

# NumPy

NumPy - это библиотека с открытым исходным кодом для языка программирования Python.

## **Возможности:**

- поддержка многомерных массивов (включая матрицы);
- поддержка высокоуровневых математических функций, предназначенных для работы с многомерными массивами.

Написана с использованием языков C и *Fortran*

# NumPy

## **Области применения:**

- Обработка изображений;
- Линейная алгебра;
- Биология;
- Машинное обучение;
- и многие другие...

# Использование NumPy

- **SciPy** - фундаментальная библиотека для научных вычислений
- **Matplotlib** - мощная библиотека для визуализации
- **IPython** - улучшенная интерактивная консоль
- **Sympy** - символьные вычисления
- **pandas** - структуры данных и их анализ
- **scikit-learn** - машинное обучение
- **scikit-image** - обработка изображений
- **Biopython** - вычислительная молекулярная биология
- и многие другие...

# Создание массивов с использованием модуля NumPy

## Массивы в NumPy:

- Эффективны;
- Однородны;
- Хранятся в памяти.

```
In [1]: import numpy as np
from IPython.display import Image
def describe_array(np_array):
    print(repr(np_array))
    print('Тип элементов массива: {}'.format(np_array.dtype))
    print('Число измерений (осей): {}'.format(np_array.ndim))
    print('Размеры (форма): {}'.format(np_array.shape))
    print('Размер каждого элемента, байт: {}'.format(np_array.itemsize))
```

# Одномерные массивы

```
In [2]: vector = np.array([0, 2, 4, 6, 8])  
describe_array(vector)  
Image('1d_array.jpeg')
```

```
array([0, 2, 4, 6, 8])  
Тип элементов массива: int64  
Число измерений (осей): 1  
Размеры (форма): (5,)  
Размер каждого элемента, байт: 8
```

Out[2]:

0	2	4	6	8
---	---	---	---	---

```
In [3]: vector = np.arange(0, 9, 2)
describe_array(vector)
Image('1d_array.jpeg')
```

```
array([0, 2, 4, 6, 8])
Тип элементов массива: int64
Число измерений (осей): 1
Размеры (форма): (5,)
Размер каждого элемента, байт: 8
```

Out[3]:

0	2	4	6	8
---	---	---	---	---

```
In [4]: vector = np.linspace(0, 8, 5, dtype='int64')  
describe_array(vector)  
Image('1d_array.jpeg')
```

```
array([0, 2, 4, 6, 8])  
Тип элементов массива: int64  
Число измерений (осей): 1  
Размеры (форма): (5,)  
Размер каждого элемента, байт: 8
```

Out[4]:

0	2	4	6	8
---	---	---	---	---

# Двумерные массивы

```
In [5]: matrix = np.ones(shape=[3, 3], dtype=np.int64)
describe_array(matrix)
Image('2d_array.jpeg')
```

```
array([[1, 1, 1],
       [1, 1, 1],
       [1, 1, 1]])
```

Тип элементов массива: int64

Число измерений (осей): 2

Размеры (форма): (3, 3)

Размер каждого элемента, байт: 8

Out[5]:

1	1	1
1	1	1
1	1	1



```
In [6]: matrix = np.array([[1, 1, 1], [1, 1, 1], [1, 1, 1]])  
describe_array(matrix)  
Image('2d_array.jpeg')
```

```
array([[1, 1, 1],  
       [1, 1, 1],  
       [1, 1, 1]])
```

Тип элементов массива: int64

Число измерений (осей): 2

Размеры (форма): (3, 3)

Размер каждого элемента, байт: 8

Out[6]:

1	1	1
1	1	1
1	1	1

```
In [7]: matrix = np.eye(3, dtype=int)
        describe_array(matrix)
```

```
array([[1, 0, 0],
       [0, 1, 0],
       [0, 0, 1]])
```

Тип элементов массива: int64

Число измерений (осей): 2

Размеры (форма): (3, 3)

Размер каждого элемента, байт: 8

```
In [8]: matrix = np.diag([1, 2, 3])  
describe_array(matrix)
```

```
array([[1, 0, 0],  
       [0, 2, 0],  
       [0, 0, 3]])
```

Тип элементов массива: int64

Число измерений (осей): 2

Размеры (форма): (3, 3)

Размер каждого элемента, байт: 8

# Трёхмерные массивы

```
In [9]: three_d_array = np.zeros([3, 3, 5], dtype='int64')
        describe_array(three_d_array)
```

```
array([[[0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0]],

       [[0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0]],

       [[0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0],
        [0, 0, 0, 0, 0]]])
```

Тип элементов массива: int64

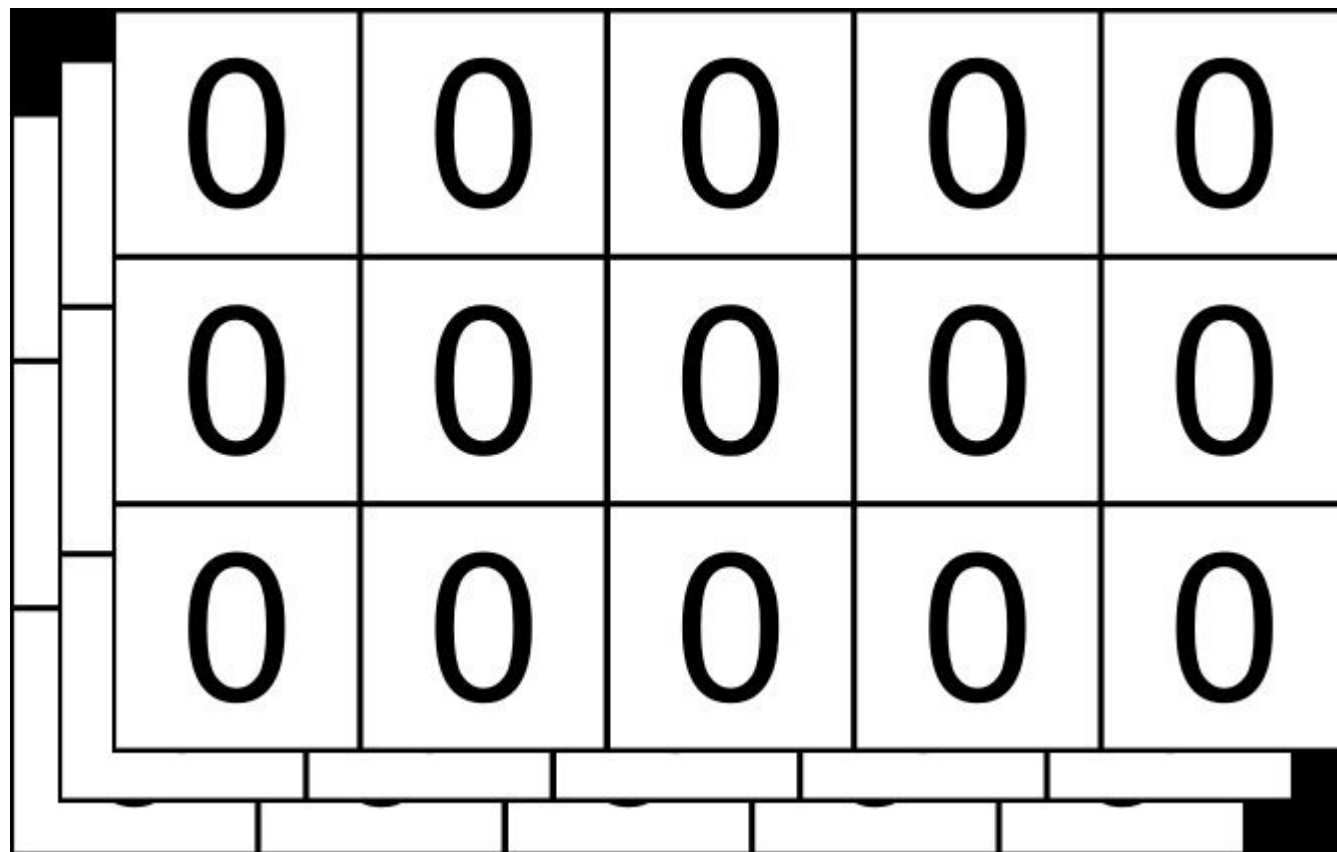
Число измерений (осей): 3

Размеры (форма): (3, 3, 5)

Размер каждого элемента, байт: 8

```
In [10]: Image('3d_array.jpeg')
```

```
Out[10]:
```



## Доступ к элементам массива NumPy

```
In [11]: matrix = np.random.rand(3, 3)
         describe_array(matrix)
```

```
array([[ 0.70614557,  0.13091445,  0.97986052],
       [ 0.20388896,  0.63337256,  0.72058994],
       [ 0.25695512,  0.23485504,  0.72023657]])
```

Тип элементов массива: float64

Число измерений (осей): 2

Размеры (форма): (3, 3)

Размер каждого элемента, байт: 8

```
In [12]: print('Матрица:\n{}'.format(matrix))
print('Вторая строка:', matrix[1])
print('Третий столбец:', matrix[:, 2])
print('Первый элемент третьей строки:', matrix[2, 0])
```

```
Матрица:
[[ 0.70614557  0.13091445  0.97986052]
 [ 0.20388896  0.63337256  0.72058994]
 [ 0.25695512  0.23485504  0.72023657]]
Вторая строка: [ 0.20388896  0.63337256  0.72058994]
Третий столбец: [ 0.97986052  0.72058994  0.72023657]
Первый элемент третьей строки: 0.256955120043
```

```
In [13]: matrix[1] = 0.5
matrix
```

```
Out[13]: array([[ 0.70614557,  0.13091445,  0.97986052],
                [ 0.5         ,  0.5         ,  0.5         ],
                [ 0.25695512,  0.23485504,  0.72023657]])
```

```
In [14]: matrix[1, 1] = 'Hello, world!'
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-14-dce008357541> in <module>()
----> 1 matrix[1, 1] = 'Hello, world!'
```

```
ValueError: could not convert string to float: 'Hello, world!'
```

```
In [15]: matrix[1, 1] = np.pi  
matrix
```

```
Out[15]: array([[ 0.70614557,  0.13091445,  0.97986052],  
                [ 0.5          ,  3.14159265,  0.5          ],  
                [ 0.25695512,  0.23485504,  0.72023657]])
```

```
In [16]: matrix[:, 2] = np.e  
matrix
```

```
Out[16]: array([[ 0.70614557,  0.13091445,  2.71828183],  
                [ 0.5          ,  3.14159265,  2.71828183],  
                [ 0.25695512,  0.23485504,  2.71828183]])
```



## Resources

- <https://docs.scipy.org/doc/numpy/reference/> (<https://docs.scipy.org/doc/numpy/reference/>)
- <https://docs.scipy.org/doc/numpy-dev/user/quickstart.html> (<https://docs.scipy.org/doc/numpy-dev/user/quickstart.html>)
- <https://docs.scipy.org/doc/numpy/reference/routines.html> (<https://docs.scipy.org/doc/numpy/reference/routines.html>)

## Слайды доступны по ссылке:

[https://github.com/antsyferov/numpy\\_slides](https://github.com/antsyferov/numpy_slides) ([https://github.com/antsyferov/numpy\\_slides](https://github.com/antsyferov/numpy_slides))