

Introduction

What Is MATLAB? (p. 1-2)

See how MATLAB[®] can provide solutions for you in technical computing, what are some of the common applications of MATLAB, and what types of add-on application-specific solutions are available in MATLAB toolboxes.

MATLAB Documentation (p. 1-4)

Find out where to look for instruction on how to use each component of MATLAB, and where to find help when you need it.

Starting and Quitting MATLAB
(p. 1-6)

Start a new MATLAB session, use the desktop environment, and terminate the session.

What Is MATLAB?

MATLAB® is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

The name MATLAB stands for *matrix laboratory*. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called *toolboxes*. Very important to most users of MATLAB, toolboxes allow you to *learn* and *apply* specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

The MATLAB System

The MATLAB system consists of five main parts:

Development Environment. This is the set of tools and facilities that help you use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for viewing help, the workspace, files, and the search path.

The MATLAB Mathematical Function Library. This is a vast collection of computational algorithms ranging from elementary functions, like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

The MATLAB Language. This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both “programming in the small” to rapidly create quick and dirty throw-away programs, and “programming in the large” to create large and complex application programs.

Graphics. MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

The MATLAB External Interfaces/API. This is a library that allows you to write C and Fortran programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

MATLAB Documentation

MATLAB provides extensive documentation, in both printed and online format, to help you learn about and use all of its features. If you are a new user, start with this Getting Started book. It covers all the primary MATLAB features at a high level, including many examples.

The MATLAB online help provides task-oriented and reference information about MATLAB features. MATLAB documentation is also available in printed form and in PDF format.

MATLAB Online Help

To view the online documentation, select **MATLAB Help** from the **Help** menu in MATLAB. The MATLAB documentation is organized into these main topics:

- Desktop Tools and Development Environment — Startup and shutdown, the desktop, and other tools that help you use MATLAB
- Mathematics — Mathematical operations and data analysis
- Programming — The MATLAB language and how to develop MATLAB applications
- Graphics — Tools and techniques for plotting, graph annotation, printing, and programming with Handle Graphics®
- 3-D Visualization — Visualizing surface and volume data, transparency, and viewing and lighting techniques
- Creating Graphical User Interfaces — GUI-building tools and how to write callback functions
- External Interfaces/API — MEX-files, the MATLAB engine, and interfacing to Java, COM, and the serial port

MATLAB also includes reference documentation for all MATLAB functions:

- Functions - By Category — Lists all MATLAB functions grouped into categories
- Handle Graphics Property Browser — Provides easy access to descriptions of graphics object properties
- External Interfaces/API Reference — Covers those functions used by the MATLAB external interfaces, providing information on syntax in the calling language, description, arguments, return values, and examples

The MATLAB online documentation also includes

- Examples — An index of examples included in the documentation
- Release Notes — New features and known problems in the current release
- Printable Documentation — PDF versions of the documentation suitable for printing

For more information about using the Help browser, see Chapter 6, “Desktop Tools and Development Environment.”

Starting and Quitting MATLAB

Starting MATLAB

On Windows platforms, start MATLAB by double-clicking the MATLAB shortcut icon  on your Windows desktop.

On UNIX platforms, start MATLAB by typing `matlab` at the operating system prompt.

You can customize MATLAB startup. For example, you can change the directory in which MATLAB starts or automatically execute MATLAB statements in a script file named `startup.m`.

For More Information See “Starting MATLAB” in the Desktop Tools and Development Environment documentation.

Quitting MATLAB

To end your MATLAB session, select **File -> Exit MATLAB** in the desktop, or type `quit` in the Command Window. You can run a script file named `finish.m` each time MATLAB quits that, for example, executes functions to save the workspace, or displays a quit confirmation dialog box.

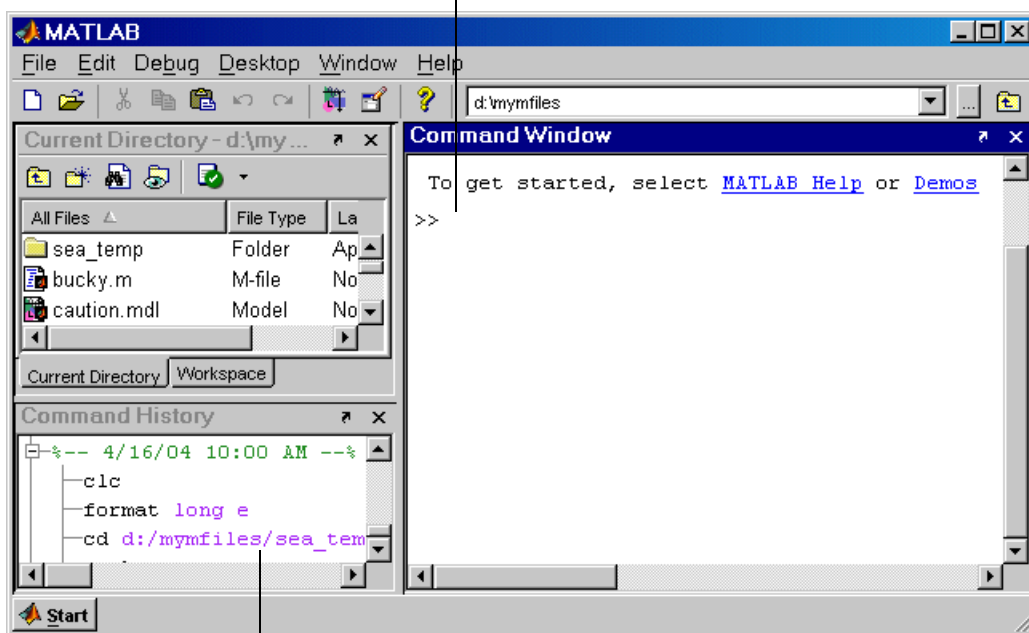
For More Information See “Quitting MATLAB” in the Desktop Tools and Development Environment documentation.

MATLAB Desktop

When you start MATLAB, the MATLAB desktop appears, containing tools (graphical user interfaces) for managing files, variables, and applications associated with MATLAB.

The following illustration shows the default desktop. You can customize the arrangement of tools and documents to suit your needs. For more information about the desktop tools, see Chapter 6, “Desktop Tools and Development Environment.”

Enter MATLAB functions at the Command Window prompt.



The Command History maintains a record of the MATLAB functions you ran.