

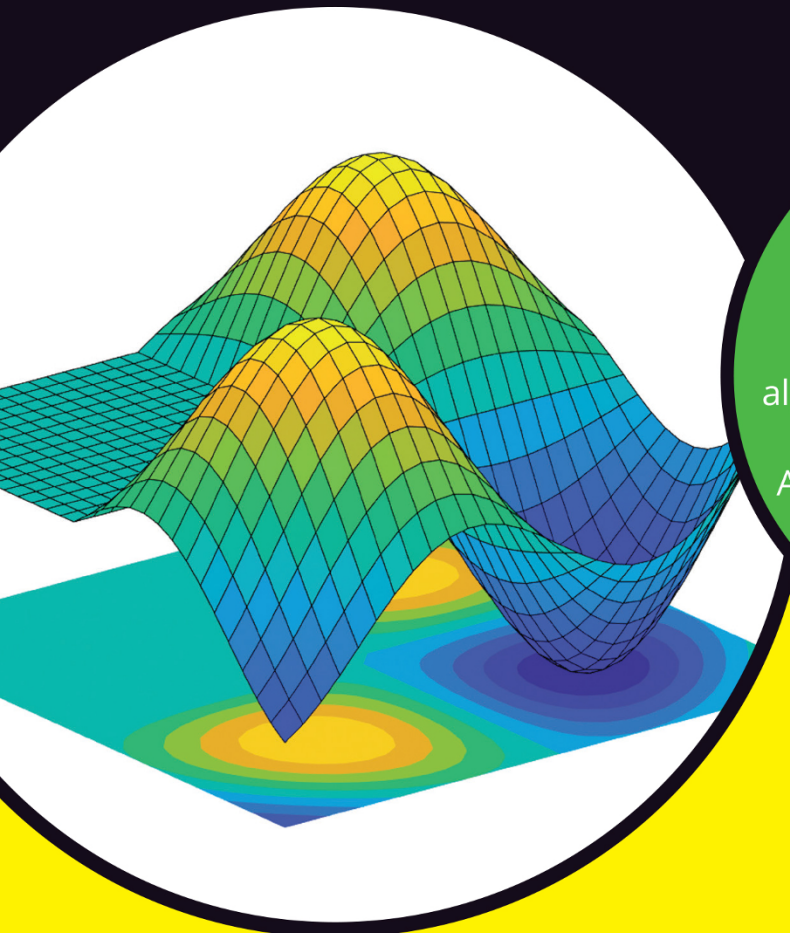
LEARNING MADE EASY



2nd Edition

MATLAB[®]

for
dummies[®]
A Wiley Brand



Plot and manipulate
3D information

Analyze data, develop
algorithms, and create models

Automate your work with Live
Scripts and Live Functions

John Paul Mueller
Author of *Algorithms For Dummies* and
Artificial Intelligence For Dummies

Jim Sizemore, PhD

MATLAB[®]

for
dummies[®]
A Wiley Brand



MATLAB[®]

2nd Edition

**by John Paul Mueller
and Jim Sizemore**

**for
dummies[®]**
A Wiley Brand

MATLAB® For Dummies®, 2nd Edition

Published by: **John Wiley & Sons, Inc.**, 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Copyright © 2021 by John Wiley & Sons, Inc., Hoboken, New Jersey

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, For Dummies, the Dummies Man logo, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and may not be used without written permission. MATLAB is a registered trademark of The MathWorks, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit <https://hub.wiley.com/community/support/dummies>.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2021938618

ISBN: 978-1-119-79688-6

ISBN 978-1-119-79689-3 (ebk); ISBN 978-1-119-79690-9 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Contents at a Glance

Introduction	1
Part 1: Getting Started With MATLAB	7
CHAPTER 1: Introducing MATLAB and Its Many Uses	9
CHAPTER 2: Starting Your Copy of MATLAB	21
CHAPTER 3: Interacting with MATLAB	41
CHAPTER 4: Starting, Storing, and Saving MATLAB Files	61
Part 2: Manipulating and Plotting Data in MATLAB	81
CHAPTER 5: Embracing Vectors, Matrices, and Higher Dimensions	83
CHAPTER 6: Understanding Plotting Basics	121
CHAPTER 7: Using Advanced Plotting Features	139
Part 3: Streamlining MATLAB	163
CHAPTER 8: Automating Your Work	165
CHAPTER 9: Expanding MATLAB's Power with Functions	185
CHAPTER 10: Adding Structure to Your Scripts	207
CHAPTER 11: Working with Live Scripts	229
CHAPTER 12: Working with Live Functions	245
CHAPTER 13: Designing and Using Classes	263
CHAPTER 14: Creating MATLAB Apps	281
CHAPTER 15: Building Projects	301
Part 4: Employing Advanced MATLAB Techniques	315
CHAPTER 16: Importing and Exporting Data	317
CHAPTER 17: Printing and Publishing Your Work	335
CHAPTER 18: Recovering from Mistakes	357
Part 5: Specific MATLAB Applications	379
CHAPTER 19: Solving Equations and Finding Roots	381
CHAPTER 20: Performing Analysis	405

Part 6: The Part of Tens	419
CHAPTER 21: The Top Ten Uses of MATLAB.....	421
CHAPTER 22: Ten Ways to Make a Living Using MATLAB.....	429
Appendix A: MATLAB Functions	437
Appendix B: MATLAB's Plotting Routines	447
Index	453

Table of Contents

INTRODUCTION	1
About This Book	1
Foolish Assumptions	2
Icons Used in This Book	3
Beyond the Book	4
Where to Go from Here	4
PART 1: GETTING STARTED WITH MATLAB	7
CHAPTER 1: Introducing MATLAB and Its Many Uses	9
Putting MATLAB in Its Place	10
Understanding how MATLAB relates to a Turing machine	10
Using MATLAB as more than a calculator	13
Determining why you need MATLAB	14
Discovering Who Uses MATLAB for Real-World Tasks	17
Knowing How to Get the Most from MATLAB	18
Getting the basic computer skills	18
Defining the math requirements	19
Applying what you know about other procedural languages	19
Understanding how this book will help you	20
Getting Over the Learning Curve	20
CHAPTER 2: Starting Your Copy of MATLAB	21
Installing MATLAB	22
Discovering which platforms MATLAB supports	22
Getting your copy of MATLAB	23
Performing the installation	23
Activating the product	24
Meeting the MATLAB Interface	25
Starting MATLAB for the first time	25
Considering the default Toolstrip tabs	27
Working with the Quick Access toolbar (QAT)	29
Employing the Command Window	30
Getting additional help with MATLAB	31
Using the Current Folder toolbar	32
Viewing the Current Folder window	32
Changing the MATLAB layout	38

CHAPTER 3: Interacting with MATLAB	41
Using MATLAB as a Calculator	42
Entering information at the prompt	42
Entering a formula	44
Copying and pasting formulas	44
Changing the Command Window formatting	46
Suppressing Command Window output	47
Understanding the MATLAB Math Syntax	48
Adding, subtracting, multiplying, and dividing	48
Working with exponents	50
Organizing Your Storage Locker	51
Using ans — the default storage locker	52
Creating your own storage lockers	52
Operating MATLAB as More Than a Calculator	54
Learning the truth	54
Using the built-in functions	54
Accessing the function browser	56
Recovering from Mistakes	57
Understanding the MATLAB error messages	58
Stopping MATLAB when it hangs	58
CHAPTER 4: Starting, Storing, and Saving MATLAB Files	61
Examining MATLAB's File Structure	62
Understanding the MATLAB files and what they do	63
Exploring folders with the GUI	65
Exploring folders with commands	69
Working with files in MATLAB	72
Accessing and Sharing MATLAB Files	74
Opening	74
Exporting	75
Importing	76
Saving Your Work	78
Saving variables with the GUI	78
Saving variables using commands	78
Saving commands with the GUI	79
Saving commands using commands	79
Using online storage	80
PART 2: MANIPULATING AND PLOTTING	
DATA IN MATLAB	81
CHAPTER 5: Embracing Vectors, Matrices, and Higher	
 Dimensions	83
Working with Vectors and Matrices	84
Understanding MATLAB's perspective of linear algebra	84
Entering data	86

Adding and Subtracting	92
Understanding the Many Ways to Multiply and Divide	94
Performing scalar multiplication and division.	94
Employing matrix multiplication	95
Dividing two vectors	98
Effecting matrix division	98
Creating powers of matrices	101
Using complex numbers.	102
Working with exponents.	104
Working with Higher Dimensions	105
Creating a multidimensional matrix	105
Accessing a multidimensional matrix.	108
Replacing individual elements.	109
Replacing a range of elements	110
Modifying the matrix size.	112
Using cell arrays and structures	113
Using the Matrix Helps	117
CHAPTER 6: Understanding Plotting Basics	121
Considering Plots.	121
Understanding what you can do with plots.	122
Comparing MATLAB plots to spreadsheet graphs	122
Creating a plot using commands	123
Creating a plot using the Workspace window.	125
Creating a plot using the Plots tab options	126
Using the Plot Function.	128
Working with line color, markers, and line style.	128
Creating multiple plots in a single command	129
Modifying Any Plot	130
Making simple changes	130
Adding to a plot	132
Deleting a plot	134
Working with subplots	134
Plotting with 2D Information	137
CHAPTER 7: Using Advanced Plotting Features.	139
Plotting with 3D Information	140
Using the bar() function to obtain a flat 3D plot.	140
Using bar3() to obtain a dimensional 3D plot	144
Using barh() and more	146
Enhancing Your Plots	147
Getting an axes handle.	147
Modifying axes labels	148
Adding a title	150

Rotating label text	151
Employing annotations.	153
Printing your plot.	154
Using the Plot Extras.	155
Creating axes dates using datetick().	155
Creating plots with colorbar()	157
Interacting with daspect.	159
Interacting with pbaspect.	161
PART 3: STREAMLINING MATLAB	163
CHAPTER 8: Automating Your Work	165
Understanding What Scripts Do	166
Creating less work for yourself	166
Defining when to use a script	167
Creating a Script	168
Writing your first script.	168
Using commands for user input	170
Copying and pasting into a script	171
Converting the Command History into a script	172
Continuing long strings.	172
Adding comments to your script.	174
Revising Scripts	179
Calling Scripts.	179
Improving Script Performance	180
Analyzing Scripts for Errors	181
Using the MATLAB Profiler to Improve Performance	182
CHAPTER 9: Expanding MATLAB's Power with Functions	185
Working with Built-in Functions	186
Learning about built-in functions	186
Sending data in and getting data out	191
Creating a Function.	191
Understanding script and function differences	192
Understanding built-in function and custom function differences	193
Writing your first function	193
Using the new function.	195
Passing data in.	196
Passing data out	198
Creating and using global variables	199
Using subfunctions	202
Nesting functions.	203

Using Other Types of Functions	204
Inline functions	204
Anonymous functions	205
CHAPTER 10: Adding Structure to Your Scripts	207
Making Decisions	208
Using the if statement	208
Using the switch statement	212
Understanding the switch difference	214
Deciding between if and switch	214
Creating Recursive Functions	215
Performing Tasks Repetitively	219
Using the for statement	219
Using the while statement	220
Starting a new loop iteration using continue	221
Ending processing using break	222
Ending processing using return	223
Determining which loop to use	225
Creating Menus	225
CHAPTER 11: Working with Live Scripts	229
Comparing a Live Script to a Regular Script	230
Working with the Live Editor	231
Opening the Live Editor	232
Working with the Output pane	232
Adding formatted text	235
Using plots within the coded area	236
Incorporating graphics	237
Incorporating controls	238
Running Live Script Sections	241
Diagnosing Coding Errors	242
CHAPTER 12: Working with Live Functions	245
Comparing a Live Function to a Regular Function	246
Understanding Live Function Flexibility Differences	247
Creating a Live Function	247
Running a Live Function	248
Refactoring a Live Function	249
Using the specialized coding buttons	253
Going to a specific function	254
Converting a Function to a Live Function	255
Sharing Live Functions and Live Scripts	256
Using an interactive document	256
Employing a full screen presentation	257

Working with plain text	257
Creating a static document	257
Performing Comparisons and Merges	259
Comparing Live Functions and Live Scripts	259
Merging Live Functions and Live Scripts	260
CHAPTER 13: Designing and Using Classes	263
A Brief Overview of Object-Oriented Programming (OOP)	264
Defining an object	264
Considering how properties define an object	265
Using methods to interact with an object	266
Listening to an object using events	267
Understanding the need for privacy	268
Understanding OOP in MATLAB	268
Comparing MATLAB OOP to other languages	269
Uses of classes and objects in MATLAB	270
Performing tasks with objects in MATLAB	271
Creating a Basic MATLAB Class	272
Starting the class	272
Adding properties	275
Specifying methods	278
CHAPTER 14: Creating MATLAB Apps	281
Working with the App Designer	282
Understanding apps	282
Getting apps	283
Starting the App Designer	286
Defining an Interface	290
Understanding the various components	290
Changing the component properties	292
Making the Interface Functional	293
Working with Code View	294
Creating a callback function	295
Running the App	297
Packaging Your App	298
CHAPTER 15: Building Projects	301
Considering the Need for Projects	302
Creating a New Project	303
Choosing a project type	303
Using the Project editor	305
Understanding the Project Dependencies	309
Running Required Checks	311
Checking project integrity	311
Looking for potential updates	311

PART 4: EMPLOYING ADVANCED MATLAB TECHNIQUES	315
CHAPTER 16: Importing and Exporting Data	317
Importing Data.	318
Avoiding older import/export function calls	319
Performing import basics.	319
Importing mixed strings and numbers	325
Importing selected rows or columns	327
Exporting Data.	328
Performing export basics.	328
Exporting scripts and functions.	331
Working with Images	332
Exporting images.	332
Importing images	333
CHAPTER 17: Printing and Publishing Your Work	335
Using Commands to Format Text	336
Modifying font appearance	336
Using special characters.	344
Adding math symbols.	346
Publishing Your MATLAB Data.	350
Performing advanced script and function publishing tasks	350
Saving your figures to disk.	354
Printing Your Work	355
Configuring the output page	355
Printing the data	356
CHAPTER 18: Recovering from Mistakes	357
Working with Error Messages	358
Responding to error messages	359
Understanding the MException class	361
Creating error and warning messages	362
Setting warning message modes	365
Understanding Quick Alerts.	366
Relying on Common Fixes for MATLAB's Error Messages	368
Making Your Own Error Messages	369
Developing the custom error message	370
Creating useful error messages	374
Using Good Coding Practices.	375

PART 5: SPECIFIC MATLAB APPLICATIONS	379
CHAPTER 19: Solving Equations and Finding Roots	381
Working with the Symbolic Math Toolbox	382
Obtaining your copy of the Symbolic Math Toolbox	382
Installing the Symbolic Math Toolbox	384
Working with the GUI	386
Typing a simple command in the Command Window	387
Performing Algebraic Tasks	388
Differentiating between numeric and symbolic algebra	389
Solving quadratic equations	390
Working with cubic and other nonlinear equations	392
Understanding interpolation	393
Working with Statistics	395
Understanding descriptive statistics	395
Understanding robust statistics	398
Employing the Symbolic Math Toolbox for plotting	401
CHAPTER 20: Performing Analysis	405
Using Linear Algebra	406
Working with determinants	406
Performing reduction	407
Using eigenvalues	409
Understanding factorization	409
Employing Calculus	411
Working with differential calculus	411
Using integral calculus	413
Working with multivariate calculus	413
Solving Differential Equations	415
Using the numerical approach	415
Using the symbolic approach	416
PART 6: THE PART OF TENS	419
CHAPTER 21: The Top Ten Uses of MATLAB	421
Working with Linear Algebra	422
Performing Numerical Analysis	422
Designing a Neural Network Simulation	423
Getting Involved in Science	423
Logging Sensor Data	424
Exploring Research	424
Creating Light Animations Using Arduino	425
Employing Image Processing	425
Controlling Industrial Equipment	426
Performing Audio Compression Using Wavelets	427

CHAPTER 22: Ten Ways to Make a Living Using MATLAB	429
Working with Green Technology	430
Creating Speech Recognition Software	430
Performing Antenna Analysis and Design	431
Getting Disease under Control	432
Becoming a Computer Chip Designer	432
Working with Robots	433
Keeping the Trucks Rolling	433
Designing Equipment Used in the Field	434
Reducing Risks Using Simulation	434
Creating Security Solutions	435
 APPENDIX A: MATLAB FUNCTIONS	 437
 APPENDIX B: MATLAB'S PLOTTING ROUTINES	 447
 INDEX	 453

Introduction

MATLAB is an amazing product that helps you perform math-related tasks of all sorts using the same techniques that you'd use if you were performing the task manually (using pencil and paper, slide rule, or abacus if necessary, but more commonly using a calculator). However, MATLAB makes it possible to perform these tasks at a speed that only a computer can provide. In addition, using MATLAB reduces errors, streamlines many tasks, and makes you more efficient.

More important, MATLAB makes sharing your efforts with others incredibly easy. You can use Live Scripts to create report-like output that management can understand, or to develop apps for coworkers to employ when performing their tasks.

MATLAB is also a big product with numerous tools and features that you might never have used in the past. For example, instead of simply working with numbers, you have the ability to plot them in a variety of ways that help you communicate the significance of your data to other people. To get the most from MATLAB, you really need a book like *MATLAB For Dummies*, 2nd Edition.

About This Book

The main purpose of *MATLAB For Dummies*, 2nd Edition is to reduce the learning curve that comes with using a product that offers as much as MATLAB does. When you first start MATLAB, you might become instantly overwhelmed by everything you see. This book helps you get past that stage and become productive quickly so that you can get back to performing amazing feats of math wizardry.

In addition, this book introduces you to techniques that you might not know about or even consider because you haven't been exposed to them before. For example, MATLAB provides a rich plotting environment that helps you not only communicate better but also present numeric information in a manner that helps others see your perspective.

Using scripts and functions will also reduce the amount of work you have to do. This book shows you how to create custom code, which you can use to customize the environment to meet your specific needs. This edition introduces you to Live

Scripts and Live Functions, which enable you to combine code and output into a single report-like version that everyone can use, even if they don't necessarily understand the math. Using classes helps you package your code to make it easier to reuse and understand. If you want to create a form of your code that is accessible to coworkers and people who may not want to know *why* something works, just that it does, you can also discover apps and toolboxes.

After you've successfully installed MATLAB on whatever computer platform you're using, you start with the basics and work your way up. By the time you finish working through the examples in this book, you'll be able to perform a range of simple tasks in MATLAB that includes writing scripts, writing functions, creating plots, and performing advanced equation solving. No, you won't be an expert, but you will be able to use MATLAB to meet specific needs in the job environment.

To make absorbing the concepts even easier, this book uses the following conventions:

- » Text that you're meant to type just as it appears in the book is **bold**. The exception is when you're working through a step list: Because each step is bold, the text to type is not bold.
- » When you see words in *italics* as part of a typing sequence, you need to replace that value with something that works for you. For example, if you see "Type **Your Name** and press Enter," you need to replace *Your Name* with your actual name.
- » Web addresses and programming code appear in monospace. If you're reading a digital version of this book on a device connected to the Internet, note that you can click the web address to visit that website, like this: `https://www.dummies.com`.
- » When you need to type command sequences, you see them separated by a special arrow, like this: File ⇨ New File. In this case, you go to the File menu first and then select the New File entry on that menu. The result is that you see a new file created.

Foolish Assumptions

You might find it difficult to believe that we've assumed anything about you — after all, we haven't even met you yet! Although most assumptions are indeed foolish, we made these assumptions to provide a starting point for the book.

Being familiar with the operating system platform you want to use is important because the book doesn't provide any guidance in this regard. (Chapter 2 does provide MATLAB installation instructions.) You really do need to know how to install applications, use applications, and generally work with your chosen platform before you begin working with this book.

This book isn't a math primer. Yes, you see lots of examples of complex math, but the emphasis is on helping you use MATLAB to perform math tasks rather than learn math theory. Chapter 1 helps you understand precisely what you need to know from a math perspective in order to use this book successfully.

This book also assumes that you can access items on the Internet. Sprinkled throughout are numerous references to online material that aren't mandatory but can enhance your learning experience. However, these added sources are useful only if you actually find and use them.

Icons Used in This Book

As you read this book, you see icons in the margins that indicate material of interest (or not, as the case may be). This section briefly describes each icon in this book.



TIP

Tips are nice because they help you save time or perform some task without a lot of extra work. The tips in this book are time-saving techniques or pointers to resources that you should try in order to get the maximum benefit from MATLAB.



WARNING

We don't want to sound like angry parents or some kind of maniacs, but you should avoid doing anything that's marked with a Warning icon. Otherwise, you might find that your application fails to work as expected, you get incorrect answers from seemingly bulletproof equations, or (in the worst-case scenario) you lose data.



TECHNICAL
STUFF

Whenever you see this icon, think advanced tip or technique. You might find these tidbits of useful information just too boring for words, or they could contain the solution you need to get a program running. Skip these bits of information whenever you like.



REMEMBER

If you don't get anything else out of a particular chapter or section, remember the material marked by this icon. This text usually contains an essential process or a bit of information that you must know to work with MATLAB successfully.

Beyond the Book

This book isn't the end of your MATLAB experience — it's really just the beginning. We provide online content to make this book more flexible and better able to meet your needs. That way, as we receive email from you, we can address questions and tell you how updates to either MATLAB or its associated add-ons affect book content. In fact, you gain access to all these cool additions:

- » **Cheat sheet:** You remember using crib notes in school to make a better mark on a test, don't you? You do? Well, a cheat sheet is sort of like that. It provides you with some special notes about tasks that you can do with MATLAB that not every other person knows. You can find the Cheat Sheet for this book at www.dummies.com and entering **MATLAB For Dummies, 2nd Edition** in the Search field. Click Cheat Sheets in the row of options under the book title.
- » **Errata:** You can find errata by going to www.dummies.com/go/matlabfd2e. Scroll under the image of the book cover to find the Errata link, if there is one. In addition to errata, check out the blog posts with answers to reader questions and demonstrations of useful book-related techniques at <http://blog.johnmuelเลอร์books.com/>.
- » **Companion files:** Hey! Who really wants to type all the code in the book and reconstruct all those plots by hand? Most readers would prefer to spend their time actually working with MATLAB and seeing the interesting things it can do, rather than typing. Fortunately for you, the examples used in the book are available for download. You can find these files by going to www.dummies.com/go/matlabfd2e. Scroll under the image of the book cover to find the Downloads link. Alternatively, you can obtain the source code at John Mueller Books Writing with Style (<http://www.johnmuelเลอร์books.com/source-code/>). Just locate the book's name and click the Download button.

Where to Go from Here

It's time to start your MATLAB adventure! If you're completely new to MATLAB, you should start with Chapter 1 and progress through the book at a pace that allows you to absorb as much of the material as possible.

If you're a novice who's in an absolute rush to get going with MATLAB as quickly as possible, you could skip to Chapter 2 with the understanding that you may find some topics a bit confusing later. Skipping to Chapter 3 is possible if you already have MATLAB installed, but be sure to at least skim Chapter 2 so that you know what assumptions we made writing this book.

Readers who have some exposure to MATLAB can save reading time by moving directly to Chapter 5. You can always go back to earlier chapters as necessary when you have questions. However, it's important that you understand how each technique works before moving to the next one. Every technique, coding example, and procedure has important lessons for you, and you could miss vital content if you start skipping too much information.

1

Getting Started With MATLAB

IN THIS PART . . .

Considering what MATLAB can do for you

Getting MATLAB installed and ready for use

Working with the MATLAB interface

Interacting with the MATLAB files

IN THIS CHAPTER

- » Understanding how MATLAB fits in as a tool for performing math tasks
- » Seeing where MATLAB is used today
- » Discovering how to get the most from MATLAB
- » Overcoming the MATLAB learning curve

Chapter **1**

Introducing MATLAB and Its Many Uses

Math is the basis of all our science and even some of our art. In fact, math itself can be an art form — consider the beauty of fractals (a visual presentation of a specialized equation). However, math is also abstract and can be quite difficult and complex to understand and to use for practical purposes. MATLAB makes performing math-related tasks easier. You use MATLAB to perform math-related tasks such as

- » Numerical computation
- » Visualization
- » Scripting
- » Application development
- » Machine learning
- » Deep learning
- » Signal processing
- » Other tasks allowed by its various toolboxes (see <https://www.mathworks.com/help/thingspeak/matlab-toolbox-access.html> for details)

This chapter introduces you to MATLAB, an application that performs a variety of math tasks. It helps you understand the role that MATLAB can play in reducing the overall complexity of math and in explaining math-related information to others more easily. You also discover that many organizations and major developers use MATLAB to perform real-world tasks in a manner that improves accuracy, efficiency, and consistency. (A partial list of such tasks appears at <https://stackshare.io/matlab>.) Of course, knowing how you can translate these benefits of MATLAB to your own workplace is important.



REMEMBER

Because MATLAB can do so much, it does have a learning curve. This chapter also discusses what you can do to reduce the learning curve so that you become productive much faster. The less time you spend learning about MATLAB, the more time you spend applying math to your particular specialty, and the better the results you achieve. Getting things done quickly and accurately is the overall goal of MATLAB.

Putting MATLAB in Its Place

MATLAB is all about math. Yes, it's a powerful tool and yes, it includes its own language to make the execution of math-related tasks faster, easier, and more consistent. However, when you get right down to it, the focus of MATLAB is the math. For example, you could type $2 + 2$ as an equation, and MATLAB would dutifully report the sum of 4 as output. Of course, no one would buy an application to compute $2 + 2$ — you could easily do that with a calculator. So you need to understand just what MATLAB can do. The following sections help you put MATLAB into perspective so that you can better understand how you can use it to perform complex math tasks.

Understanding how MATLAB relates to a Turing machine

Today's computers are mostly Turing machines, named after the British mathematician Alan Turing (1912–1954). The main emphasis of a Turing machine is performing tasks step by step. A single processor performs one step at a time. It may work on multiple tasks, but only a single step of a specific task is performed at any given time. Knowing about the Turing machine orientation of computers is important because MATLAB follows precisely the same strategy. It, too, performs tasks one step at a time in a procedural fashion. In fact, you can download an application that simulates a Turing machine using MATLAB at <https://www.mathworks.com/matlabcentral/fileexchange/23006-turing-machine-emulator>. The code is surprisingly short. (Note that the actual design of a computer relies on

principles defined by John von Neumann; see <https://cacm.acm.org/magazines/2020/1/241712-von-neumann-thought-turings-universal-machine-was-simple-and-neat/fulltext> for details.)

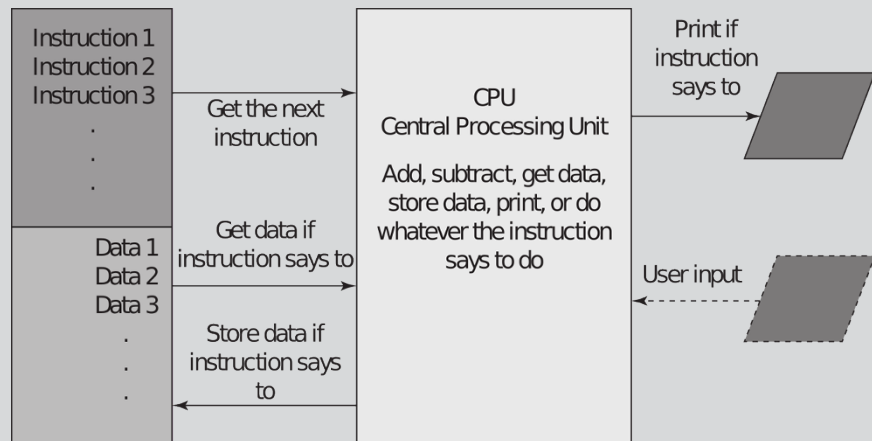
UNDERSTANDING HOW COMPUTERS WORK

Many older programmers are geeks who punched cards before TVs had transistors. One advantage of punching cards is getting to physically touch and feel the computer's instructions and data. This physicality gave programmers a good understanding of what happens when a program runs.

Today, the instructions and data are stored as charges of electrons in tiny pieces of silicon too small to be seen through even the most powerful optical microscope. Today's computers can handle much more information much more quickly than early machines. But the way they use that information is basically the same as early computers.

In those old card decks, programmers wrote one instruction on each card. After completing all the instructions, they put the data cards into a card reader. The computer read a card, and the computer did what the card told it to do: Get some data, get more data, add it together, divide, and so on until all the instructions were executed.

A series of instructions is a program. The following figure shows a basic schematic block diagram of how a computer works.



(continued)

(continued)

Unchanged from the old days, when cards were read one at a time, computer instructions continue to be read one at a time at a low level (think machine code). The instruction is executed, and then the computer goes to the next instruction. MATLAB executes programs in this manner as well.

Don't confuse the one-instruction-after-another approach with parallel programming. A parallel program relies on individual processors working in tandem to make application execution faster, but the instructions still execute one at a time on each processor. You can employ parallel programming in MATLAB using the Parallel Computing Toolbox, described at <https://www.mathworks.com/products/parallel-computing.html>.

It's important to realize that the *flow* of a program can change. Computers can make decisions based on a specific criterion (based on one comparison even when an expression contains multiple comparisons), such as whether something is true or false, and take the route indicated for that decision. For example, when the computer has read all the data for a task, the program tells the computer to quit reading data and start doing calculations. One way to map how the computer executes programs is called a *flow chart*, which is similar to a road map with intersections where decisions must be made. MATLAB relies on well-designed flow charts to make it easy to see what the computer will do, when it will do it, and how it will accomplish the required tasks.

The whole concept of a program may seem foreign to many — something that only geeks would ever love — but you've already used the concept of a program before. When using a calculator, you first think of the steps and numbers you want to enter and in what sequence to enter them to solve your problem. A program, including a MATLAB program, is simply a sequence of similar steps stored in a file that the computer reads and executes one at a time. You don't need to fear computer programming — you've probably done something very similar quite often and can easily do it again.



REMEMBER

Don't confuse the underlying computer, which relies on machine code, with the high-level programming languages used to create applications for it. Even though the programs that drive the computer may be designed to give the illusion of some other technique, called a *programming paradigm*, when you look at how the computer works at a low level, you see that it goes step by step. (In fact, some languages support multiple programming paradigms, as described at <https://blog.newrelic.com/engineering/python-programming-styles/>.) If you've never learned how computers run programs, this information serves as meaningful background. Refer to the nearby sidebar "Understanding how computers work" for a discussion of this important background information.