**MATLAB Function Reference**

**zeros**
Create array of all zeros

**Syntax**

\[
B = \text{zeros}(n) \\
B = \text{zeros}(m,n) \\
B = \text{zeros}([m n]) \\
B = \text{zeros}(m,n,p,...) \\
B = \text{zeros}([m n p ...]) \\
B = \text{zeros}(\text{size}(A)) \\
\text{zeros}(m, n,...,\text{classname}) \\
\text{zeros}([m,n,...],\text{classname})
\]

**Description**

\(B = \text{zeros}(n)\) returns an \(n\)-by-\(n\) matrix of zeros. An error message appears if \(n\) is not a scalar.

\(B = \text{zeros}(m,n)\) or \(B = \text{zeros}([m n])\) returns an \(m\)-by-\(n\) matrix of zeros.

\(B = \text{zeros}(m,n,p,...)\) or \(B = \text{zeros}([m n p ...])\) returns an \(m\)-by-\(n\)-by-\(p\)-by-\(\ldots\) array of zeros.

**Note** The size inputs \(m, n, p, \ldots\) should be nonnegative integers. Negative integers are treated as 0.

\(B = \text{zeros}(\text{size}(A))\) returns an array the same size as \(A\) consisting of all zeros.

\(\text{zeros}(m, n,...,\text{classname})\) or \(\text{zeros}([m,n,...],\text{classname})\) is an \(m\)-by-\(n\)-by-\(\ldots\) array of zeros of data type \(\text{classname}\). \(\text{classname}\) is a string specifying the data type of the output. \(\text{classname}\) can have the following values: 'double', 'single', 'int8', 'uint8', 'int16', 'uint16', 'int32', 'uint32', 'int64', or 'uint64'.

**Example**

\[x = \text{zeros}(2,3,'\text{int8}')\];
Remarks

The MATLAB language does not have a dimension statement; MATLAB automatically allocates storage for matrices. Nevertheless, for large matrices, MATLAB programs may execute faster if the `zeros` function is used to set aside storage for a matrix whose elements are to be generated one at a time, or a row or column at a time. For example

```matlab
    x = zeros(1,n);
    for i = 1:n, x(i) = i; end
```

See Also

`eye`, `ones`, `rand`, `randn`, `complex`