

**MATLAB I/O and External Interfaces**

**Course Number:** MTLB-110
**Duration:** 1 day

**Overview**

This advanced-level MATLAB I/O and External Interfaces training course teaches attendees how to connect MATLAB to a wide variety of external connectors, including COM, Java, C++, .NET, Python, website URLs, disk, I/O, and more.

**Note:** This course can be condensed to a half-day with fewer examples and less hands-on practice. We strongly recommend the full-day version if possible.

**Prerequisites**

Attendees should have taken Accelebrate's [Introduction to MATLAB course](file:////training/matlab-scripts-programs) or have equivalent knowledge. Students must be comfortable using the MATLAB environment and have some basic MATLAB programming experience.

**Materials**

All training students will receive comprehensive courseware.

**Software Needed on Each Student PC**

* Any Windows, Linux, or macOS operating system
* A recent version of MATLAB

**Objectives**

* Access file data programmatically in MATLAB
* Access external Java/C++/.NET/COM functionality within MATLAB
* Use the custom programmatic styling of MS Office documents in MATLAB
* Manipulate external objects via MATLAB code
* Use MATLAB’s new C++ and Python interfaces

**Outline**

* Introduction
* MATLAB File I/O
	+ High/medium/low level I/O
	+ Binary vs. text format
	+ Performance considerations and other tradeoffs
* Connecting MATLAB to a URL
	+ The urlread and webread functions, and their relatives
	+ Customizing the underlying network object
	+ Using JSON data format
* Connecting MATLAB to a COM Server
	+ The actxserver function and its relatives
	+ Manipulating Microsoft Office documents (Excel, PowerPoint, Word)
	+ Enumerations
	+ Avoiding COM access pitfalls
* Using .NET Objects in MATLAB
	+ The import command
	+ Implicit data-type conversion
	+ Calling .NET functions and processing the returned results
* Using Java Objects in MATLAB
	+ The javaaddpath function and its relatives
	+ Using Java objects within MATLAB
	+ Data type conversions
	+ Java-MATLAB arrays
	+ Static vs. dynamic classpath
	+ Attaching MATLAB callbacks to Java events (if time permits)
	+ Avoiding Java access pitfalls
* Using Python in MATLAB
	+ In-process vs. external Python engine
	+ Calling Python functions and processing the returned results
* Synchronicity
	+ MATLAB’s Main Thread and Java/Python/.NET function calls
	+ Implications of argument data being passed by value
	+ Using Java/.NET/Python threads
* Using C++ Code in MATLAB
	+ MEX
	+ MATLAB’s changed memory access
	+ Avoiding MEX access pitfalls
	+ The loadlibrary function and its relatives
	+ The new import functionality
* Conclusion