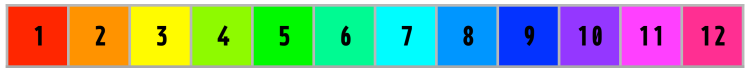
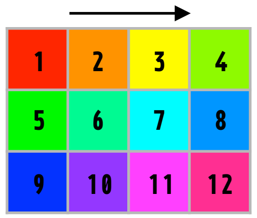
Python numpy reshape and stack cheatsheet

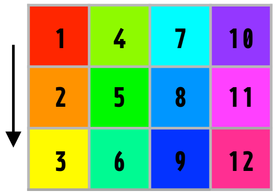
reshape & ravel

```
a1 = np.arange(1, 13)
```





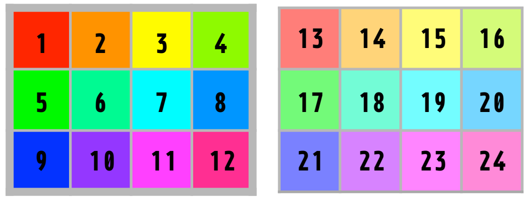
```
a1.reshape(3, 4)
a1.reshape(-1, 4)
a1.reshape(3, -1)
  .ravel()  # back to 1D
```



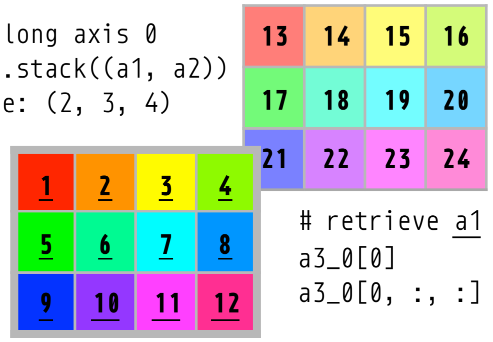
```
a1.reshape(3, -1, order='F')
  .ravel(order='F')  # back to 1D
```

3D array from 2D arrays

```
a1 = np.arange(1, 13).reshape(3, 4)
a2 = np.arange(13, 25).reshape(3, -1)
```



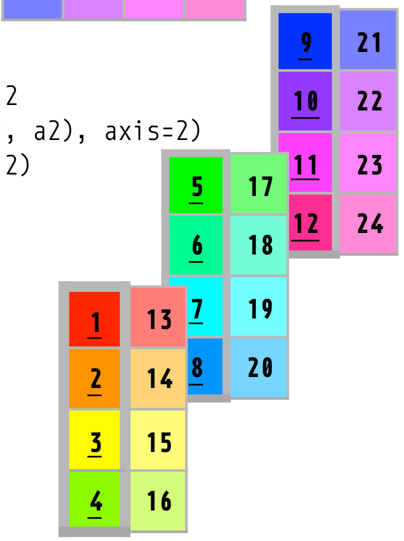
```
# stack along axis 0
a3_0 = np.stack((a1, a2))
a3_0.shape: (2, 3, 4)
```



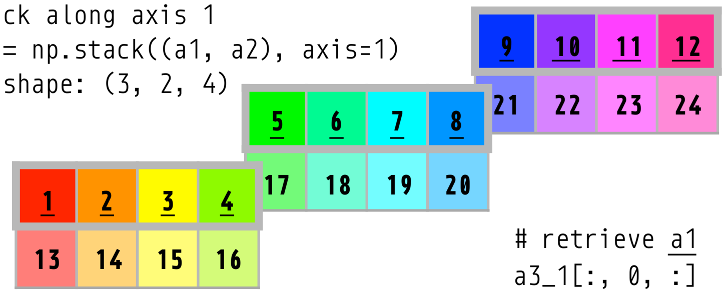
```
# retrieve a1
a3_0[0]
a3_0[0, :, :]
```

```
# stack along axis 2
a3_2 = np.stack((a1, a2), axis=2)
a3_2.shape: (3, 4, 2)
```

```
# retrieve a1
a3_2[:, :, 0]
```



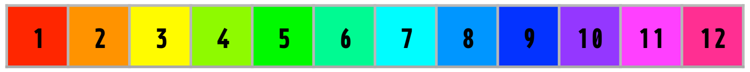
```
# stack along axis 1
a3_1 = np.stack((a1, a2), axis=1)
a3_1.shape: (3, 2, 4)
```



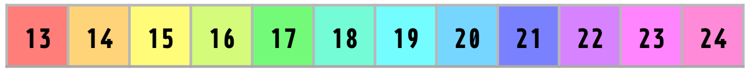
```
# retrieve a1
a3_1[:, 0, :]
```

stack

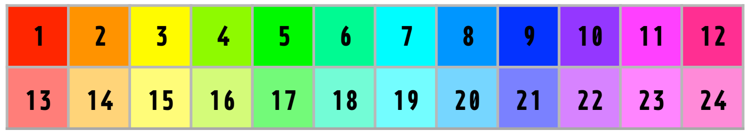
```
a1 = np.arange(1, 13)
```



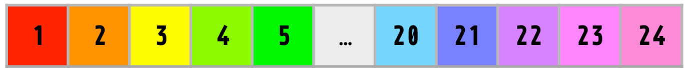
```
a2 = np.arange(13, 25)
```



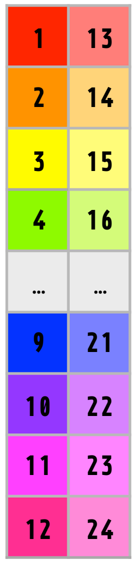
```
np.stack((a1, a2))
```



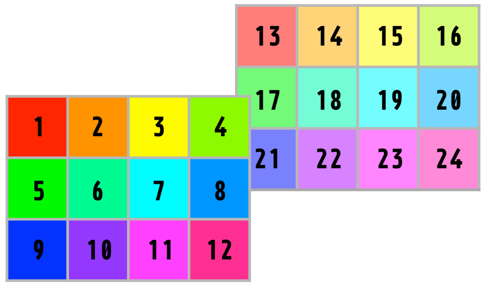
```
np.hstack((a1, a2))
```



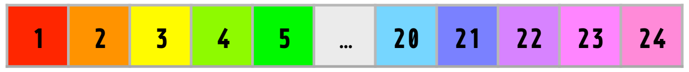
```
np.stack((a1, a2), axis=1)
```



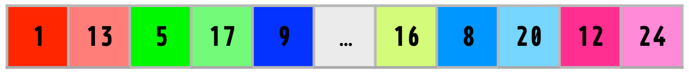
flatten 3D array



```
# flatten/ravel
a3_0.ravel()
```

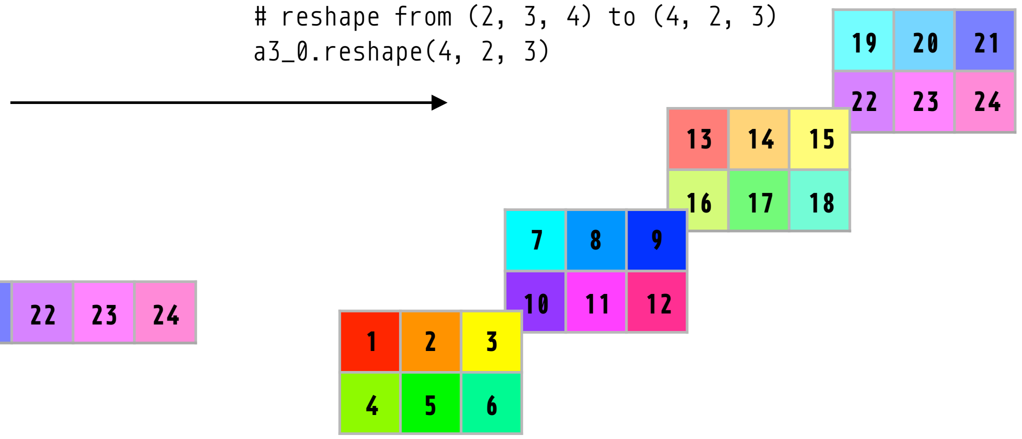


```
# flatten/ravel
a3_0.ravel(order='F')
```



reshape 3D array

```
# reshape from (2, 3, 4) to (4, 2, 3)
a3_0.reshape(4, 2, 3)
```



Created by Hause Lin. Tutorial on Medium.