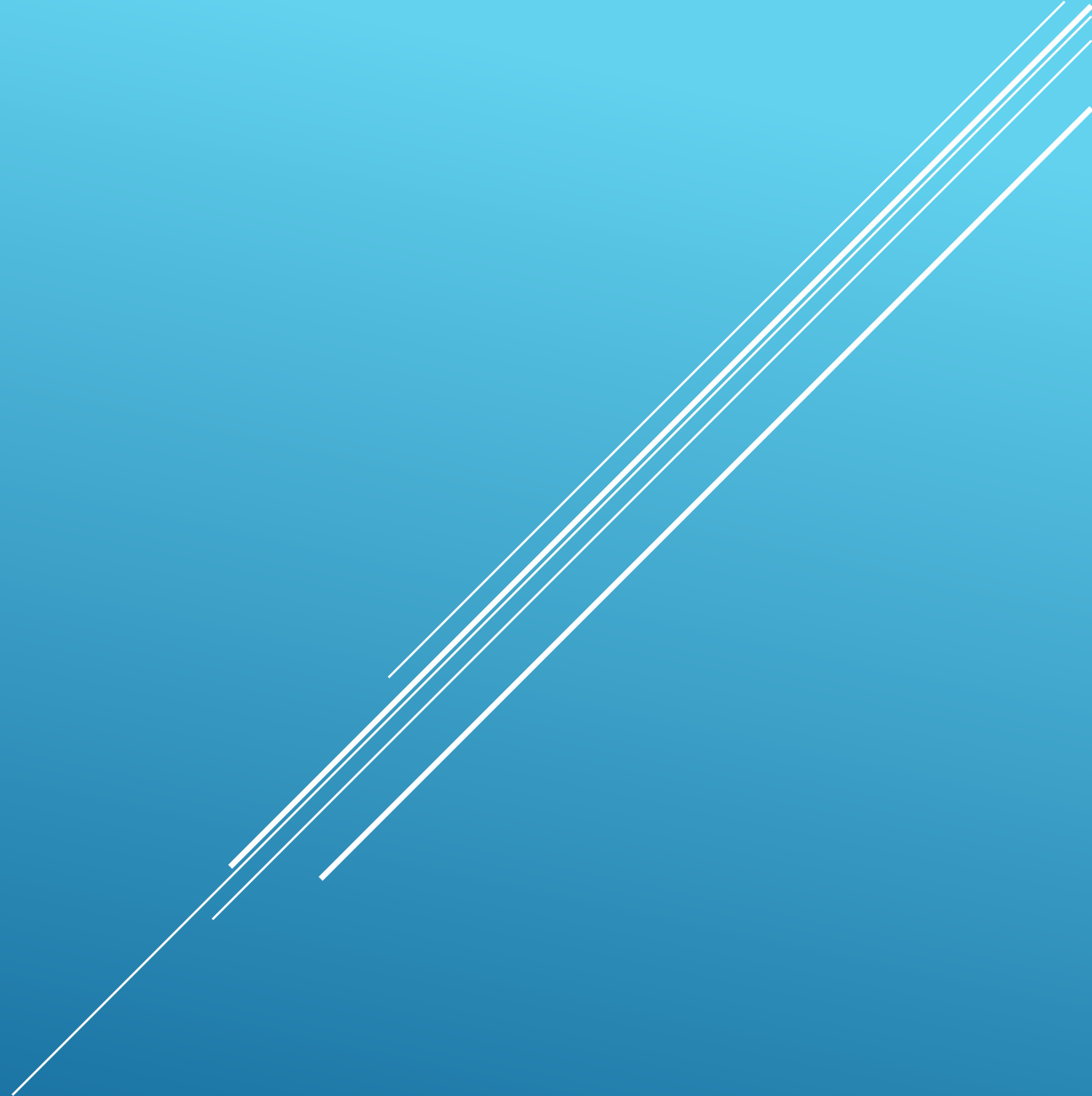


# PYTHON

## Типы данных



**Переменная** – это ячейка памяти, хранящая информацию разных типов, доступ к которой осуществляется по заданному ранее имени.

При использовании *переменной* значение ячейки может *меняться* в ходе программы.

При использовании *константы* значение ячейки остается *неизменным*.

Важно знать, какого типа переменные используются в программе, хотя, в отличие от других языков программирования, Python не требует их описания.

## Рассмотрим основные типы переменных, используемые в программировании на языке Python.

1. Различают три вида чисел: целые, вещественные (дробные) и комплексные, но при составлении простых программ обычно используют целые и вещественные.

Целые (int)	5, 16, 208, -96	a = 1
Вещественные (float)	-8.9, 5.6, 2.3	a = 5.6

**Строка (str)** – набор символов, заключенных в одинарные или двойные кавычки.

Например, a='Hello, World!' и a=' 'Hello, World!' ' – идентичные записи.

Также возможно использование тройных апострофов или кавычек при написании длинных текстов:

a = '''Hello, Maral! How are you?'''	a = """Hello, Maral! How are you? """
Записи идентичны.	

### Логический тип (bool).

Переменные логического типа могут принимать два значения: Истина (True), Ложь (False).

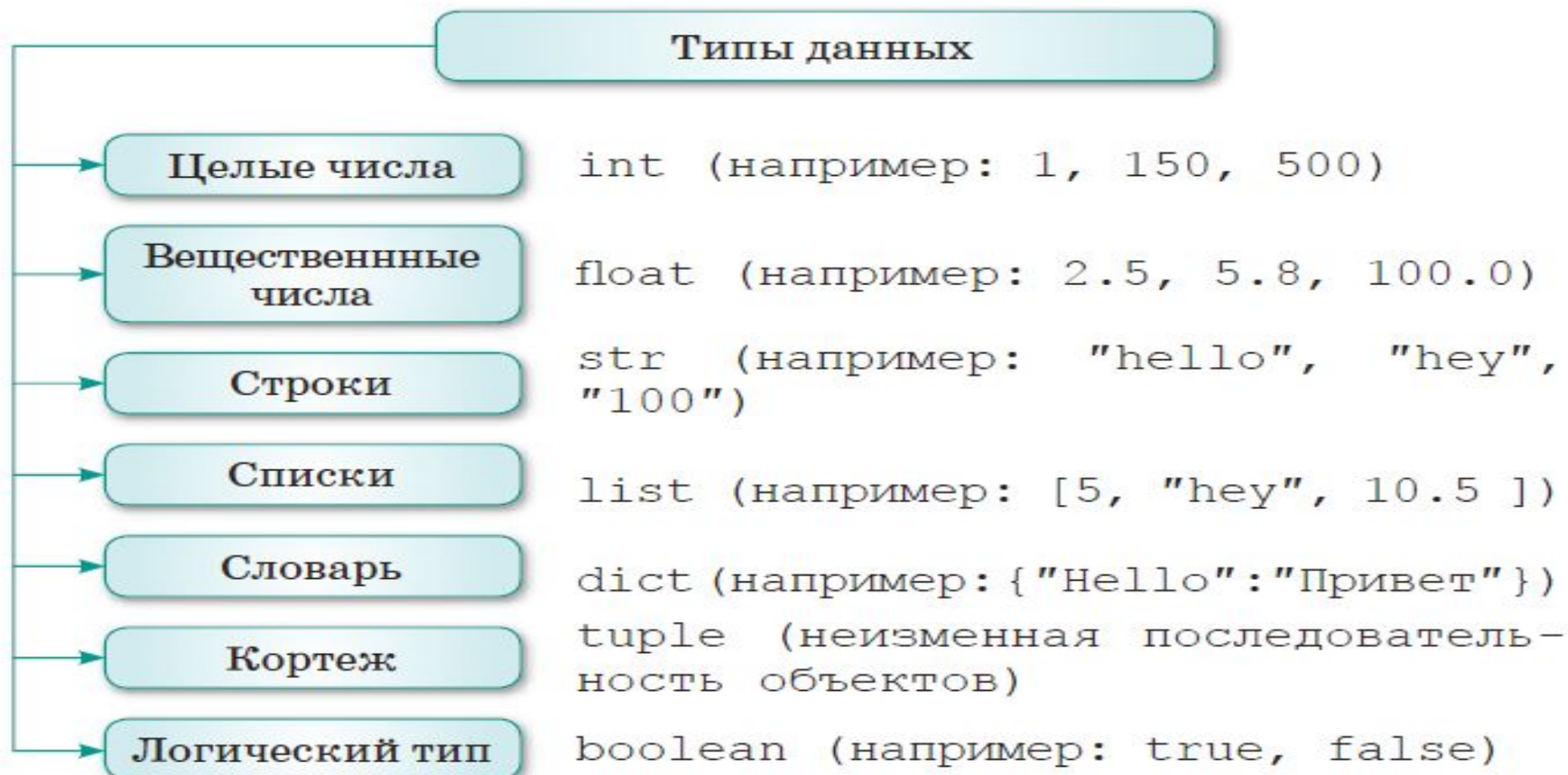
Запомни! «True» и «False» необходимо писать с большой буквы, иначе программа выдаст ошибку.

В отличие от других языков программирования, в Python есть функция, которая преобразовывает переменную любого типа в логический тип.

Значение = True		Значение = False	
Непустая строка	bool('123')	Пустая строка	bool("")
Ненулевое число	bool(123) bool(23.45)	Нулевое число	bool(0) bool(0.0)

Типы данных – множество допустимых значений, которые могут принимать данные, принадлежащие к этому типу, то есть **тип** – это описание, данное значениям допустимых данных.

Существует несколько видов типов данных языка Python



# ТИПЫ ДАННЫХ PYTHON

- ▶ **Логический**, может принимать одно из двух значений — True (истина) или False (ложь).
- ▶ **Числа**, могут быть целыми (1 и 2), с плавающей точкой (1.1 и 1.2), дробными (1/2 и 2/3), и даже комплексными.
- ▶ **Строки** — последовательности символов Юникода, например, HTML-документ.
- ▶ **Байты** и **массивы байтов**, например, файл изображения в формате JPEG.
- ▶ **Списки** — упорядоченные последовательности значений.
- ▶ **Кортежи** — упорядоченные неизменяемые последовательности значений.
- ▶ **Множества** — неупорядоченные наборы значений.
- ▶ **Словари** — неупорядоченные наборы пар вида ключ-значение.

# ЛОГИЧЕСКИЙ

- ▶ Логический тип данных может принимать одно из двух значений: истина или ложь (True/False)
- ▶ Результатом вычисления выражений также может быть логическое значение.

```
if size < 0:  
    raise ValueError('число должно быть неотрицательным')
```

- ▶ Логические значения могут трактоваться как числа. True как 1, и False как 0.

# ЧИСЛА

- ▶ Python поддерживает как целые числа, так и с плавающей точкой.
- ▶ Нет необходимости объявлять тип для их различия; Python определяет его по наличию или отсутствию десятичной точки.
- ▶ Можно использовать функцию `type()` для проверки типа любого значения или переменной.

```
>>> type(1000)
<class 'int'>
```

- ▶ Функцию `isinstance(переменная, тип)` тоже можно использовать для проверки принадлежности значения или переменной определенному типу.

# ЦЕЛЫЕ ЧИСЛА И ЧИСЛА С ПЛАВАЮЩЕЙ ТОЧКОЙ

- ▶ `float`(целое) – преобразование в число с плавающей точкой
- ▶ `int`(дробь) – преобразование дроби в целое отбрасывая дробную часть
- ▶ Точность чисел с плавающей точкой равна 15 десятичным знакам в дробной части.
- ▶ Целые числа могут быть сколь угодно большими.
- ▶ Если ваше целое число больше чем  $2^{32} - 1$ , операции с ним будут медленными.



# ПОВТОРЕНИЕ. ОСНОВНЫЕ ОПЕРАЦИИ С ЧИСЛАМИ

- ▶ + Сложение двух чисел:

```
print(6 + 2) # 8
```

- ▶ - Вычитание двух чисел:

```
print(6 - 2) # 4
```

- ▶ \* Умножение двух чисел:

```
print(6 * 2) # 12
```

- ▶ / Деление двух чисел:

```
print(6 / 2) # 3.0
```

- ▶ // Целочисленное деление двух чисел:

```
print(7 / 2) # 3.5
```

```
print(7 // 2) # 3
```

Данная операция возвращает целочисленный результат деления, отбрасывая дробную часть

- ▶ \*\* Возведение в степень:

```
print(6 ** 2) # Возводим число 6 в степень 2. Результат - 36
```

- ▶ % Получение остатка от деления:

```
print(7 % 2) # Получение остатка от деления числа 7 на 2. Результат - 1
```

Пример 1. Найти значение выражения  $\frac{(5x-2)^2+2y}{2xy} - 3x$

```
File Edit Format Run Options Window H
x=int(input())
y=int(input())
k=((5*x-2)**2 + 2*y)/(2*x*y)-3*x
print('k=', k, sep='')
>>>
=====
1
1
k=2.5
>>>
```

- ▣ # - комментарий (содержимое комментария не отображается в консоли)

## Пример 2. Функция `type()`

Тип переменной можно определить с помощью функции `type()`.

Пример использования приведен ниже

```
402.py - D:/Рабо
File Edit Format Run Options Window He
x=int(input())
y=int(input())
k=((5*x-2)**2 + 2*y)//(2*x*y)-3*x
print('k=',k,sep='')

1
2
k=0
>>> type(k)
<class 'int'>
>>>
```

```
402.py - D:/Рабо
File Edit Format Run Options Window He
x=int(input())
y=int(input())
k=((5*x-2)**2 + 2*y)/(2*x*y)-3*x
print('k=',k,sep='')

1
1
k=2.5
>>> type(k)
<class 'float'>
>>>
```

# ФОРМАТИВНОЕ ОЦЕНИВАНИЕ.

Пройдите по ссылке и выполните ФО

<https://onlinetestpad.com/jo5sgc3y7f4te>

Оценка за формативное оценивание будет выставлена в кунделик.