

## ▼ About reduce\_mean

The functionality of `numpy.mean` and `tensorflow.reduce_mean` are the same

```
import numpy as np
import tensorflow as tf
```

```
c = np.array([[3.,4], [5.,6], [6.,7]])
print(c)
```

```
↳ [[3. 4.]
    [5. 6.]
    [6. 7.]]
```

```
m1 = np.mean(c, 1)
print(m1)
```

```
↳ [3.5 5.5 6.5]
```

`mean()`은 평균값을 의미하므로

- $(3 + 4) / 2 = 3.5$
- $(5 + 6) / 2 = 5.5$
- $(6 + 7) / 2 = 6.5$

```
m2 = tf.reduce_mean(c,1)
with tf.Session() as sess: # 세션을 실행시킬 때 세션 파트를 설정하고 이 세션 파트 안에서 밖에 세션
    result = sess.run(m2) # run()은 세션을 실행시키는 함수이다.
    print(result)
```

```
↳ [3.5 5.5 6.5]
```

## ▼ RESULT

결과적으로 `numpy`의 `mean()`함수와 `tensorflow`의 `reduce_mean()`함수와 같다.

```
test = np.mean(c, 0) # 열로 나눌 땐 인자 값으로 0을 주면 된다.
print(test)
```

```
↳ [4.66666667 5.66666667]
```

## numpy의 mean

- `mean(array, axis)` `axis`는 행렬의 행과 열을 의미 한다.

