numpy: Broadcasting

In [1]:

```
import numpy as np
```

In [2]:

```
a = np.arange(9).reshape(3, 3)
print(a.shape)
print(a)
b = np.arange(4, 4+9).reshape(3, 3)
print(b.shape)
print(b)
```

```
(3, 3)
[[0 1 2]
 [3 4 5]
 [6 7 8]]
(3, 3)
[[ 4  5  6]
 [ 7  8  9]
 [10 11 12]]
```

In [3]:

```
a+b
```

Out[3]:

```
array([[ 4,  6,  8],
       [10, 12, 14],
       [16, 18, 20]])
```

So this is easy and one-to-one.

What if the shapes do not match?

In [4]:

```
a = np.arange(9).reshape(3, 3)
print(a.shape)
print(a)
b = np.arange(3)
print(b.shape)
print(b)
```

```
(3, 3)
[[0 1 2]
 [3 4 5]
 [6 7 8]]
(3,)
[0 1 2]
```

What will this do?

In [5]:

```
a+b
```

Out[5]:

```
array([[ 0,  2,  4],
       [ 3,  5,  7],
       [ 6,  8, 10]])
```

It has *broadcast* along the last axis!

Can we broadcast along the *first* axis?

In [6]:

```
a+b.reshape(3, 1)
```

Out[6]:

```
array([[ 0,  1,  2],
       [ 4,  5,  6],
       [ 8,  9, 10]])
```

Rules:

- Shapes are matched axis-by-axis from last to first.
- A length-1 axis can be *broadcast* if necessary.