In [2]:

```
# operatii cu numere
# adunare
2 + 2
```

Out[2]:

```
4
```

In [2]:

```
x = 50
y = 5
z = 6
x - y * z
```

Out[2]:

```
20
```

In [3]:

```
(50 - 5 * 6) / 4
```

Out[3]:

```
5.0
```

In [5]:

```
8 / 5 # impartirea returneaza un numar de tip "float"
```

Out[5]:

```
1.6
```

In [4]:

```
a = 20
type(a)
```

Out[4]:

```
int
```

In [8]:

```
b = 1.6
type(b)
```

Out[8]:

```
float
```

In [9]:

```
8 // 5 # pentru a obtine catul impartirii
```

Out[9]:

```
1
```

In [10]:

```
8 % 5 # pentru a obtine restul impartirii
```

Out[10]:

```
3
```

In [10]:

```
5 ** 2 # 5 la puterea a doua sau pow(5, 2)
```

Out[10]:

```
25
```

In [14]:

```
abs(-2)
```

Out[14]:

```
2
```

In [15]:

```
int(2.6) # se converteste la numar intreg
```

Out[15]:

```
2
```

In [16]:

```
float(4) # se converteste la numar de tip "float"
```

Out[16]:

```
4.0
```

In [2]:

```
u = complex(2, 3) # se creeaza un numar complex cu parte reala 2 si parte imaginara 3
u
```

Out[2]:

```
(2+3j)
```

In [22]:

```
u.conjugate() # conjugatul numarului complex u
```

Out[22]:

```
(2-3j)
```

In [28]:

```
divmod(8, 5) # perechea (8 // 5, 8 % 5)
```

Out[28]:

```
(1, 3)
```

In [6]:

```
'string in Python'
```

Out[6]:

```
'string in Python'
```

In [7]:

```
"string in Python"
```

Out[7]:

```
'string in Python'
```

In [1]:

```
s = 'o linie de cod'
print(s)
```

```
o linie de cod
```

In [36]:

```
3 * "doi" + "trei"
```

Out[36]:

```
'doidoidoitrei'
```

In [3]:

```
str = 'Py' 'thon'
str
```

Out[3]:

```
'Python'
```

In [38]:

```
prefix = 'Py'
prefix 'thon' # nu putem concatena o variabila si un string
```

```
  File "<ipython-input-38-779914c62160>", line 2
    prefix 'thon'
                ^
SyntaxError: invalid syntax
```

In [40]:

```
prefix = 'Py'
prefix + 'thon' # aceasta este varianta corecta pentru a concatena o variabila si un string
```

Out[40]:

```
'Python'
```

In [2]:

```
word = 'Python'
word[0] # primul caracter are indexul 0
```

Out[2]:

```
'P'
```

In [45]:

```
word[-1] # ultimul caracter din string
```

Out[45]:

```
'n'
```

In [47]:

```
word[0:2] # caracterele de la pozitia 0 (inclusiv) la pozitia 2 (exclusiv)
```

Out[47]:

```
'Py'
```

In [8]:

```
word[:2] # caracterele de la inceputul stringului pana la pozitia 2 (exclusiv)
```

Out[8]:

```
'Py'
```

In [49]:

```
word[4:] # caracterele de la pozitia 4 (inclusiv) pana la sfarsitul stringului
```

Out[49]:

```
'on'
```

In [50]:

```
'J' + word[1:]
```

Out[50]:

```
'Jython'
```

In [51]:

```
len(word)
```

Out[51]:

```
6
```

In [3]:

```
squares = [1, 4, 9, 16, 25]
squares
```

Out[3]:

```
[1, 4, 9, 16, 25]
```

In [53]:

```
squares[-1]
```

Out[53]:

```
25
```

In [55]:

```
squares[-3:]
```

Out[55]:

```
[9, 16, 25]
```

In [5]:

```
squares + [36, 49]
```

Out[5]:

```
[1, 4, 9, 16, 25, 36, 49]
```

In [10]:

```
cubes = [1, 8, 27, 65]
4 ** 3
```

Out[10]:

```
64
```

In [11]:

```
cubes[3] = 64 # este posibil sa schimbam continutul unei liste
cubes
```

Out[11]:

```
[1, 8, 27, 64]
```

In [12]:

```
cubes.append(125)
cubes
```

Out[12]:

```
[1, 8, 27, 64, 125]
```

In [13]:

```
len(cubes)
```

Out[13]:

```
5
```

In [14]:

```
cubes[1] = 0
cubes
```

Out[14]:

```
[1, 0, 27, 64, 125]
```

In [15]:

```
cubes[1:2] = []
cubes
```

Out[15]:

```
[1, 27, 64, 125]
```

In [16]:

```
cubes[1:3]=[1, 2, 3]
cubes
```

Out[16]:

```
[1, 1, 2, 3, 125]
```

In [3]:

```
type('f')
```

Out[3]:

```
str
```

In [5]:

```
type("f")
```

Out[5]:

```
str
```

In [8]:

```
chr(1)
```

Out[8]:

```
'\x01'
```

In [10]:

```
list = [1, 2, 4, 5]
list1 = list[0:2] + [3] + list[2:]
list1
```

Out[10]:

```
[1, 2, 3, 4, 5]
```

In [11]:

```
list[200] = 4
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-11-aa9d6eb7c4c9> in <module>
----> 1 list[200] = 4

IndexError: list assignment index out of range
```

In [12]:

```
list + 34
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-12-f845326848a6> in <module>
----> 1 list + 34

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

In [ ]:

```

```