

	X	Y	Z
!	separator		
!	(transpose) / permute	rot90	system
!	for	repmat	repelem (run-length decoding)
#	specify outputs	display stack (debug)	sound, soundsc, audiowrite
\$	specify inputs	char(vpa(...))	fopen, fwrite, fclose
%	comment	class	cast
&	alternative default input/output spec	intersect	and
'	Not used. String delimiter		bitand
'		run-length encoding	now / clock
(() assignment indexing / split	() assignment ind. with final : / split	() assignment ind. with initial : / split
)	() reference indexing	() reference ind. with final :	() reference ind. with initial :
*	*	kron	matrix product
+	+		Cartesian product
+		conv2	conv2(..., 'same') / cconv
-	do twice	cos	sin
-	-	setdiff	deconv
-	break	continue	pause
/	/	angle	matrix /
0	Not used	predefined literals	predefined literals
1	Not used	predefined literals	predefined literals
2	Not used	predefined literals	predefined literals
3	Not used	predefined literals	predefined literals
4	Not used	predefined literals	predefined literals
5	Not used	predefined literals	predefined literals
6	Not used	predefined literals	predefined literals
7	Not used	predefined literals	predefined literals
8	Not used	predefined literals	
9	Not used	predefined literals	
:	colon (range)	linearize array	comma-separated list
:		acos	asin
<	<	min	cummin
==	==	isequal	strcmp
>	>	max	cummax
?	if		why
@	"for" / "do twice" value / "while" index	"for" index	
@		perms	sparse
A	all(..., 1)	dec2base. Larger base, any symbols	randperm
B	logical(dec2bin(...)-'0')	dec2bin	base2dec. Larger base, any symbols
B		bin2dec(char(...+'0'))	bin2dec
C		im2col	im2col(..., 'distinct')
D	disp(num2str(..., ...)) / mat2str	disp(num2str(...))	sprintf / fprintf
D			disp
E	multiply by 2	replace elements in array	
F	Not used. False (literal)		exponents of prime factorization
F			fft, nfft
G	Paste from clipboard G (user-input)	plot	imwrite / imagesc / image / imshow
G			appearance of graphics / format
H	Paste from clipboard H	Copy to clipboard H	
I	Paste from clipboard I	Copy to clipboard I	col2im
I			image processing functions
J	Paste from clipboard J	Copy to clipboard J	
K	Paste from clipboard K	Copy to clipboard K	
L	Paste from clipboard L (multi-level)	Copy to clipboard L (multi-level)	gallery
L			
M	Paste from clipboard M (function-input)	mode	
N	stack size	nchoosek (array)	
N		NaN	isnan
O	zeros	datestr	datenum
O			datevec
P	flip	flipud	pi
P			pdist2
Q	increment by 1	accumarray	rat
Q			polyval / roots / polyfit / inpolygon
R	triu	triu(...,1) / build matrix	tril
R			tril(...,-1) / build matrix
S	sort	sortrows	circshift
S			sign / fftshift
T	Not used. True (literal)		toeplitz
U	str2num / string to array / square	str2double	
V	num2str		
W	2 raised to input		
X	Not used	regex	regexprep
Y	Not used		inf
Y			isinf
Z	Not used		
[Not used. Array delimiter	ind2sub	
[mod(...-1)+1	
\	mod	matrix \	divisors
]	end (loops or conditional branches)	sub2ind	
^	.^	sqrt	
^		matrix ^	Cartesian power
-	unary minus / normalize uint8		
do...while	do...while	tic	toc
any	any(..., 1)	padarray / unpad array	base2base
bubble	bubble	strsplit	
c	char (also for cell array)	strcat	strjoin / convert to '#' and char 0
d	diff	diag / spdiags	gcd
e	reshape / squeeze	blkdiag	exp
f	find	strfind	
f		factor	
g	logical / cell2mat	ndgrid	gamma / gammaln / betainc
g			gammaln / betaln
h	horzcat	{..., ...}	hankel
h			hypergeom
i	input	unread	imread
j	input(...,'s')	real	imag
j			conj / real and imag
k	lower / floor	upper / ceil	
k		closest values	
l	ones	clamp (limit to a range)	log. With two inputs, specifies base
l			log2
m	ismember	ismember(...,'rows')	mean
m			lcm
n	numel / size	nchoosek (numbers) / multinomial c.	poly / interp1
o	double / cell array to numeric / parity	int64	round / change case
o			fix
p	prod	prod(..., 1, ...)	cumprod
p			isprime / totient function
q	decrement by 1	quantile	n-th prime / next prime
q			primes
r	rand	randn	randi
r			randsample
s	sum	sum(..., 1, ...)	cumsum
s			std / cov / skewness / kurtosis
t	duplicate elements		strrep
t		unique(...,'rows')	strjust
u	unique		
v	vertcat	eig / svd / strtrim	symmetric range / array / deblank
w	swap		
x	delete from stack	clc	
y	duplicate element	eye	hypot
y			size
z	nnz	nonzeros / remove whitespace	
z	Not used. Cell array delimiter	num2cell	mat2cell(x,ones(size(x,1),1),size(x,2))
{	abs / norm / determinant	union	or
{			bitor
}	else / finally		
}			split array
~	Not	setxor	xor
~			bitxor