In [1]:
```
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"
```

Python 3

строки, байты, работа с файлами

MIPT 2020

In [2]:
```
s = "string"
"string" is 'string'
id("string") == id('string')
id("string")
s[1]
s[2:4]
```

```
<>:2: SyntaxWarning: "is" with a literal. Did you mean "=="?
<>:2: SyntaxWarning: "is" with a literal. Did you mean "=="?
<ipython-input-2-7f54ad4d6884>:2: SyntaxWarning: "is" with a literal. Did you m
ean "=="?
  "string" is 'string'
```

Out[2]: True

Out[2]: True

Out[2]: 140337854289456

Out[2]: 't'

Out[2]: 'ri'

In [3]:
```
'aa' * 10000 is "aa" * 10000
```

Out[3]: False

In [4]:
```
s += "abadaba"
```

In [5]:
```
a = ""
for i in range(10):
    a += "bb"
    print(id(a))
```

```
140337823443696
140337718124016
140337718124016
140337718124016
140337718124016
140337718124016
140337718124016
140337675647696
140337675647696
140337675647696
```

In [6]:
```
a = ""
for i in range(10):
    a += "bb"
    print(a)
    print(id(a))
```

```
bb
140337823443696
bbbb
140337675643440
bbbbbb
140337675644592
bbbbbbbb
140337675555760
bbbbbbbbbb
140337675556272
bbbbbbbbbbbb
140337675860720
bbbbbbbbbbbbbb
140337675670320
bbbbbbbbbbbbbbbb
140337675648736
bbbbbbbbbbbbbbbbbb
140337675648816
bbbbbbbbbbbbbbbbbbbb
140337675648896
```

In [7]:
```
s[1] = 2
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-7-f2266874e093> in <module>
----> 1 s[1] = 2

TypeError: 'str' object does not support item assignment
```

In [8]:
```
s = list("string1")
s[0] = "S"
' '.join(s)
s
```

Out[8]:
```
'S t r i n g 1'
```

Out[8]:
```
['S', 't', 'r', 'i', 'n', 'g', '1']
```

In [9]:
```
words = {'some', 'order', 'words'}
a = [x for x in words]
' '.join(a)
```

Out[9]:
```
'order some words'
```

In [10]:
```
words = "A  few words".split()
words
' '.join(words)
```

Out[10]:
```
['A', 'few', 'words']
```

Out[10]:
```
'A few words'
```

In [11]:
```
"str" + "ing"
"a" * 10
```

Out[11]: 'string'

Out[11]: 'aaaaaaaaaa'

In [12]:
```
"str" + 10
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-12-54f143868309> in <module>
----> 1 "str" + 10

TypeError: can only concatenate str (not "int") to str
```

In [13]:
```
s = 'abadaba'
'abada' in s
```

Out[13]: True

In [14]:
```
', '.join(dir("string"))
```

Out[14]:
```
'__add__, __class__, __contains__, __delattr__, __dir__, __doc__, __eq__, __for
mat__, __ge__, __getattribute__, __getitem__, __getnewargs__, __gt__, __hash__,
__init__, __init_subclass__, __iter__, __le__, __len__, __lt__, __mod__, __mul_
_, __ne__, __new__, __reduce__, __reduce_ex__, __repr__, __rmod__, __rmul__, __
setattr__, __sizeof__, __str__, __subclasshook__, capitalize, casefold, center,
count, encode, endswith, expandtabs, find, format, format_map, index, isalnum,
isalpha, isascii, isdecimal, isdigit, isidentifier, islower, isnumeric, isprint
able, isspace, istitle, isupper, join, ljust, lower, lstrip, maketrans, partiti
on, replace, rfind, rindex, rjust, rpartition, rsplit, rstrip, split, splitline
s, startswith, strip, swapcase, title, translate, upper, zfill'
```

In [15]:
```
"Word".lower()
"Word".upper()
"Word".swapcase()
"word lala".title()
"word lala another".capitalize()
```

Out[15]: 'word'

Out[15]: 'WORD'

Out[15]: 'wORD'

Out[15]: 'Word Lala'

Out[15]: 'Word lala another'

In [16]:
```
" aaa ".rstrip()
" aaa ".lstrip()
" aaa \t".strip()
```

Out[16]: ' aaa'

Out[16]: 'aaa '

Out[16]: 'aaa'

In [17]:
```
"aaaaa".replace("aa", "2a")
```

Out[17]: '2a2aa'

In [18]:
```
"abc".startswith("a")
"abc".endswith("c")
```

Out[18]: True

Out[18]: True

In [19]:
```
"aaa\nbb\n\nc".split()
"aaa\nbb\n\nc".splitlines()
"aaa\nbb\n\nc".split('\n')
```

Out[19]: ['aaa', 'bb', 'c']

Out[19]: ['aaa', 'bb', '', 'c']

Out[19]: ['aaa', 'bb', '', 'c']

In [20]:
```
strings = ["a", "1", "1.0", " ", "\t"]

for s in strings:
    print(f"'{s}'", s.isalpha(), s.isdigit(), s.isspace(), sep="\t")
```

```
'a'     True    False   False
'1'     False   True    False
'1.0'   False   False   False
' '     False   False   True
'       '       False   False   True
```

String formatting

In [21]:
```
f"2 + 2 = {2 + 2}"
"2 + 2 = {}".format(2+2)
"2 + 2 = %s" % (2+2) # very old variant
```

Out[21]: '2 + 2 = 4'

Out[21]: '2 + 2 = 4'

Out[21]: '2 + 2 = 4'

In [22]:
```
"{1} {2} {1} {3}".format(1, 2, 3, 4)
```

Out[22]: '2 3 2 4'

In [23]:
```
i = 1
s = "s"
d = {}

f"a {s} {i}"
f"{ 'inner:' + s }"
f"{3.1415:.2f}"
f"{{}}"
f"{'{}'}"
```

Out[23]: 'a s 1'

Out[23]: 'inner:s'

Out[23]: '3.14'

Out[23]: '{}'

Out[23]: '{}'

In [24]:
```
repr(x for x in range(5))
str(x for x in range(5))
```

Out[24]:
```
'<generator object <genexpr> at 0x7fa2e94d8040>'
```

Out[24]:
```
'<generator object <genexpr> at 0x7fa2e8beac80>'
```

In [25]:
```
s = "{} {}"
i = 1

s.format(i, s)
s.format(1, 2)

"{1} {0} {2} {0}".format("first", "second", "third")

"{:.2f}".format(3.1415)

"{{}} {}".format("a")

"x = {x}, y = {y}".format(1, y=3)
```

Out[25]:
```
'1 {} {}'
```

Out[25]:
```
'1 2'
```

Out[25]:
```
'second first third first'
```

Out[25]:
```
'3.14'
```

Out[25]:
```
'{} a'
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-25-7960d2dc3bef> in <module>
     11 "{{}} {}".format("a")
     12
---> 13 "x = {x}, y = {y}".format(1, y=3)

KeyError: 'x'
```

string

In [26]:
```
import string

string.ascii_letters
string.ascii_lowercase
string.ascii_uppercase
string.digits
string.hexdigits
string.octdigits
string.whitespace
```

Out[26]:
```
'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'
```

Out[26]:
```
'abcdefghijklmnopqrstuvwxyz'
```

Out[26]:
```
'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
```

Out[26]:
```
'0123456789'
```

Out[26]:
```
'0123456789abcdefABCDEF'
```

Out[26]:
```
'01234567'
```

Out[26]:
```
' \t\n\r\x0b\x0c'
```

In [27]:
```
import sys

sys.getsizeof("abbb😊😊😊😊")
```

Out[27]: 108

In [28]:
```
smile = "😊"

'a'.encode()
'a'.encode('utf-32')

smile.encode()
smile.encode('utf-8')
smile.encode('utf-16')
smile.encode('utf-32')
smile.encode("ascii")
```

Out[28]: b'a'

Out[28]: b'\xff\xfe\x00\x00a\x00\x00\x00'

Out[28]: b'\xf0\x9f\x98\x8a'

Out[28]: b'\xf0\x9f\x98\x8a'

Out[28]: b'\xff\xfe=\xd8\n\xde'

Out[28]: b'\xff\xfe\x00\x00\n\xf6\x01\x00'

```
---------------------------------------------------------------------------
UnicodeEncodeError                        Traceback (most recent call last)
<ipython-input-28-1bc00bae1fbb> in <module>
      8 smile.encode('utf-16')
      9 smile.encode('utf-32')
---> 10 smile.encode("ascii")

UnicodeEncodeError: 'ascii' codec can't encode character '\U0001f60a' in positi
on 0: ordinal not in range(128)
```

In [29]:
```
"string".encode("utf-8").decode("utf-8")

smile = "smile: 😊"
a = smile.encode('ascii', errors="replace")  # errors='ignore'
a.decode('ascii')
```

Out[29]: 'string'

Out[29]: 'smile: ?'

In [30]:
```
a = bytearray()
a.extend(b'hey there')
a[5]
ord('h'), ord('😊')
```

Out[30]: 104

Out[30]: (104, 128522)

In [31]:
```
chr(128522)
```

Out[31]: '😊'

In [32]:
```
with open("hello_utf16le.txt", encoding="utf-16le", mode="w") as file:
    file.write("Hello!")

file = open("hello_utf16le.txt", "rb")
binary = file.read()
file.close()

binary
binary2 = b'\x00H\x00e\x00l\x00l\x00o\x00!'

binary.decode('utf-16le')
binary2.decode('utf-16be')
```

Out[32]: 6

Out[32]: b'H\x00e\x00l\x00l\x00o\x00!\x00'

Out[32]: 'Hello!'

Out[32]: 'Hello!'

In [33]:
```
with open("hello.txt", mode="w+b") as file:
    file.write(b"Hello!")
    file.tell()
    file.seek(0)
    file.read()
    file.seek(-2, io.SEEK_END)
    file.read()
```

Out[33]: 6

Out[33]: 6

Out[33]: 0

Out[33]: b'Hello!'

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-33-217dc07e8271> in <module>
      4     file.seek(0)
      5     file.read()
----> 6     file.seek(-2, io.SEEK_END)
      7     file.read()

NameError: name 'io' is not defined
```

modes:

- w - write
- r - read
- a - append
- w+ - read and write
- x - create new file, error if exists
- t - text mode (default)
- b - binary mode

In [34]:
```
import io

with io.open("io_hello", "w") as out:
    out.write("the same")
```

Out[34]: 8

In [35]:
```
ss = io.StringIO("init value\nsecond value\n")

print(ss.readline().strip())
for line in ss.readlines():
    print(line)

ss.getvalue()
ss.close()

with io.StringIO("init value\nsecond value\n") as ss:
    print(ss.readline().strip())
    for line in ss.readlines():
        print(line)
    ss.getvalue()
```

```
init value
second value
```

Out[35]:
```
'init value\nsecond value\n'
```

```
init value
second value
```

Out[35]:
```
'init value\nsecond value\n'
```