

## and, &

Find logical AND

### Syntax

```
A & B & ...
and(A, B)
```

### Description

`A & B & ...` performs a logical AND of all input arrays `A`, `B`, etc., and returns an array containing elements set to either logical 1 (true) or logical 0 (false). An element of the output array is set to 1 if all input arrays contain a nonzero element at that same array location. Otherwise, that element is set to 0.

Each input of the expression can be an array or can be a scalar value. All nonscalar input arrays must have equal dimensions. If one or more inputs are an array, then the output is an array of the same dimensions. If all inputs are scalar, then the output is scalar.

If the expression contains both scalar and nonscalar inputs, then each scalar input is treated as if it were an array having the same dimensions as the other input arrays. In other words, if input `A` is a 3-by-5 matrix and input `B` is the number 1, then `B` is treated as if it were a 3-by-5 matrix of ones.

`and(A, B)` is called for the syntax `A & B` when either `A` or `B` is an object.

**Note** The symbols `&` and `&&` perform different operations in the MATLAB® software. The element-wise AND operator described here is `&`. The short-circuit AND operator is `&&`.

### Examples

If matrix `A` is

```
0.4235    0.5798         0    0.7942         0
0.5155         0    0.7833    0.0592    0.8744
0.3340         0         0         0    0.0150
0.4329    0.6405    0.6808    0.0503         0
```

and matrix `B` is

```
0     1     0     1     0
1     1     1     0     1
0     1     1     1     0
0     1     0     0     1
```

then

```
A & B
```

```
ans =
```

```
0     1     0     1     0
1     0     1     0     1
0     0     0     0     0
0     1     0     0     0
```

### More About

- [Truth Table for Logical Operations](#)

## See Also

---

[all](#) | [any](#) | [bitand](#) | [Logical Operators: Short Circuit](#) | [not](#) | [or](#) | [xor](#)

---

**Introduced before R2006a**

---