numpy.dot

numpy.dot(*a*, *b*, *out=None*)

Dot product of two arrays.

For 2-D arrays it is equivalent to matrix multiplication, and for 1-D arrays to inner product of vectors (without complex conjugation). For N dimensions it is a sum product over the last axis of *a* and the second-to-last of *b*:

```
dot(a, b)[i,j,k,m] = sum(a[i,j,:] * b[k,:,m])
```

Parameters:

a : *array_like*

First argument.

b : *array_like*

Second argument.

out : *ndarray, optional*

Output argument. This must have the exact kind that would be returned if it was not used. In particular, it must have the right type, must be C-contiguous, and its dtype must be the dtype that would be returned for *dot(a,b)*. This is a performance feature. Therefore, if these conditions are not met, an exception is raised, instead of attempting to be flexible.

Returns:

output : *ndarray*

Returns the dot product of *a* and *b*. If *a* and *b* are both scalars or both 1-D arrays then a scalar is returned; otherwise an array is returned. If *out* is given, then it is returned.

Raises:

ValueError

If the last dimension of *a* is not the same size as the second-to-last dimension of *b*.

See also:

vdot **(numpy.vdot.html#numpy.vdot)** Complex-conjugating dot product.
tensordot **(numpy.tensordot.html#numpy.tensordot)** Sum products over arbitrary axes.
einsum **(numpy.einsum.html#numpy.einsum)** Einstein summation convention.
matmul **(numpy.matmul.html#numpy.matmul)** ‘@’ operator as method with out parameter.

Examples

```
>>> np.dot(3, 4)
12
```

Neither argument is complex-conjugated:

```
>>> np.dot([2j, 3j], [2j, 3j])
(-13+0j)
```

For 2-D arrays it is the matrix product:

```
>>> a = [[1, 0], [0, 1]]
>>> b = [[4, 1], [2, 2]]
>>> np.dot(a, b)
array([[4, 1],
       [2, 2]])
```

```
>>> a = np.arange(3*4*5*6).reshape((3,4,5,6))
>>> b = np.arange(3*4*5*6)[::-1].reshape((5,4,6,3))
>>> np.dot(a, b)[2,3,2,1,2,2]
499128
>>> sum(a[2,3,2,:] * b[1,2,:,2])
499128
```


Previous topic

numpy.linalg.LinAlgError (numpy.linalg.LinAlgError.html)

Next topic

numpy.vdot (numpy.vdot.html)