замените всё на свои данные!

* Имя: Абдулло, Мария, Фархад
* Город: Худжанд, Курган-Тюбе, Гарм, Куляб
* Школа: СШ №12, Лицей №1, Гимназия им. Ф. Таджиева
* Оценки: поставьте свои!

Запустите. Посмотрите. Это — ваш дневник!

## **🔥 ДОПОЛНИТЕЛЬНЫЙ ПРОЕКТ**

"Инвентарь компьютерной игры"

|  |
| --- |
| python # ИНВЕНТАРЬ КОМПЬЮТЕРНОЙ ИГРЫ print("🎮 ИНВЕНТАРЬ ИГРОКА 🎮") print("=" \* 40)  # Предмет 1: Меч item1\_name = "Стальной меч" item1\_attack = 15 item1\_durability = 100 item1\_price = 250.50  # Предмет 2: Зелье item2\_name = "Зелье здоровья" item2\_heal = 50 item2\_quantity = 3 item2\_price = 75.25  # Предмет 3: Щит item3\_name = "Драконий щит" item3\_defense = 20 item3\_durability = 85 item3\_price = 180.75  # Вывод инвентаря print("⚔️ ПРЕДМЕТ 1:") print(" Название:", item1\_name) print(" Атака:", item1\_attack) print(" Прочность:", item1\_durability) print(" Цена:", item1\_price, "золотых")  print("\n🧪 ПРЕДМЕТ 2:") print(" Название:", item2\_name) print(" Лечение:", item2\_heal) print(" Количество:", item2\_quantity) print(" Цена:", item2\_price, "золотых")  print("\n🛡️ ПРЕДМЕТ 3:") print(" Название:", item3\_name) print(" Защита:", item3\_defense) print(" Прочность:", item3\_durability) print(" Цена:", item3\_price, "золотых")  # Общая стоимость инвентаря total\_value = item1\_price + item2\_price + item3\_price print("\n💰 ОБЩАЯ СТОИМОСТЬ ИНВЕНТАРЯ:", total\_value, "золотых") print("=" \* 40) |

✅ Ваша задача:  
Добавьте ещё один предмет — например, "Книга 'Рудаки'" или "Колчан 'Афросиаб'".  
Придумайте ему название, характеристики и цену в сомони.

Сделайте так, чтобы ваш инвентарь отражал ваш интерес:

* Любите историю? → "Книга 'История Таджикистана'"
* Любите спорт? → "Кроссовки 'Памир'"
* Любите музыку? → "Уд 'Рудаки'"

Это ваш личный инвентарь!

## **🤔 ВОПРОСЫ НА РАЗМЫШЛЕНИЕ**

1. Что произойдет, если попытаться сложить строку и число без преобразования?
2. Почему важно давать переменным понятные имена?
3. В чем разница между age = "16" и age = 16?

## **❌ ТАБЛИЦА ЧАСТЫХ ОШИБОК**

|  |  |  |
| --- | --- | --- |
| Ошибка | Пример | Решение |
| Неправильное имя | 1st\_place = "Я" | first\_place = "Я" |
| Смешение типов | "Возраст: " + 16 | "Возраст: " + str(16) |
| Опечатка | user\_nmae = "Аня" | user\_name = "Аня" |
| Использование ключевых слов | print = "текст" | output\_text = "текст" |

## **📋 ШПАРГАЛКА**

|  |
| --- |
| python # Создание переменных name = "Текст" # Строка (str) age = 20 # Целое число (int) price = 99.99 # Дробное число (float) is\_active = True # Логический тип (bool)  # Изменение значений age = age + 1 # Увеличить на 1 name = "Новое имя" # Поменять текст  # Преобразование типов str(25) # Число → строка "25" int("100") # Строка → число 100 float("3.14") # Строка → дробное 3.14  # Вывод нескольких переменных print("Имя:", name, "Возраст:", age) |

## **🏠 ДОМАШНЕЕ ЗАДАНИЕ**

Обязательная часть (для всех):

1. "Профиль соцсети" — создайте переменные для профиля в соцсети:
   * никнейм: "Рахмон\_Таджик"
   * количество подписчиков: 1250
   * есть ли верификация: True
   * рейтинг: 4.8  
     Выведите красиво:  
     >>> Ник: Рахмон\_Таджик | Подписчики: 1250 | ✅ Верифицирован | Рейтинг: 4.8
2. "Школьный калькулятор" — сохраните оценки за 5 предметов (например: математика, физика, русский, химия, информатика) и вычислите средний балл.  
   Используйте свои реальные оценки из тетради!

Для продвинутых (⭐):  
"Система учета книг в библиотеке" — создайте программу для учета 3 книг с информацией:

* название ("Саади: Гулистан")
* автор ("Саади Ширази")
* год издания (1998)
* есть ли в наличии (True)  
  Вычислите общее количество книг и выведите:  
  >>> В библиотеке 3 книги. Самая старая — "Саади: Гулистан" (1998).

## **🌍 PYTHON В РЕАЛЬНОЙ ЖИЗНИ**

Где используются переменные и типы данных:

* 🏦 Банковские приложения — хранение баланса, личных данных
* 🛒 Интернет-магазины — цены товаров, количество на складе
* 📱 Социальные сети — профили пользователей, посты, лайки
* 🎮 Игры — здоровье персонажа, очки, инвентарь
* 📊 Аналитика — статистические данные, отчеты

💡 Интересный факт:

В Таджикистане, как и везде, все цифры и тексты в компьютерах хранятся как переменные.

Когда вы смотрите расписание уроков в школе на телефоне — это список переменных.

Когда вы платите за интернет через "Телеком" — это вычисления с переменными.

Вы уже используете Python, даже не зная!

## **✅ ТЕСТ ДЛЯ САМОПРОВЕРКИ**

1. Какой тип данных у переменной temperature = -5.5?
2. Что выведет код: x = "10"; y = 5; print(x + str(y))?
3. Почему userName лучше чем usernm?

**🎉 Потрясающе! Вы освоили основы хранения данных — теперь вы умеете работать с информацией, как настоящий программист из Таджикистана!**

**Следующий урок — математика в коде. Будем считать скидки в магазинах Душанбе! 💪**

# 

# **УРОК 3: МАТЕМАТИКА В КОДЕ - СТАНЕМ КУЛЬТУРНЫМИ ВЫЧИСЛИТЕЛЯМИ!**

## **🎯 Цели урока:**

* **Освоить математические операции в Python**
* **Понять приоритет операций**
* **Научиться преобразовывать типы данных**
* **Создать полезные калькуляторы**

## **📚 Теория**

### **1. Математические операторы в Python**

|  |  |  |  |
| --- | --- | --- | --- |
| Оператор | Пример | Результат | Описание |
| **+** | **5 + 3** | **8** | **Сложение** |
| **-** | **10 - 4** | **6** | **Вычитание** |
| **\*** | **6 \* 7** | **42** | **Умножение** |
| **/** | **15 / 4** | **3.75** | **Обычное деление** |
| **//** | **15 // 4** | **3** | **Целочисленное деление** |
| **%** | **15 % 4** | **3** | **Остаток от деления** |
| **\*\*** | **2 \*\* 3** | **8** | **Возведение в степень** |

### **💡 Почему это важно?** Представьте, что вы едете на рынок в Душанбе и хотите купить:

### 3 кг картошки по 25 сомони за кг

### 2 кг лука по 40 сомони за кг

### 1 бутылку масла за 180 сомони

### Без математики в коде вы будете считать на пальцах. С Python вы напишете одну строку и получите точный итог — и даже скидку рассчитаете!

### **2. Приоритет операций (как в математике)**

**Python вычисляет выражения в определенном порядке:**

|  |
| --- |
| ***python # 1. Скобки () - самый высокий приоритет # 2. Возведение в степень \*\* # 3. Умножение \*, деление /, //, % # 4. Сложение +, вычитание -  result = 5 + 3 \* 2 # 11 (сначала умножение) result = (5 + 3) \* 2 # 16 (сначала скобки) result = 2 \*\* 3 + 1 # 9 (сначала степень)*** |

### 📌 **Пример из жизни: Скидка в магазине "Сомони-маркет" в Худжанде** Товар стоит 1200 сомони. Есть скидка 15%. **Неправильно:** 1200 \* 15 / 100 + 1200 → Это будет *цена с прибавленной скидкой*! **Правильно:** 1200 - (1200 \* 15 / 100) → Сначала считаем скидку, потом вычитаем её.

### Или ещё проще: 1200 \* (1 - 15/100) → Умножаем на 0.85 — это и есть цена со скидкой!

### **3. Преобразование типов для вычислений**

|  |
| --- |
| ***python # ❌ ОШИБКА - разные типы # print("Сумма: " + 100)  # ✅ ПРАВИЛЬНО - преобразуем типы print("Сумма: " + str(100))  # Преобразование ввода пользователя age = int(input("Сколько вам лет? ")) # str → int price = float(input("Цена товара: ")) # str → float*** |

💬 **Важно!**

В Таджикистане цены часто пишут с копейками (например, 250.50 сомони).

Используйте float(), а не int(), чтобы не терять точность!

## **🎨 Визуальная схема: Приоритет операций**

|  |
| --- |
| **text ВЫРАЖЕНИЕ: (5 + 3) \* 2 \*\* 2  ШАГИ ВЫЧИСЛЕНИЯ: 1. Скобки: (5 + 3) = 8 2. Степень: 2 \*\* 2 = 4  3. Умножение: 8 \* 4 = 32  ↓ РЕЗУЛЬТАТ: 32** |

📌 **Пример из жизни:**Вы купили:

* 500 сомони — молоко
* 200 сомони — яйца  
  И получили **скидку 15% на всю покупку**.  
  Какая итоговая сумма?  
  (500 + 200) \* (1 - 15/100) = 595 сомони

## **💻 Практика**

### **Задание 3.1: Базовые операции**

|  |
| --- |
| **python # Исследуем разные операции a = 15 + 7 b = 20 - 9 c = 6 \* 8 d = 45 / 9 e = 17 // 5 f = 17 % 5 g = 2 \*\* 5  print("Сложение:", a) print("Вычитание:", b) print("Умножение:", c) print("Деление:", d) print("Целое от деления:", e) print("Остаток от деления:", f) print("Степень:", g)**  **print("🛒 РАСЧЕТ ПОКУПКИ В БАКАЛЕЕ")**  **# Цена за единицу**  **цена\_яблок = 60 # сомони за кг**  **цена\_картошки = 25 # сомони за кг**  **цена\_лука = 40 # сомони за кг**  **цена\_масла = 180 # сомони за бутылку**  **# Количество**  **кг\_яблок = 2**  **кг\_картошки = 3**  **кг\_лука = 1**  **бутылок\_масла = 1**  **# Рассчитываем стоимость каждого товара**  **стоимость\_яблок = цена\_яблок \* кг\_яблок**  **стоимость\_картошки = цена\_картошки \* кг\_картошки**  **стоимость\_лука = цена\_лука \* кг\_лука**  **стоимость\_масла = цена\_масла \* бутылок\_масла**  **# Общая сумма без скидки**  **итого\_без\_скидки = стоимость\_яблок + стоимость\_картошки + стоимость\_лука + стоимость\_масла**  **print(f"🛒 Яблоки ({кг\_яблок} кг): {стоимость\_яблок} сомони")**  **print(f"🥔 Картошка ({кг\_картошки} кг): {стоимость\_картошки} сомони")**  **print(f"🧅 Лук ({кг\_лука} кг): {стоимость\_лука} сомони")**  **print(f"🫙 Масло ({бутылок\_масла} бут.): {стоимость\_масла} сомони")**  **print(f"💰 Итого без скидки: {итого\_без\_скидки} сомони")** |

### 📌 **Ваша задача: Замените цены и количества на свои реальные!** Сколько вы купили на рынке в **Гарме** или **Кулябе**? Попробуйте посчитать:

### 4 кг картошки по 20 сомони

### 2 кг лука по 45 сомони

### 1 кг яблок по 70 сомони

### 2 бутылки масла по 190 сомони

### **Задание 3.2: Приоритет операций**

|  |
| --- |
| **python # Сравниваем разные способы расчета result1 = 10 + 5 \* 2 # 20 result2 = (10 + 5) \* 2 # 30 result3 = 10 + 5 \* 2 \*\* 2 # 30 result4 = (10 + 5) \* 2 \*\* 3 # 120  print("10 + 5 \* 2 =", result1) print("(10 + 5) \* 2 =", result2) print("10 + 5 \* 2 \*\* 2 =", result3) print("(10 + 5) \* 2 \*\* 3 =", result4)** |

### **Задание 3.3: Работа с дробными числами**

|  |
| --- |
| **python # Расчет скидки в магазине original\_price = 1200.0 discount\_percent = 15 # 15% скидка  discount\_amount = original\_price \* (discount\_percent / 100) final\_price = original\_price - discount\_amount  print("Исходная цена:", original\_price, "руб.") print("Скидка:", discount\_percent, "%") print("Сумма скидки:", discount\_amount, "руб.") print("Итоговая цена:", final\_price, "руб.")** |

📌 **Ваша задача:**Запустите код и введите:

* Цена: 1500.50
* Скидка: 10  
  Посмотрите, как округляется итоговая цена.  
  **Это как в чеке в супермаркете — ровно 2 знака после запятой!**

## **🚀 МИНИ-ПРОЕКТ**

"Умный калькулятор чаевых"

|  |
| --- |
| **python # УМНЫЙ КАЛЬКУЛЯТОР ЧАЕВЫХ print("💰 УМНЫЙ КАЛЬКУЛЯТОР ЧАЕВЫХ 💰") print("=" \* 40)  # Ввод данных bill\_amount = float(input("Введите сумму счета: ")) people\_count = int(input("Сколько человек платят: ")) service\_quality = input("Качество обслуживания (хорошее/отличное): ")  # Расчет чаевых в зависимости от качества обслуживания if service\_quality.lower() == "отличное":  tip\_percent = 20 elif service\_quality.lower() == "хорошее":  tip\_percent = 15 else:  tip\_percent = 10  # Расчеты tip\_amount = bill\_amount \* (tip\_percent / 100) total\_amount = bill\_amount + tip\_amount amount\_per\_person = total\_amount / people\_count  # Вывод результатов print("\n" + "=" \* 40) print("📊 РЕЗУЛЬТАТЫ РАСЧЕТА:") print("Сумма счета:", bill\_amount, "руб.") print("Качество обслуживания:", service\_quality) print("Процент чаевых:", tip\_percent, "%") print("Сумма чаевых:", round(tip\_amount, 2), "руб.") print("ОБЩАЯ СУММА:", round(total\_amount, 2), "руб.") print("С каждого человека:", round(amount\_per\_person, 2), "руб.") print("=" \* 40)** |

Пример работы:

|  |
| --- |
| **text 💰 УМНЫЙ КАЛЬКУЛЯТОР ЧАЕВЫХ 💰 ======================================== Введите сумму счета: 1500 Сколько человек платят: 3 Качество обслуживания (хорошее/отличное): отличное  ======================================== 📊 РЕЗУЛЬТАТЫ РАСЧЕТА: Сумма счета: 1500.0 руб. Качество обслуживания: отличное Процент чаевых: 20 % Сумма чаевых: 300.0 руб. ОБЩАЯ СУММА: 1800.0 руб. С каждого человека: 600.0 руб. ========================================** |

📌 **Ваша задача:**Представьте, что вы с друзьями были в кафе "Рудаки" в Худжанде.  
Счет — 1800 сомони.  
Обслуживание — **отличное**.  
Человек — 4.

**Запустите код. Посмотрите, сколько должен заплатить каждый из вас.**

💡 **Реальная практика:**В Таджикистане чаевые не обязательны, но если вы довольны — 10–20% — это признание хорошей работы.  
**Сейчас вы умеете считать это точно!**

## **🔥 ДОПОЛНИТЕЛЬНЫЙ ПРОЕКТ**

"Система учета времени"

|  |
| --- |
| **python # СИСТЕМА УЧЕТА ВРЕМЕНИ print("⏰ СИСТЕМА УЧЕТА ВРЕМЕНИ ⏰") print("=" \* 45)  # Ввод общего количества секунд total\_seconds = int(input("Введите общее количество секунд: "))  # Расчет компонентов времени days = total\_seconds // (24 \* 3600) # Дни remaining\_seconds = total\_seconds % (24 \* 3600)  hours = remaining\_seconds // 3600 # Часы remaining\_seconds %= 3600  minutes = remaining\_seconds // 60 # Минуты seconds = remaining\_seconds % 60 # Секунды  # Вывод результата print("\n📅 РАЗБИВКА ВРЕМЕНИ:") print("Общее секунд:", total\_seconds) print("Дней:", days) print("Часов:", hours) print("Минут:", minutes) print("Секунд:", seconds)  # Дополнительные расчеты total\_minutes = total\_seconds // 60 total\_hours = total\_seconds // 3600  print("\n📊 АЛЬТЕРНАТИВНЫЕ ЕДИНИЦЫ:") print("Всего минут:", total\_minutes) print("Всего часов:", round(total\_hours, 2)) print("Всего дней:", round(total\_seconds / (24 \* 3600), 3)) print("=" \* 45)** |

📌 **Ваша задача:**Представьте, что вы с друзьями были в кафе "Рудаки" в Худжанде.  
Счет — 1800 сомони.  
Обслуживание — **отличное**.  
Человек — 4.

**Запустите код. Посмотрите, сколько должен заплатить каждый из вас.**

💡 **Реальная практика:**В Таджикистане чаевые не обязательны, но если вы довольны — 10–20% — это признание хорошей работы.  
**Сейчас вы умеете считать это точно!**

## **🤔 ВОПРОСЫ НА РАЗМЫШЛЕНИЕ**

1. В чем разница между / и //? Когда использовать каждый?
2. Что вернет выражение 2 + 3 \* 4 \*\* 2? Почему?
3. Как преобразовать строку "15.5" в дробное число для вычислений?

## **❌ ТАБЛИЦА ЧАСТЫХ ОШИБОК**

|  |  |  |  |
| --- | --- | --- | --- |
| Ошибка | Пример | Решение | Почему Так? |
| **Деление на ноль** | **10 / 0** | **Проверить делитель: if y != 0: x / y** | **Компьютер не может делить на ноль — будет ошибка!** |
| **Смешение типов** | **"Цена: " + 100** | **"Цена: " + str(100)** | **Строка + число — нельзя! Нужно преобразовать.** |
| **Неправильный приоритет** | **5 + 3 \* 2 (ожидает 16)** | **(5 + 3) \* 2** | **Без скобок сначала умножение!** |
| **Потеря точности** | **10 // 3 (для денег)** | **10 / 3 или round(10 / 3, 2)** | **Для денег нужна точность до копеек!** |

## **📋 ШПАРГАЛКА**

|  |
| --- |
| ***python # Арифметические операции a + b # Сложение a - b # Вычитание  a \* b # Умножение a / b # Деление (дробный результат) a // b # Целочисленное деление a % b # Остаток от деления a \*\* b # Возведение в степень  # Приоритет операций result = (5 + 3) \* 2 # Скобки → умножение  # Преобразование типов int("123") # "123" → 123 float("45.5") # "45.5" → 45.5 str(100) # 100 → "100"  # Полезные функции round(3.14159, 2) # Округление до 2 знаков: 3.14 abs(-10) # Модуль числа: 10*** |

## **🏠 ДОМАШНЕЕ ЗАДАНИЕ**

### **Обязательная часть (для всех):**

### **"Калькулятор оценок"**: Сохраните в переменные оценки за 5 предметов (математика, физика, русский, химия, информатика) — **свои реальные оценки**. Рассчитайте средний балл. Выведите результат с двумя знаками после запятой. *Пример:* Средний балл: 4.60

### **"Конвертер валют"**: Создайте программу для конвертации **рубля в сомони** и **доллар в сомони**. Используйте реальные курсы (например, 1 USD = 10.5 сомони, 1 RUB = 0.11 сомони). Запросите у пользователя сумму в рублях или долларах — и выведите эквивалент в сомони. *Пример:* Введите сумму в долларах: 50 → Это 525.00 сомони

### **Для продвинутых (⭐):**

### **"Система расчета зарплаты"**: Программа рассчитывает зарплату учителя в Таджикистане:

### Базовая ставка: 3500 сомони

### За каждый урок: +100 сомони

### За сверхурочные (после 8 уроков): +150 сомони за урок

### Премия: +500 сомони, если отработал все дни

### Запросите:

### Количество уроков

### Количество сверхурочных

### Были ли все дни отработаны? (да/нет)

### Выведите:

### Базовая зарплата

### За уроки

### За сверхурочные

### Премия

### **Итоговая зарплата**

### 💡 **Реальная ценность:** Теперь вы можете проверить, сколько вы заработаете за месяц — и понять, как работает ваша зарплата!

## **🌍 PYTHON В РЕАЛЬНОЙ ЖИЗНИ**

Где используется математика в программировании в Таджикистане:  
· 🏦 **Банки** — расчет процентов по вкладам, кредитам, комиссий за переводы  
· 🛒 **Интернет-магазины** — расчет скидок, доставки, итоговой суммы в "Таджик-Сити", "Корзина.тj"  
· 📊 **Школы и вузы** — расчет среднего балла, статистики успеваемости, рейтингов  
· 📱 **Мобильные приложения** — конвертер валют, калькуляторы, приложения для учета расходов  
· 🚚 **Логистика** — расчет расстояний, времени доставки, стоимости перевозки  
· 🧮 **Рынки** — продавцы используют калькуляторы, чтобы быстро посчитать вес, цену, скидку

💡 **Интересный факт:**В 2024 году в Таджикистане запустили **цифровые кассы** в супермаркетах.  
Каждый чек генерируется автоматически — с расчетом скидок, НДС и итоговой суммы.  
**Это работает на Python!**

**Вы уже знаете, как он считает.**

## **✅ ТЕСТ ДЛЯ САМОПРОВЕРКИ**

1. Что вернет выражение 17 // 5?
2. Какой тип данных у результата 15 / 3?
3. В каком порядке вычисляется: 3 + 4 \* 2 \*\* 2?

**🎉 Потрясающе! Теперь вы умеете проводить вычисления в Python!**

**Вы не просто считаете — вы учитесь считать как настоящий программист из ТАДЖИКИСТАНА.**

**Следующий урок — вы научитесь общаться с компьютером.  
Вы спросите: "Сколько стоит?" — а он ответит: "1200 сомони!"**

**Вы уже не просто ученик. Вы — создатель цифровых инструментов.**

# **УРОК 4: ИНТЕРАКТИВ - ВВОД ДАННЫХ И ФОРМАТИРОВАНИЕ**

## **🎯 Цели урока:**

* Научиться получать данные от пользователя
* Освоить разные способы форматирования вывода
* Создать truly интерактивные программы
* Понять обработку пользовательского ввода

## **📚 Теория**

### **1. Функция input() - ваш диалог с программой**

|  |
| --- |
| python # Простейший ввод name = input("Как вас зовут? ") print("Привет, " + name + "!")  # Числовой ввод (требует преобразования) age = int(input("Сколько вам лет? ")) height = float(input("Ваш рост (м): ")) |

### **2. Способы форматирования вывода**

Таблица методов форматирования:

|  |  |  |
| --- | --- | --- |
| Метод | Пример | Преимущества |
| Конкатенация | "Имя: " + name | Простота |
| f-строки | f"Имя: {name}" | Читаемость, скорость |
| format() | "Имя: {}".format(name) | Гибкость |
| % formatting | "Имя: %s" % name | Классический способ |

### **3. f-строки - современный стандарт**

|  |
| --- |
| *python name = "Анна" age = 16 average\_grade = 4.75  # Базовое использование print(f"Имя: {name}, Возраст: {age}")  # Вычисления внутри f-строк print(f"Через год будет: {age + 1} лет")  # Форматирование чисел print(f"Средний балл: {average\_grade:.2f}") # 4.75 print(f"Процент: {0.15:.1%}") # 15.0%* |

## **🎨 Визуальная схема: Процесс ввода-вывода**

|  |
| --- |
| text ПОЛЬЗОВАТЕЛЬ ВВОДИТ ДАННЫЕ  ↓ input() → ПРОГРАММА ОБРАБАТЫВАЕТ → ПРЕОБРАЗОВАНИЕ ТИПОВ  ↓ ФОРМАТИРОВАНИЕ ВЫВОДА → ПОЛЬЗОВАТЕЛЬ ВИДИТ РЕЗУЛЬТАТ |

## **💻 Практика**

### **Задание 4.1: Базовый диалог**

|  |
| --- |
| python # Простая программа-знакомство print("👋 Давайте познакомимся!") name = input("Как ваше имя? ") city = input("Из какого вы города? ") hobby = input("Какое у вас хобби? ")  print(f"\n🎉 Очень приятно, {name}!") print(f"📍 {city} - прекрасный город!") print(f"🎯 {hobby} - отличное занятие!") |

### **Задание 4.2: Калькулятор с вводом**

|  |
| --- |
| python # Интерактивный калькулятор print("🧮 ПРОСТОЙ КАЛЬКУЛЯТОР")  num1 = float(input("Введите первое число: ")) num2 = float(input("Введите второе число: ")) operation = input("Выберите операцию (+, -, \*, /): ")  if operation == "+":  result = num1 + num2  print(f"{num1} + {num2} = {result}") elif operation == "-":  result = num1 - num2  print(f"{num1} - {num2} = {result}") elif operation == "\*":  result = num1 \* num2  print(f"{num1} \* {num2} = {result}") elif operation == "/":  if num2 != 0:  result = num1 / num2  print(f"{num1} / {num2} = {result}")  else:  print("Ошибка: деление на ноль!") else:  print("Неизвестная операция!") |

### **Задание 4.3: Форматирование разных типов данных**

|  |
| --- |
| *python # Демонстрация форматирования product = input("Название товара: ") price = float(input("Цена товара: ")) quantity = int(input("Количество: ")) discount = float(input("Скидка (%): "))  total = price \* quantity final\_price = total \* (1 - discount/100)  print("\n📋 ДЕТАЛИ ЗАКАЗА:") print(f"Товар: {product:>15}") # Выравнивание вправо print(f"Цена за шт: {price:10.2f} руб.") # Ширина 10 символов print(f"Количество: {quantity:10d} шт.") # Целое число print(f"Скидка: {discount:10.1f}%") # Дробное число print("-" \* 30) print(f"ИТОГО: {final\_price:12.2f} руб.") # Итог с выравниванием* |

## **🚀 МИНИ-ПРОЕКТ**

"Умная анкета студента"

|  |
| --- |
| python # УМНАЯ АНКЕТА СТУДЕНТА print("🎓 УМНАЯ АНКЕТА СТУДЕНТА 🎓") print("=" \* 45)  # Сбор информации full\_name = input("Ваше полное имя: ") age = int(input("Ваш возраст: ")) course = int(input("Курс: ")) specialization = input("Специализация: ") grades = input("Оценки за последнюю сессию (через пробел): ")  # Обработка оценок grades\_list = [int(grade) for grade in grades.split()] average\_grade = sum(grades\_list) / len(grades\_list)  # Дополнительные вопросы programming\_exp = input("Опыт в программировании (мес): ") projects\_count = int(input("Количество завершенных проектов: "))  # Расчет "уровня студента" student\_level = (average\_grade \* 2) + (int(programming\_exp) \* 0.1) + (projects\_count \* 5)  # Вывод результатов print("\n" + "=" \* 45) print("📊 РЕЗУЛЬТАТЫ АНКЕТЫ:") print(f"👤 Студент: {full\_name}") print(f"🎂 Возраст: {age} лет") print(f"📚 Курс: {course}, {specialization}") print(f"⭐ Средний балл: {average\_grade:.2f}") print(f"💻 Опыт программирования: {programming\_exp} мес.") print(f"🚀 Завершенных проектов: {projects\_count}")  if student\_level > 50:  level\_status = "ПРОДВИНУТЫЙ" elif student\_level > 30:  level\_status = "СРЕДНИЙ" else:  level\_status = "НАЧИНАЮЩИЙ"  print(f"🎯 Ваш уровень: {level\_status} ({student\_level:.1f} баллов)") print("=" \* 45) |

Пример работы:

|  |
| --- |
| text 🎓 УМНАЯ АНКЕТА СТУДЕНТА 🎓 ============================================= Ваше полное имя: Иванов Алексей Петрович Ваш возраст: 18 Курс: 2 Специализация: Информатика Оценки за последнюю сессию (через пробел): 5 4 5 4 5 Опыт в программировании (мес): 12 Количество завершенных проектов: 3  ============================================= 📊 РЕЗУЛЬТАТЫ АНКЕТЫ: 👤 Студент: Иванов Алексей Петрович 🎂 Возраст: 18 лет 📚 Курс: 2, Информатика ⭐ Средний балл: 4.60 💻 Опыт программирования: 12 мес. 🚀 Завершенных проектов: 3 🎯 Ваш уровень: ПРОДВИНУТЫЙ (49.2 баллов) ============================================= |

## **🔥 ДОПОЛНИТЕЛЬНЫЙ ПРОЕКТ**

"Система бронирования столиков"

|  |
| --- |
| python # СИСТЕМА БРОНИРОВАНИЯ СТОЛИКОВ print("🍽️ СИСТЕМА БРОНИРОВАНИЯ СТОЛИКОВ 🍽️") print("=" \* 50)  # Ввод данных о бронировании client\_name = input("Ваше имя: ") phone = input("Контактный телефон: ") guests = int(input("Количество гостей: ")) date = input("Дата бронирования (дд.мм.гггг): ") time = input("Время бронирования (чч:мм): ") special\_requests = input("Особые пожелания: ")  # Расчет стоимости base\_price\_per\_person = 500 # Базовая стоимость if guests >= 6:  discount = 0.1 # 10% скидка на компании от 6 человек elif guests >= 4:  discount = 0.05 # 5% скидка на компании от 4 человек else:  discount = 0  total\_cost = guests \* base\_price\_per\_person \* (1 - discount) deposit = total\_cost \* 0.2 # 20% предоплата  # Генерация номера брони import random booking\_number = f"BK{random.randint(1000, 9999)}"  # Вывод подтверждения print("\n" + "=" \* 50) print("✅ БРОНИРОВАНИЕ ПОДТВЕРЖДЕНО!") print(f"📋 Номер брони: {booking\_number}") print(f"👤 Клиент: {client\_name}") print(f"📞 Телефон: {phone}") print(f"📅 Дата: {date} в {time}") print(f"👥 Количество гостей: {guests}") print(f"💰 Общая стоимость: {total\_cost:.2f} руб.") print(f"💳 Предоплата: {deposit:.2f} руб.")  if discount > 0:  print(f"🎁 Скидка: {discount\*100:.0f}% применена!")  if special\_requests:  print(f"💬 Особые пожелания: {special\_requests}")  print("=" \* 50) print("📧 Подтверждение отправлено на ваш телефон!") |

## **🤔 ВОПРОСЫ НА РАЗМЫШЛЕНИЕ**

1. Почему input() всегда возвращает строку, даже если ввели число?
2. В чем преимущество f-строк перед конкатенацией?
3. Что произойдет, если пользователь введет текст вместо числа в int(input())?

## **❌ ТАБЛИЦА ЧАСТЫХ ОШИБОК**

|  |  |  |
| --- | --- | --- |
| Ошибка | Пример | Решение |
| Забыли преобразовать тип | age = input() затем age + 1 | age = int(input()) |
| Неинформативный запрос | input() | input("Введите имя: ") |
| Ошибка ValueError | int("abc") | Обработка исключений (изучим позже) |
| Сложное форматирование | Много + и str() | Использовать f-строки |

## **📋 ШПАРГАЛКА**

|  |
| --- |
| *python # Ввод данных text = input("Введите текст: ") number\_int = int(input("Целое число: ")) number\_float = float(input("Дробное число: "))  # Форматирование вывода name = "Анна" age = 16  # f-строки (рекомендуется) print(f"Имя: {name}, Возраст: {age}")  # Конкатенация print("Имя: " + name + ", Возраст: " + str(age))  # Метод format() print("Имя: {}, Возраст: {}".format(name, age))  # Форматирование чисел price = 99.9876 print(f"Цена: {price:.2f}") # 99.99 print(f"Процент: {0.15:.1%}") # 15.0%* |

## **🏠 ДОМАШНЕЕ ЗАДАНИЕ**

### **Обязательная часть (для всех):**

1. "Персональный бюджет": Программа запрашивает доходы и расходы за месяц, рассчитывает остаток и дает рекомендации
2. "Конвертер единиц": Создайте конвертер для разных единиц измерения (метры/футы, кг/фунты, ℃/℉)

### **Для продвинутых (⭐):**

"Система учета книг": Программа позволяет добавлять книги (название, автор, год, жанр), рассчитывает статистику (самый старый автор, самый популярный жанр) и выводит красивые отчеты.

## **🌍 PYTHON В РЕАЛЬНОЙ ЖИЗНИ**

Где используется ввод данных и форматирование:

* 🏦 Банковские приложения — формы ввода, выписки
* 🛒 Интернет-магазины — корзина покупок, оформление заказов
* 📊 Аналитические системы — ввод параметров, генерация отчетов
* 🎮 Игры — создание персонажа, настройки
* 📱 Мобильные приложения — регистрация, профили пользователей

Интересный факт: f-строки появились в Python 3.6 и сразу стали самым популярным способом форматирования благодаря скорости и читаемости!

## **✅ ТЕСТ ДЛЯ САМОПРОВЕРКИ**

1. Как преобразовать ввод пользователя в дробное число?
2. Что выведет код: print(f"Результат: {5 + 3 \* 2}")?
3. Как выровнять текст по правому краю в f-строке?

**🎉 Великолепно! Теперь ваши программы могут общаться с пользователем!**

### 

### **УРОК 5: STRING PROCESSING — МАГИЯ РАБОТЫ С ТЕКСТОМ**

**🎯 Цели урока:  
· Освоить основные методы работы со строками  
· Научиться форматировать и преобразовывать текст  
· Понять работу со срезами (slicing)  
· Создать текстовые процессоры и генераторы**

**📚 Теория**

**1. Основные методы строк  
В Python строки — это не просто текст. Это объекты с множеством полезных методов, которые позволяют легко изменять, анализировать и форматировать текст.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Метод** | **Пример** | **Результат** | **Описание** |
| .upper() | "привет".upper() | "ПРИВЕТ" | Переводит в верхний регистр |
| .lower() | "ПРИВЕТ".lower() | "привет" | Переводит в нижний регистр |
| .title() | "алексей иванов".title() | "Алексей Иванов" | Каждое слово с заглавной буквы |
| .strip() | " Душанбе ".strip() | "Душанбе" | Удаляет пробелы по краям |
| .replace() | "Python".replace("n", "m") | "Pythom" | Заменяет один символ на другой |
| .split() | "яблоко,банан,апельсин".split(",") | ["яблоко", "банан", "апельсин"] | Разбивает строку на список по разделителю |
| .find() | "Программирование".find("м") | 4 | Возвращает позицию первого вхождения |
| len() | len("Python") | 6 | Возвращает длину строки |

**💡 Почему это важно?  
Представьте, что вы создаёте электронный дневник для школы в Худжанде. Ученик вводит имя как " алишер мирзоев " — с пробелами и строчными буквами.  
Без .strip() и .title() ваша система покажет:  
Ученик: алишер мирзоев — это выглядит неаккуратно.  
С методами — вы получите:  
Ученик: Алишер Мирзоев — как в официальном документе!**

**2. Срезы (slicing) — мощный инструмент  
Срезы позволяют извлекать части строки, как кусочек пирога!**

|  |
| --- |
| **text = "Программирование" *# Базовый синтаксис: [начало:конец:шаг]* print(text[0:5]) *# "Прогр" -- первые 5 символов* print(text[5:]) *# "аммирование" -- с 5-го до конца* print(text[:7]) *# "Програм" -- с начала до 7-го* print(text[::2]) *# "Пормиоаи" -- каждый 2-й символ* print(text[-5:]) *# "вание" -- последние 5 символов* print(text[::-1]) *# "ениваморгамаргоП" -- в обратном порядке*** |

**💬 Практический пример — извлечение домена из email:**

|  |
| --- |
| **email = "student@school.tj" domain = email[email.find("@")+1:] *# Начинаем после @* print(domain) *# Выведет: school.tj*** |

**3. f-строки с форматированием — красивый вывод  
Вы уже знаете f-строки. Теперь добавим выравнивание и форматирование!**

**python**

|  |
| --- |
| **name = "анна" age = 16 price = 99.99  *# Выравнивание текста* print(f"Имя: {name:>10}") *# Выравнивание вправо: " анна"* print(f"Возраст: {age:<10}") *# Выравнивание влево: "16 "* print(f"Цена: {price:^10.2f}") *# По центру: " 99.99 "*  *# Методы внутри f-строк!* print(f"Имя: {name.upper()}") *# Выведет: Имя: АННА* print(f"Имя: {name.title()}") *# Выведет: Имя: Анна*** |

**🎨 Визуальная схема: Срезы строк**

|  |
| --- |
| **Строка: "ПРОГРАММИРОВАНИЕ" Индексы: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14  П Р О Г Р А М М И Р О В А Н И Е  Срезы: [0:5] → "ПРОГР" [5:10] → "АММИР"  [::2] → "ПОРМИОАИ" (каждый 2-й) [-5:] → "АНИЕ" (последние 5)** |

**💻 Практика**

**Задание 5.1: Базовые методы строк**

|  |
| --- |
| ***# Исследуем методы работы со строками* text = " Python - лучший язык программирования! "  print("Исходный текст:", text) print("В верхнем регистре:", text.upper()) print("В нижнем регистре:", text.lower()) print("Без пробелов:", text.strip()) print("Замена 'лучший' на 'отличный':", text.replace("лучший", "отличный")) print("Длина строки:", len(text)) print("Разделение по пробелам:", text.split())** |

**📌 Ваша задача:  
Введите свой город или название школы вместо "Python - лучший язык..." и примените все методы.  
Как изменится результат, если убрать .strip()?**

**Задание 5.2: Работа со срезами**

**python**

|  |
| --- |
| ***# Практика срезами* phrase = "Программирование это искусство"  print("Исходная фраза:", phrase) print("Первые 10 символов:", phrase[:10]) print("Последние 8 символов:", phrase[-8:]) print("Каждый 3-й символ:", phrase[::3]) print("С 5 по 15 символ:", phrase[5:15]) print("В обратном порядке:", phrase[::-1])  *# Полезный пример -- извлечение домена из email* email = "user@school.tj" domain = email[email.find("@")+1:] print("Домен email:", domain)** |

**🔍 Эксперимент!  
Что выведет phrase[10:5]? Почему?  
*Подсказка: срез не работает "назад", если начало > конец.***

**Задание 5.3: Форматирование текста**

|  |
| --- |
| ***# Создаем красивые отчеты* product\_name = "смартфон" brand = "samsung" price = 29999.99 rating = 4.75623  *# Форматируем вывод* print("🏪 МАГАЗИН ЭЛЕКТРОНИКИ") print("=" \* 40) print(f"Товар: {product\_name.title():<15}") *# С заглавной, выравнивание влево* print(f"Бренд: {brand.upper():<15}") *# В верхнем регистре* print(f"Цена: {price:>15.2f} руб.") *# Выравнивание вправо, 2 знака* print(f"Рейтинг: {rating:>14.1f}/5") *# Округление до 1 знака* print("=" \* 40)  *# Расчет скидки* discount\_price = price \* 0.85 *# 15% скидка* print(f"Цена со скидкой: {discount\_price:>10.2f} руб.")** |

**💡 Реальный контекст:  
Это как чек в магазине "Таджик-Сити" в Душанбе — все выровнено, с копейками, с заглавными буквами.  
Так выглядят профессиональные системы!**

**🚀 МИНИ-ПРОЕКТ**

### ГЕНЕРАТОР БЕЗОПАСНЫХ ПАРОЛЕЙ — ДЛЯ ШКОЛЬНОГО ПОРТАЛА

|  |
| --- |
| ***# ГЕНЕРАТОР БЕЗОПАСНЫХ ПАРОЛЕЙ* print("🔐 ГЕНЕРАТОР БЕЗОПАСНЫХ ПАРОЛЕЙ 🔐") print("=" \* 45)  *# Ввод данных пользователя* full\_name = input("Введите ваше полное имя: ") birth\_year = input("Введите год рождения: ") favorite\_word = input("Введите любимое слово: ")  *# Генерация паролей на основе введенных данных*  *# Пароль 1: Комбинация имени и года* password1 = full\_name[:3].lower() + birth\_year[-2:] + favorite\_word[-3:].upper()  *# Пароль 2: Перемешанная версия* password2 = full\_name[1:4].upper() + birth\_year + favorite\_word[:2].lower()  *# Пароль 3: Специальный алгоритм* mixed\_name = full\_name[::2] + full\_name[1::2] *# Четные + нечетные символы* password3 = mixed\_name[:4].lower() + birth\_year[-1] + favorite\_word.upper()[:3]  *# Анализ паролей* def analyze\_password(password):  length = len(password)  has\_upper = any(char.isupper() for char in password)  has\_lower = any(char.islower() for char in password)  has\_digit = any(char.isdigit() for char in password)    strength = "СЛАБЫЙ"  if length >= 8 and has\_upper and has\_lower and has\_digit:  strength = "СИЛЬНЫЙ"  elif length >= 6 and (has\_upper or has\_lower):  strength = "СРЕДНИЙ"  return length, strength  *# Вывод результатов* print("\n" + "=" \* 45) print("🎲 СГЕНЕРИРОВАННЫЕ ПАРОЛИ:") print("-" \* 45)  passwords = [password1, password2, password3] for i, pwd in enumerate(passwords, 1):  length, strength = analyze\_password(pwd)  print(f"Пароль {i}: {pwd}")  print(f" Длина: {length} символов")  print(f" Надежность: {strength}")  print()  print("💡 СОВЕТЫ ПО БЕЗОПАСНОСТИ:") print("• Используйте пароли длиной от 8 символов") print("• Комбинируйте буквы (заглавные и строчные), цифры и символы") print("• Не используйте личную информацию в паролях") print("=" \* 45)** |

**📌 Пример работы:**

|  |
| --- |
| **Введите ваше полное имя: Алексей Иванов Введите год рождения: 2000 Введите любимое слово: программирование  🎲 СГЕНЕРИРОВАННЫЕ ПАРОЛИ: --------------------------------------------- Пароль 1: але00НИЕ  Длина: 8 символов  Надежность: СРЕДНИЙ  Пароль 2: ЛЕК2000пр  Длина: 9 символов  Надежность: СРЕДНИЙ  Пароль 3: АееI2000ПРО  Длина: 11 символов  Надежность: СИЛЬНЫЙ** |

**💡 Почему это важно?  
В электронном дневнике школы или портале образовательного центра ученики должны создавать надёжные пароли.  
Эта программа — реальный инструмент, который помогает ученикам защитить свои данные!**

**🔥 ДОПОЛНИТЕЛЬНЫЙ ПРОЕКТ**

### ТЕКСТОВЫЙ АНАЛИЗАТОР И РЕДАКТОР — ДЛЯ УЧИТЕЛЕЙ

|  |
| --- |
| ***# ТЕКСТОВЫЙ АНАЛИЗАТОР И РЕДАКТОР* print("📊 ТЕКСТОВЫЙ АНАЛИЗАТОР И РЕДАКТОР 📊") print("=" \* 50)  *# Ввод текста для анализа* text = input("Введите текст для анализа:\n").strip()  if not text:  print("❌ Вы не ввели текст!") else:  print("\n" + "=" \* 50)  print("📈 РЕЗУЛЬТАТЫ АНАЛИЗА:")  print("-" \* 50)    *# Базовый анализ текста*  total\_chars = len(text)  total\_words = len(text.split())  total\_sentences = text.count('.') + text.count('!') + text.count('?')    *# Поиск самого длинного слова*  words = text.split()  longest\_word = max(words, key=len) if words else "нет"    *# Анализ регистра*  upper\_count = sum(1 for char in text if char.isupper())  lower\_count = sum(1 for char in text if char.islower())  digit\_count = sum(1 for char in text if char.isdigit())    *# Преобразования текста*  cleaned\_text = text.strip()  upper\_text = text.upper()  title\_text = text.title()  reversed\_text = text[::-1]    *# Вывод анализа*  print(f"Общее количество символов: {total\_chars}")  print(f"Количество слов: {total\_words}")  print(f"Количество предложений: {total\_sentences}")  print(f"Самое длинное слово: '{longest\_word}'")  print(f"Заглавных букв: {upper\_count}")  print(f"Строчных букв: {lower\_count}")  print(f"Цифр: {digit\_count}")    *# Вывод преобразований*  print("\n🔄 ПРЕОБРАЗОВАНИЯ ТЕКСТА:")  print("-" \* 50)  print(f"Очищенный текст: '{cleaned\_text}'")  print(f"В верхнем регистре: '{upper\_text}'")  print(f"В формате заголовка: '{title\_text}'")  print(f"В обратном порядке: '{reversed\_text}'")    *# Дополнительные возможности*  search\_word = input("\n🔍 Введите слово для поиска в тексте: ")  if search\_word:  occurrences = text.lower().count(search\_word.lower())  print(f"Слово '{search\_word}' встречается {occurrences} раз(а)")    replace\_word = input("\n✏️ Введите слово для замены: ")  if replace\_word:  new\_word = input("Введите новое слово: ")  updated\_text = text.replace(replace\_word, new\_word)  print(f"Обновленный текст: '{updated\_text}'")  print("=" \* 50)** |

**💡 Реальная польза для учителя:  
Ученик присылает сочинение с ошибками:  
"я люблю пайтон. он лекго и интересно"  
Эта программа покажет:**

* **Сколько слов — 6**
* **Сколько заглавных букв — 1 (в начале)**
* **И предложит:  
  "Я люблю Python. Он легко и интересно"**

**Это первый шаг к автоматической проверке!**

**🤔 ВОПРОСЫ НА РАЗМЫШЛЕНИЕ**

1. **В чем разница между text.upper() и text.title()?  
   *Подсказка:* upper() делает ВСЁ заглавными. title() — только первые буквы слов.**
2. **Как получить последние 3 символа строки разными способами?  
   *Подсказка:* text[-3:] или text[len(text)-3:]**
3. **Почему text.strip() важен при обработке пользовательского ввода?  
   *Подсказка:* Пользователь может ввести " Душанбе " — с пробелами. Без .strip() ваша программа не найдёт "Душанбе" в списке городов!**

**❌ ТАБЛИЦА ЧАСТЫХ ОШИБОК**

|  |  |  |
| --- | --- | --- |
| **ОШИБКА** | **ПРИМЕР** | **РЕШЕНИЕ** |
| **Индекс вне диапазона** | **text[100]** | **Проверить длину:if len(text) > 100** |
| **Мутация строки** | **text[0] = "A"** | **Создать новую:"A" + text[1:]** |
| **Неправильный срез** | **text[5:1]** | **Указать правильно:text[1:5]** |
| **Забыли скобки** | **text.upper** | **Вызвать метод:text.upper()** |

**📋 ШПАРГАЛКА**

|  |
| --- |
| ***# Основные методы строк* text.upper() *# В верхний регистр* text.lower() *# В нижний регистр* text.title() *# Каждое слово с заглавной* text.strip() *# Удаление пробелов по краям* text.replace("a", "b") *# Замена символов* text.split() *# Разделение на список* len(text) *# Длина строки*  *# Срезы (slicing)* text[start:end:step] *# Базовый синтаксис* text[:5] *# Первые 5 символов* text[5:] *# Все после 5-го символа* text[-3:] *# Последние 3 символа* text[::2] *# Каждый 2-й символ* text[::-1] *# Обратный порядок*  *# f-строки с форматированием* f"{text:>10}" *# Выравнивание вправо* f"{text:<10}" *# Выравнивание влево* f"{text:^10}" *# Выравнивание по центру* f"{number:.2f}" *# 2 знака после запятой*** |

**🏠 ДОМАШНЕЕ ЗАДАНИЕ**

**Обязательная часть (для всех):**

1. **"Корректор текста"  
   Напишите программу, которая исправляет распространённые ошибки:**
   * **Убирает двойные пробелы**
   * **Делает первую букву предложения заглавной**
   * **Удаляет лишние точки в конце  
     *Пример:*"я люблю пайтон... " → "Я люблю пайтон."**
2. **"Генератор email-адресов"  
   Создайте программу, которая на основе ФИО и названия школы создаёт корпоративный email в формате:  
   имя.фамилия@school.tj  
   *Пример:*Алишер Мирзоев → alisher.mirzoev@school.tj  
   *Подсказка:* используйте .lower() и .replace()!**

**Для продвинутых (⭐):**

**"Шифратор/дешифратор текста"**  
Создайте программу, которая шифрует текст по алгоритму Цезаря (сдвиг букв на N позиций).  
Добавьте возможность расшифровать текст.  
***Пример:***"Привет" при сдвиге +1 → "Срйгжу"  
*Подсказка:* используйте ord() и chr() — функции для работы с кодами символов.

**🌍 PYTHON В РЕАЛЬНОЙ ЖИЗНИ — ТАДЖИКСТАН**

**Где используется обработка строк:  
· 🔍 Поисковые системы — обработка запросов, стемминг (поиск "учебник" и "учебники")  
· 📧 Email-клиенты — парсинг адресов, тем писем  
· 📊 Анализ данных — очистка и подготовка текстовых данных (например, из анкет учеников)  
· 🎮 Игры — обработка команд игрока, диалоги с персонажами  
· 🤖 Чат-боты — анализ и генерация ответов (например, в Telegram-боте школы)**

**💡 Интересный факт:  
Python используется для обработки текстов в Wikipedia!  
Миллионы статей обрабатываются с помощью библиотек для работы с текстом — точно так же, как вы делаете это сейчас.**

**✅ ТЕСТ ДЛЯ САМОПРОВЕРКИ**

1. **Как получить длину строки?  
   *Ответ:* len(text)**
2. **Что вернет "hello".replace("l", "p")?  
   *Ответ:* "heppo"**
3. **Как получить каждый 3-й символ строки "Python"?  
   *Ответ:* text[::3] → "Ph"**

**Вы уже не просто ученик. Вы — создатель цифровых инструментов.**