

Ilya Savin Python HW 2

1. Составить таблицу соответствия между различными объектами основных классов: `int`, `str` и объектами класса `bool`.
2. Разобраться с различиями между `input()` и `raw_input()`. Также в контексте разных версий python: 2 и 3.
3. Найти и прочитать PEP про изменение `print` между python2 и python3.

Составить таблицу соответствия между различными объектами основных классов: `int`, `str` и объектами класса `bool`.

int	str	bool
Поддерживается длинная арифметика, нельзя получить ошибку переполнения	неизменяемы, невозможно сделать <code>str = "Stop"</code> , <code>str[2] = '0'</code> и получить <code>St0p</code> . все работы со строками возвращают новую строку	Булевы значения в Python это две константы <code>True</code> и <code>False</code> . Также истинное значение: любое ненулевое число, любая непустая строка, любой непустой объект ложное значение: <code>0</code> , <code>None</code> , пустая строка, пустой объект

Разобраться с различиями между `input()` и `raw_input()`. Также в контексте разных версий python: 2 и 3.

Python 2:

- `input()` пытается определить тип принимаемого значения если ввести `100`, то `input()` вернет объект типа `int`, если `3.1415` - `float`, а если `abracadbra` - то строку, если будет математическое выражение - то вычислит его
- `raw_input()` всегда возвращает строку

Python 3:

- `input()` поступает аналогично `raw_input()` в Python2

Найти и прочитать PEP про изменение `print` между python2 и python3.

<https://www.python.org/dev/peps/pep-3105/>

`print` становится вместо ключевого слова истинной функцией с переменным числом параметров, и как пример во второй версии работало так

```
>>> print ("hello")
hello
>>> print ("hello", "world")
('hello', 'world')
```

А в третьей

```
>>> print ("Hello")
Hello
>>> print ("Hello", "world")
Hello world
```