
















Organizing and Accessing Data in MATLAB

This reference shows common use cases, so it is not an exhaustive list.

The [>>](#) icon provides links to relevant sections of the MATLAB documentation.

Representing Data			
Homogenous			
Data type		Purpose	Syntax
Double, single, (u)int8, (u)int16, (u)int32, (u)int64, complex		Numeric arrays, matrix computations, math	<code>[1,2,3], [1;2;3], uint8(), int16()</code>
String		Text arrays	<code>"hello world"</code>
Char		Single characters, character arrays	<code>'hello'</code>
Categorical		Discrete, nonnumeric data	<code>categorical()</code>
Datetime		Absolute dates and timestamps, including time zones	<code>datetime('July 12, 2001 08:15:01')</code>
Duration		Elapsed times	<code>duration(h,m,s), hours(), minutes()</code>
Calendar of duration		Relative time based on calendar	<code>caldays(), calweeks()</code>
Logical		True/false, test state, identify data by condition	<code>logical(), ==, ~=, >, >=, <, <=, &, &&, , </code>
Other specialized types		sparse, enumeration, custom, ...	>>

Heterogeneous			
Data type		Purpose	Syntax
Table		Mixed-type, column-oriented data (spreadsheet-like). Store metadata.	<code>table(x,y,z), array2table</code>
Timetable		Timestamped tabular data	<code>timetable(t,x,y)</code> <code>table2timetable,array2timetable</code>
Structure		Fields can contain data of any size and type. Ideal for nonrectangular data.	<code>struct()</code>
Cell array		Each cell in the array can contain any data type, any size	<code>cell(), {pi,ones(5), "hello"}</code>
Tall array		MATLAB data types can be made "tall" when data does not fit in memory	<code>ds = datastore(), T = tall(ds)</code>
Dictionary		Object that maps unique keys to values	<code>d= dictionary(keys,values)</code>

Data Selection

Use array indexing to select data.

Linear indexing for 1D arrays:

$x(1)$ First element

$x(\text{end})$ Last element



Row, column indexing for multidimensional arrays:

$A(1,2)$

$A(1,1,2)$



Select multiple with a vector:

$A([1,3],1)$



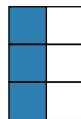
Use colon : to select a range:

$A(1:3,1)$

$A(:,1)$ All rows, column 1

$A(1,:)$ Row 1, all columns

$A(1:2:\text{end},:)$ Every other row



Remove data from array:

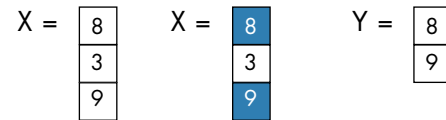
$A(1,:) = [];$

Logical Indexing

Use logical expressions to select data.

Elements of X greater than 7:

$y = x(x > 7)$



Combine conditions:

$x(x > 3 \ \& \ x \leq 7)$

$x(x > 3 \ | \ x \leq 7)$

Elements of S not equal to "hello world":

$S(S \neq \text{"hello world"})$

Multidimensional arrays

Use condition to identify rows or columns:

$\text{idx} = x > 7;$

$x(\text{idx}, :)$

Tables and timetables

$T(:, \text{vartype}(\text{'numeric'}))$

$TT(\text{timerange}(t1, t2), :)$

Container Indexing

Using parentheses () for indexing retains the initial data type. Access the underlying data with curly braces {}. Tables and structures also allow you to reference data by name.

Examples >>

Type	Subset	Contents
Table	Returns a table: $T(1,2)$ $T(:, \text{"A"})$ $T(:, [\text{"A"}, \text{"B"}])$	Returns underlying data: $T\{1,2\}$ $T\{:, \text{"A"}\}$ $T.A$ $T.Rows$ $T.Variables$
Timetable	Same as above: $TT(\text{"Apr 1, ... 2004"}, 5)$	Same as above: $TT.Time$
Cell array	Returns a cell: $C(1,2)$	$C\{1,2\}$ $C\{:\}$ -> comma separated list
Structure	Returns a struct: $S(1,1)$	$S.Field$