Лабораторная работа №1

Задание 1

Для ввода информации в переменные используется **input()**, для вывода - **print()**.

Используя эти команды создайте **5 переменных** и введите с клавиатуры в них значения различных типов (целый, вещественный, логический, строка).

Code:



Result:  


При помощи **type()** выведите типы этих переменных на экран.

Code:



Result:  


Создайте **ещё 5 переменных**, но при их записи используйте команды для преобразования данных в нужный тип. Выведите их на экран вместе с их типом.

Code:



Result:  


**Задание 2.**

Создайте три числовые переменные, указав их через запятую: **a, b, c** = 10, 2, 7.

Создайте ещё 5 переменных, которые используют переменные **a, b, c** для вычисления суммы, разности, произведения, остатка от деления, деления без остатка.

Поменяйте значения переменных **a, b одной строкой кода.**

Выведите все переменные используя f-строки.

Code:  


Result:



Найдите результат вычислений данных примеров:

* 
* 
* x % 2 + c // 4 - v / 10 + 18b

не создавайте переменные для значений x, c, v, b, а пропишите **input()** прямо в формуле. используйте модуль **Math** для сложных математических вычислений.

Выведете результаты всех 4 формул в одну строчку, через “\*\*-\*\*” и в конце “!!!!” (для этого используйте **sep**, **end** в **print** )

Code:  


Result:



Форматированный вывод.

Форматированный ввод-вывод в Python обеспечивает более удобное и наглядное представление данных при выполнении программ. В Python для форматирования используется метод format() для строк и специальные символы для форматирования чисел.

Используя форматированный вывод сделайте следующее:

Напишите программу, которая запрашивает у пользователя его имя и возраст, а затем выводит приветствие в формате "Привет, <имя>! Тебе <возраст> лет."

Code:



Result:



Напишите программу, которая запрашивает у пользователя радиус окружности и выводит её площадь. Формат вывода: "Площадь окружности радиусом <радиус> равна <площадь>."

Code:



Result:



Напишите программу, которая запрашивает у пользователя два числа (a, b) и выводит результаты деления a на b и b на a . " Округлите первое вычисление до 3 знаков после запятой, а второе - до 2.

Code:



Result:



**Задание 5.**

Подключите модуль **time** к файлу.

Попробуйте считать сегодняшнюю дату и время используя **time() и localtime().**

Code:



Result:



Используя форматированный вывод выведите на экран полученные данные в виде:

Текущая дата и время: 2024-01-23 15:30:00

Дата: 23/01/2024

Время: 15:30:00

Текущая дата и время: 23.01.2024 15:30:00

Дата: 23 января 2024

Время: 15:30

Code:



Result:



Лабораторная работа №2 (2ч). Условные конструкции (if и match). Циклы.

**Задание 1 (используем if).**

Введите в переменные два числа.

Если первое число больше второго, то вывести yes.

Если второе число больше первого поменять значения этих переменных и вывести их на экран.

Если числа равны продублировать их 5 раз в строку (без использования цикла).

Code:  


Result:  






**Задание 2 (используем match).**

Написать мини-калькулятор.

## Необходимо спросить какую операцию необходимо сделать : /, \*,-,+,//, \*\*, sin, cos, tg, корень. Далее в зависимости от выбранной операции попросить ввести одно или два числа. После этого посчитать результат и вывести его.

Code:







Result:



**Задание 3 (используем while).**

Напишите программу, которая просит ввести с клавиатуры число X с клавиатуры, исходя из введенного число выведите последовательность следующую последовательность (от 0 до 10): 0, 1\*X, 2\*X, …, 10\*X.

Code:



Result:



**Задание 4 (используем for ).**

Используя цикл for выведите на экран все числа (для каждого - отдельный цикл: всего 7):

* от 1 до 15
* от 5 до 20
* от 20 до 2
* от 2 до 16 с шагом 2
* от 4 до 44 с шагом 4
* от 5 до 25 с шагом 5
* от 80 до 0 с шагом 10

**Приведите примеры использования:**

* for \_ in range( V ): pass
* for z, x in range( V ), range( B ): pass
* for index, fruit in enumerate(fruits): pass
* for name, age in zip(names, ages): pass
* for c else

Code:



Result:



Code:



Result:  


Code:



Result:



Code:



Result:



Code:



Result:



Code:



Result:



Code:



Result:



Code:



Result:



**Задание 5.**

Пользователь вводит ненулевые целые числа до тех пор, пока не введет ноль. Найдите количество четных чисел, которые он ввел.

Code:



Result:



## Задание 7.

При помощи вложенных циклов постройте фигуры:

|  |  |
| --- | --- |
| №1  + + + + + + + + +  + +  + +  + +  + +  + + + + + + + + + | №2  + + + + + + + + +  + + +  + + + + + + + + +  + + +  + + + + + + + + +  + + +  + + + + + + + + + |

Code:



Result:

