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 6.

## Используйте модуль random. Компьютер загадывает число от А до В (вводятся с клавиатуры).

## Напишите угадайку этого числа.

## Программа должна просит игрока угадать число. После того как игрок ввел число, программа должна выводит сообщение «Твоё число больше, чем нужно!» или «Твоё число меньше, чем нужно!». И заново просить угадать число. Когда игрок угадал вывести слово «Победа!»

Code:



Result:



## Задание 7.

При помощи вложенных циклов постройте фигуры:

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

Code:



Result:



Code:



Result:



## Задание 8.

## Напишите программу, которая по номеру дня недели — целому числу от 1 до 7 — выдает в качестве результата количество занятий в вашей группе в соответствующий день.

Code:



Result:



## Задание 9.

## В зависимости от стажа работы педагогам введена надбавка в размере: для работающих от 5 до 10 лет — 10%; для работающих от 10 до 15 лет — 15%; для работающих свыше 15 лет — 20%.

## Составьте программу, которая по заданному стажу работы и размеру оклада определит размер заработной платы.

Code:



Result:



## Задание 10.

## Составить программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 999), обозначающего денежную величину, дописывает слово «рубль» в правильной форме.

## Например, 5 рублей, 21 рубль, 173 рубля.

Code:



Result:







## + Лабораторная работа №3 (4ч). Методы обработки строк и списков. Словари и множества. Регулярные выражения.

**Задание 1 (строки).**

1. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые начинаются и заканчиваются одной и той же буквой.

Code:



Result:



Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат хотя бы одну букву «А».

Code:



Result:



Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого короткого слова.

Code:



Result:



Дана строка-предложение. Зашифровать ее, поместив вначале все символы, расположенные на четных позициях строки, а затем, в обратном порядке, все символы,расположенные на нечетных позициях (например, строка «Программа» превратит-ся в «ргамамроП»).

Code:



Result:



Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним символом «.» (точка). В конце строки точку не ставить.

Result:



Result:



Дана строка, содержащая полное имя файла, то есть имя диска, список каталогов(путь), собственно имя и расширение (например, d:\ivanov\primer\prog.py). Выделить из этой строки имя файла (без расширения).

Code:



Result:



Получите строку из заданной, в которой все вхождения ее первого символа были изменены на '$', кроме самого первого символа.

Code:



Result:



Добавьте 'ing' в конце заданной строки (длина должна быть не менее 3). Если данная строка уже заканчивается на 'ing', добавьте вместо этого 'ly'. Если длина заданной строки меньше 3, оставьте ее без изменений.

Code:



Result:







Дана строка-предложение. Зашифровать ее, выполнив циклическую замену каждой буквы на следующую за ней в алфавите и сохранив при этом регистр букв («A» перейдет в «B», «a» — в «b», «B» — в «D», «z» — в «a» и т. д.). Знаки препинания и пробелы не изменять.

Code:



Result:



Преобразуйте строку-предложение так, чтобы каждое слово начиналось с большой буквы, а остальные буквы были в нижнем регистре.

Удалите все дублирующиеся символы из заданной строки.

Code:





Result:



**Задание 2 (списки).**

Управление списком покупок.

Выполнить ряд задач, используя различные методы, функции и циклы.

1. Создайте пустой список **shopping\_list**.
2. Запросите у пользователя ввод элементов списка покупок до тех пор, пока он не введёт слово "готово".
3. Добавьте каждый введенный элемент в список **shopping\_list**, за исключением слова "готово".
4. Выведите список **shopping\_list** на экран с помощью функции print().
5. Проверьте, пуст ли список. Если список пуст, выведите сообщение "Список покупок пуст!".
6. Предложите пользователю ввести индекс элемента списка, который он хочет обновить.
7. Запросите новое значение для этого элемента.
8. Обновите элемент списка с помощью присваивания.
9. Предложите пользователю ввести элемент списка, который он хочет удалить.
10. Если элемент присутствует в списке, удалите его с помощью метода **remove**(). Если элемент отсутствует, выведите сообщение об ошибке.
11. Отсортируйте список **shopping\_list** в алфавитном порядке с помощью функции **sorted**().
12. Выведите отсортированный список на экран при помощи цикла.
13. Подсчитайте количество элементов в списке **shopping\_list** с помощью функции **len**().

Code:



Result:

