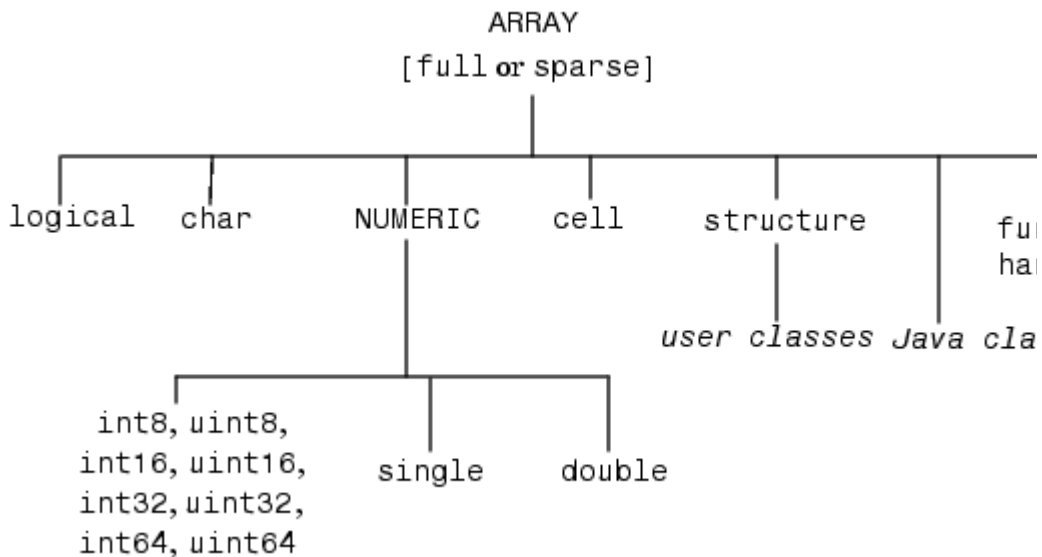


Overview of MATLAB Data Types

There are 15 fundamental data types in MATLAB. Each of these data types is in the form of a matrix or array. This matrix or array is a minimum of 0-by-0 in size and can grow to an n-dimensional array of any size.

All of the fundamental data types are shown in lowercase text in the diagram below. Additional data types are user-defined, object-oriented *user classes* and *Java classes*. You can use the latter with the MATLAB interface to Java (see [Calling Java from MATLAB](#) in the MATLAB External Interfaces documentation).

You can create two-dimensional *double* and *logical* matrices using one of two storage formats: *full* or *sparse*. For matrices with mostly zero-valued elements, a sparse matrix requires a fraction of the storage space required for an equivalent full matrix. Sparse matrices invoke methods especially tailored to solve sparse problems.



The following table describes these data types in more detail.

Data Type	Example	Description
int8 , uint8 , int16 , uint16 , int32 , uint32 , int64 , uint64	uint16(65000)	Array of signed and unsigned integers. Some integer types require less storage space than <i>single</i> or <i>double</i> . All integer types except for <i>int64</i> and <i>uint64</i> can be used in mathematical operations.

single	<code>single(3 * 10^38)</code>	Array of single-precision numbers. Requires less storage space than <code>double</code> , but has less precision and a smaller range.
double	<code>3 * 10^300</code> <code>5 + 6i</code>	Array of double-precision numbers. Two-dimensional arrays can be sparse. The default numeric type in MATLAB.
logical	<code>magic(4) > 10</code>	Array of logical values of 1 or 0 to represent <code>true</code> and <code>false</code> respectively. Two-dimensional arrays can be sparse.
char	<code>'Hello'</code>	Array of characters. Strings are represented as vectors of characters. For arrays containing more than one string, it is best to use cell arrays .
cell array	<code>a{1,1} = 12;</code> <code>a{1,2} =</code> <code>'Red';</code> <code>a{1,3} =</code> <code>magic(4);</code>	Array of indexed cells, each capable of storing an array of a different dimension and data type.
structure	<code>a.day = 12;</code> <code>a.color =</code> <code>'Red';</code> <code>a.mat =</code> <code>magic(3);</code>	Array of C-like structures, each structure having named fields capable of storing an array of a different dimension and data type.
function handle	<code>@sin</code>	Pointer to a function. You can pass function handles to other functions.
user class	<code>polynom([0 -2</code> <code>-5])</code>	Objects constructed from a user-defined class. See MATLAB Classes
Java class	<code>java.awt.Frame</code>	Objects constructed from a Java class. See Java Classes .