

Программирование на языке Python. Базовый уровень

Модуль 1. Введение в программирование

Тема 1.3. Работа с целыми числами (занятие 1)



Целые числа

Целые числа в Python представлены типом данных `int` (сокращение `int` происходит от слова `integer`). Для определения целого числа типа `int` используется последовательность цифр от 0 до 9.

`int()` и `str()` используются для перехода от одного типа данных к другому.

```
add_services = input('Стоимость дополнительных услуг:')  
add_services = int(add_services)  
total = 2500 + add_services  
print('Общая стоимость:', total)
```



Целочисленные операторы

Оператор	Описание	Примеры
+	Сложение - Суммирует значения слева и справа от оператора	<p>15 + 5 в результате будет 20</p> <p>20 + -3 в результате будет 17</p> <p>13.4 + 7 в результате будет 20.4</p>
-	Вычитание - Вычитает правый операнд из левого	<p>15 - 5 в результате будет 10</p> <p>20 - -3 в результате будет 23</p> <p>13.4 - 7 в результате будет 6.4</p>
*	Умножение - Перемножает операнды	<p>5 * 5 в результате будет 25</p> <p>7 * 3.2 в результате будет 22.4</p> <p>-3 * 12 в результате будет -36</p>
/	Деление - Делит левый операнд на правый	<p>15 / 5 в результате будет 3</p> <p>5 / 2 в результате будет 2 (В Python 2.x версии при делении двух целых чисел результат будет целое число)</p> <p>5.0 / 2 в результате будет 2.5 (Чтобы получить "правильный" результат хотя бы один операнд должен быть float)</p>

Целочисленные операторы

%

Деление по модулю - Делит левый операнд на правый и возвращает остаток.

6 % 2 в результате будет 0
7 % 2 в результате будет 1
13.2 % 5 в результате 3.2

**

Возведение в степень - возводит левый операнд в степень правого

5 ** 2 в результате будет 25
2 ** 3 в результате будет 8
-3 ** 2 в результате будет -9

//

Целочисленное деление - Деление в котором возвращается только целая часть результата. Часть после запятой отбрасывается.

12 // 5 в результате будет 2
4 // 3 в результате будет 1
25 // 6 в результате будет 4

Приоритет операторов в Python

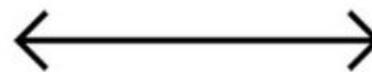
Оператор	Описание
**	Возведение в степень
~ + -	Комплементарный оператор
* / % //	Умножение, деление, деление по модулю, целочисленное деление.
+ -	Сложение и вычитание.

При обычном делении (/) получается число не являющееся целым. Деление на ноль приводит к ошибке.



Длинная арифметика

Отличительной особенностью языка Python является неограниченность целочисленного типа данных. По факту, размер числа зависит только от наличия свободной памяти на компьютере. Это отличает Python от таких языков как C++, C, C#, Java где переменные целого типа данных имеют ограничение.



$$10 * 5 + 100 / 10 - 5 + 7 \% 2$$

$$50 + 100 / 10 - 5 + 7 \% 2$$

$$50 + 10 - 5 + 7 \% 2$$

$$50 + 10 - 5 + 1$$

$$60 - 5 + 1$$

$$55 + 1$$

$$56$$

Обработка цифр числа

Запрограммируйте калькулятор веса багажа. Обычно багаж бывает трёх видов:

- основной багаж (большой чемодан/сумка),
- ручная кладь (маленькая сумочка, пакет),
- доп. ручная кладь (клетка с животным, букет цветов).

Напиши программу, которая будет запрашивать вес каждого элемента по отдельности и затем находить общий вес багажа.

```
a=int(input('Вес чемодана'))  
b=int(input('Вес ручной клади'))  
c=int(input('Вес доп. предметов ручной клади'))  
roop = a+b+c  
print('Общий вес багажа:',roop)
```

Обработка цифр числа

Напишите программу, в которой рассчитывается сумма и произведение цифр положительного трёхзначного числа.

```
num = int(input())  
digit3 = num % 10  
digit2 = (num // 10) % 10  
digit1 = num // 100  
print('Сумма цифр =', digit1+digit2+digit3)  
print('Произведение цифр =', digit1*digit2*digit3)
```

Обработка цифр числа

Напишите программу для нахождения цифр четырёхзначного числа.

Дано число: 3281

Цифра в позиции тысяч равна 3

Цифра в позиции сотен равна 2

Цифра в позиции десятков равна

8

Цифра в позиции единиц равна 1

```
a = int(input())
```

```
print('Цифра в позиции тысяч равна', a//1000)
```

```
print('Цифра в позиции сотен равна', a//100%10)
```

```
print('Цифра в позиции десятков равна', a//10%10)
```

```
print('Цифра в позиции единиц равна', a%10)
```