**🟦 Занятие 2 (90 минут) – Типы данных, операторы и выражения**

**🎯 Цели занятия**

* Закрепить знания о типах данных: числа (int, float), строки (str), логические значения (bool).
* Научиться использовать арифметические, логические и сравнительные операторы.
* Попрактиковаться в создании более сложного калькулятора и выражений.

**🔹 Структура занятия**

**0–10 мин — Разогрев и повторение**

* Вопросы группе:
  + Какие типы данных мы знаем в Python?
  + Что такое переменная?
  + Какие операторы мы использовали в калькуляторе?
* Короткая демонстрация: вывод разных типов данных через print().

**10–30 мин — Типы данных**

* **Числа:** целые (int), дробные (float), пример:

x = 10 # int

y = 3.14 # float

print(x + y) # сложение int + float

* **Строки:** создание и операции:

s = "Python"

print(s.upper())

print(s + " rocks!")

* **Логические значения:** True, False, пример:

a = 5

b = 10

print(a < b) # True

print(a == b) # False

💡 Мини-упражнения:

1. Создать переменные a=7, b=2.5, вывести их сумму, разность, произведение и деление.
2. Создать строку с вашим именем и приветствие, вывести через print.
3. Сравнить два числа и вывести результат (True/False).

**30–60 мин — Операторы и выражения**

* **Арифметические:** + - \* / // % \*\* (с примерами).
* **Сравнения:** == != > < >= <=.
* **Логические:** and, or, not.
* Демонстрация комбинированных выражений:

x = 5

y = 10

print(x < y and y > 0) # True

print(not(x > y)) # True

💡 Мини-упражнения:

1. Написать выражение, которое проверяет, что число x > 0 и < 100.
2. Проверить, равны ли две строки.
3. Использовать логические операторы and, or, not в простых выражениях.

**60–85 мин — Практика: расширенный калькулятор**

* Задача: пользователь вводит два числа и операцию (+ - \* / % \*\*).
* Обработать все операции, добавить проверку деления на ноль.
* Пример (демо):

a = float(input("Введите первое число: "))

b = float(input("Введите второе число: "))

op = input("Введите операцию (+, -, \*, /, %, \*\*): ")

if op == "+":

result = a + b

elif op == "-":

result = a - b

elif op == "\*":

result = a \* b

elif op == "/":

result = a / b if b != 0 else "Ошибка: деление на ноль"

elif op == "%":

result = a % b if b != 0 else "Ошибка: деление на ноль"

elif op == "\*\*":

result = a \*\* b

else:

result = "Неизвестная операция"

print(f"Результат: {result}")

💡 Практика для группы:

* Новички: реализовать базовые операции + - \* /.
* Продвинутые: добавить % и \*\*, проверку деления на ноль, красиво оформить вывод с f-строками.

**85–90 мин — Итог и домашнее задание**

* Обсудить, что нового узнали.
* Вопросы, разбор сложных выражений.
* Анонс следующего занятия: условные конструкции (if) и циклы.

## Задача 1. Проверка диапазона числа

**ТЗ:**  
Пользователь вводит число.  
Программа должна проверить, попадает ли число в диапазон от 1 до 100 (включительно).  
Вывести True, если попадает, и False, если нет.

## Задача 2. Простое условное вычисление

**ТЗ:**  
Пользователь вводит два числа.  
Если первое число больше второго — вывести их сумму,  
иначе — вывести их произведение.

## Задача 3. Проверка делимости

**ТЗ:**  
Пользователь вводит число.  
Программа должна проверить, делится ли оно на 3 и на 5 одновременно.  
Вывести сообщение "Делится на 3 и 5" или "Не делится на 3 и 5".

## Задача 4. Проверка текста

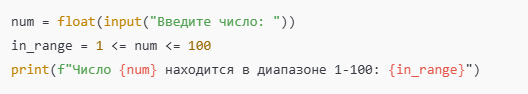
**ТЗ:**  
Пользователь вводит строку.  
Программа должна проверить:

* начинается ли строка с заглавной буквы,
* заканчивается ли точкой.  
  Вывести True, если оба условия выполняются, иначе False.

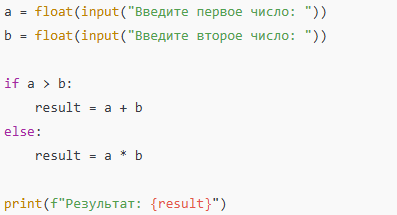
## Задача 5. Мини-калькулятор с отрицательными числами

**ТЗ:**  
Пользователь вводит два числа и операцию (+, -, \*, /).  
Программа должна правильно обрабатывать отрицательные числа и деление на ноль.  
Вывести результат с пояснением.

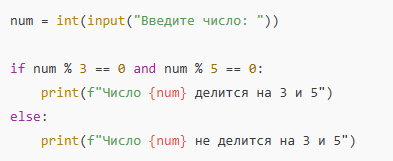
1



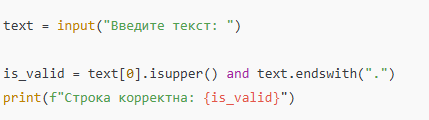
2



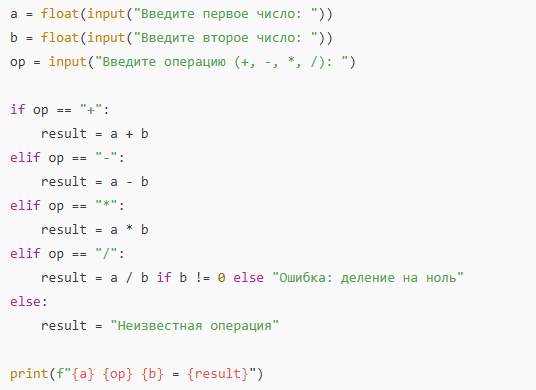
3



4



5



**1️⃣ Повторяем, что такое тип данных**

* **Тип данных** — это «вид» информации, с которой работает программа.
* Основные типы данных в Python:
  1. **Числа** (int – целые, float – дробные)
  2. age = 10 # int
  3. height = 1.45 # float
  4. **Строки** (str) – текст
  5. name = "Анна"
  6. **Логические значения** (bool) – True или False
  7. is\_raining = False

💡 **Аналогия:**

* Число – как количество яблок 🍎
* Строка – как надпись на коробке 📦
* Логическое значение – как переключатель: включено/выключено ⚡

**2️⃣ Арифметические операторы**

* + сложение, - вычитание, \* умножение, / деление
* // целочисленное деление, % остаток от деления, \*\* возведение в степень

Примеры:

print(2 + 3) # 5

print(10 / 4) # 2.5

print(10 // 4) # 2

print(2 \*\* 3) # 8

**3️⃣ Операторы сравнения**

* == равно, != не равно
* > больше, < меньше
* >= больше или равно, <= меньше или равно

Примеры:

print(5 > 3) # True

print(5 == 5) # True

print(2 != 2) # False

**4️⃣ Логические операторы**

* and – «и», or – «или», not – «не»  
  Примеры:

print(True and False) # False

print(True or False) # True

print(not True) # False

💡 **Аналогия:**

* and – все условия должны быть верны (как две лампочки, обе должны гореть).
* or – хотя бы одно верно (одна лампочка горит — свет есть).
* not – противоположное значение (включено → выключено).

**5️⃣ Простое упражнение для закрепления**

1. Создать две переменные с числами, вывести их сумму и разность.
2. Создать строку с именем и поприветствовать её через print.
3. Проверить, больше ли одно число другого, вывести True/False.