

MATLAB Fundamentals

Objective:

The course in 40 hours provides a gentle introduction to the MATLAB computing environment, and is intended for beginning users and those looking for a review. It is designed to give students a basic understanding of MATLAB, including popular toolboxes. The course consists of interactive lectures and sample MATLAB problems given as assignments and discussed in class. No prior programming experience or knowledge of MATLAB is assumed. Concepts covered include basic use, graphical representations and tips for designing and implementing MATLAB code.

The main objectives are:

- Understanding the MATLAB environment
- Being able to do simple calculations using MATLAB
- Being able to carry out simple numerical computations and analyses using MATLAB

Prerequisites:

Undergraduate-level mathematics and experience with basic computer operations

Overview of Training:

This three-day course provides a comprehensive introduction to the MATLAB[®] technical computing environment. No prior programming experience or knowledge of MATLAB is assumed. Themes of data analysis, visualization, modeling, and programming are explored throughout the course. Topics include:

- Working with the MATLAB user interface
- Entering commands and creating variables

- Analyzing vectors and matrices
- Visualizing vector and matrix data
- Working with data files
- Working with data types
- Automating commands with scripts
- Writing programs with branching and loops
- Writing functions

Benefits of Learning MATLAB:

- It enables you to solve many numerical problems in a fraction of the time it would take you to write a program in a lower level language. MATLAB helps you better understand and apply concepts in applications ranging from engineering and mathematics to chemistry, biology, and economics.

Learning Outcome:

Upon successful completion of this course, the student should be able to:

- Understand the main features of the MATLAB development environment
- Use the MATLAB GUI effectively
- Design simple algorithms to solve problems
- Write simple programs in MATLAB to solve scientific and mathematical problems
- Know where to find help

Target Audience:

- The course is open to all B.Tech and Science graduates (BSC / MSC) with an interest in numerical computing and numerical simulations
- Students who has basic knowledge of computer programming and an understanding of matrix and linear algebra will need MATLAB to prepare their final thesis work

Course Outline:

<p>Working with the MATLAB User Interface</p>	<p>Objective: Become familiar with the main features of the MATLAB integrated design environment and its user interfaces. Get an overview of course themes.</p> <p>Reading data from files</p> <p>Saving and loading variables</p> <p>Plotting data</p> <p>Customizing plots</p> <p>Exporting graphics for use in other applications</p>	<p>4 hours</p>
<p>Variables and Commands</p>	<p>Objective: Enter MATLAB commands, with an emphasis on creating variables, accessing and manipulating data in variables, and creating basic visualizations. Collect MATLAB commands into scripts for ease of reproduction and experimentation.</p> <p>Entering commands</p> <p>Creating numeric and character variables</p> <p>Making and annotating plots</p> <p>Getting help</p>	<p>5 hours</p>

	Creating and running live scripts	
Analysis and Visualization with Vectors	<p>Objective: Perform mathematical and statistical calculations with vectors. Use MATLAB syntax to perform calculations on whole data sets with a single command. Organize scripts into logical sections for development, maintenance, and publishing.</p> <p>Performing calculations with vectors</p> <p>Accessing and modifying values in vectors</p> <p>Formatting and sharing live scripts</p>	3 hours
Analysis and Visualization with Matrices	<p>Objective: Use matrices as mathematical objects or as collections of (vector) data. Understand the appropriate use of MATLAB syntax to distinguish between these applications.</p> <p>Creating and manipulating matrices</p> <p>Performing calculations with matrices</p> <p>Calculating statistics with matrix data</p> <p>Visualizing matrix data</p>	4 hours
Tables of Data	<p>Objective: Import data as a MATLAB table. Work with data stored as a table.</p> <p>Storing data as a table</p> <p>Operating on tables</p> <p>Extracting data from tables</p> <p>Modifying tables</p>	4 hours

<p>Conditional Data Selection</p>	<p>Objective: Extract and analyze subsets of data that satisfy given criteria.</p> <p>Logical operations and variables Finding and counting Logical indexing</p>	<p>4 hours</p>
<p>Organizing Data</p>	<p>Objective: Organize table data for analysis. Represent data using appropriate native MATLAB data types.</p> <p>Combining tables of data Table metadata Dates and durations Discrete categories</p>	<p>3 hours</p>
<p>Analyzing Data</p>	<p>Objective: Perform typical data analysis tasks in MATLAB, including importing data from files, preprocessing data, fitting a model to data, and creating a customized visualization of the model.</p> <p>Importing from spreadsheets and delimited text files Dealing with missing data Plotting functions Customizing plots</p>	<p>5 hours</p>
<p>Increasing Automation with Programming Constructs</p>	<p>Objective: Create flexible code that can interact with the user, make decisions, and adapt to different situations.</p> <p>Programming constructs User interaction Decision branching</p>	<p>4 hours</p>

	Loops	
Increasing Automation with Functions	<p>Objective: Increase automation by encapsulating modular tasks as user-defined functions. Understand how MATLAB resolves references to files and variables. Use MATLAB development tools to find and correct problems with code.</p> <p>Creating functions</p> <p>Calling functions</p> <p>Setting the MATLAB path</p> <p>Debugging</p> <p>Using breakpoints</p> <p>Creating and using structures</p>	4 hours