

# University of Mohammed Essedik Benyehia Jijel Faculty of technology 2nd year Engineer



Info 3

## Introduction to MATLAB®

M. Bouzenita

### Outline

| Lecture 1: MATLAB Programming Environment                | (1 week)  |
|--|-----------|
| Lecture 2: MATLAB data types and script file             | (2 weeks) |
| Lecture 3: Read, display and save data in MATLAB         | (2 weeks) |
| Lecture 4: Vectors and matrices in MATLAB                | (2 weeks) |
| Lecture 5: if and switch statements, for and while loops | (2 weeks) |
| Lecture 6: The function file in MATLAB                   | (2 weeks) |
| Lecture 7 : Graphics                                     | (2 weeks) |
| Lecture 8: Using Toolbox                                 | (2 weeks) |

## Info 3 Introduction to MATLAB®

#### M. Bouzenita

2nd year Engineer - University of Jijel

#### Lecture 1

## Presentation of computing programming environment MATLAB

#### 1. Introduction

Nowadays, using programming language to solve scientific problems is indispensable for engineers and those working in many fields such as medicine, biology, finance and others.



## 2. Computing programming environment



A "computing programming environment" can be defined as software and tools used to write, test, and debug computer programs.

## 3. MATLAB Programming Environment

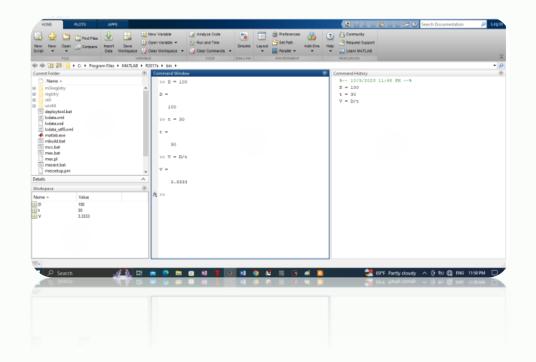
MATLAB (MATrix LABoratory) is an **interactive** computing environment developed by MathWorks



Cleve Moler in 1970 and was first released as a commercial product in 1984 at the Automatic Control Conference in Las Vegas

## 3. MATLAB Programming Environment

MATLAB provides a technical computing environment designed to enable numerical computation and data visualization.



## 3. MATLAB Programming Environment

#### MATLAB has:

**Toolbox** allowing access to symbolic computing abilities



Simulink adds graphical multi domain simulation for dynamic and embedded systems

# 4. Alternatives numerically oriented programming language



#### **Commercials Alternatives**

AMESim, GAUSS, IDL, Maple, Mathcad, Mathematica, OxMetrics, PyIMSL Studio, SAS/IML, Stata (Mata), Sysquake

4. Alternatives numerically oriented programming language

#### **Open alternatives**

Octave, Scilab, FreeMat,

JMathLib, R, SageMath, SciPy



#### 5. MATLAB online version

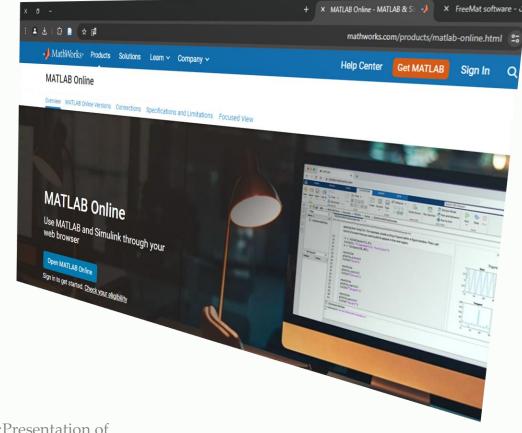
Mathworks provides

MATLAB online version

for use anywhere through

web browser without any

downloads and installation



#### 5. MATLAB online version

Two versions of access are provided:

**Basic version** 

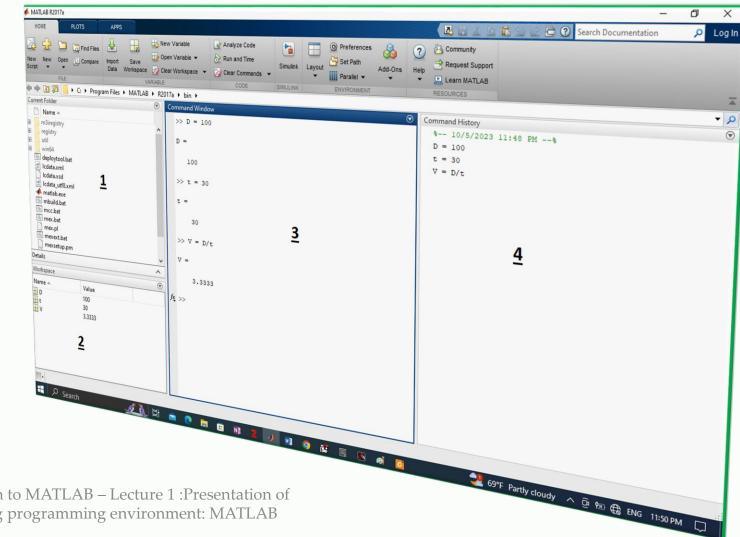
offers 20 hours of use/month and access to 10 commonly used products.

Licensed version

provides a **full access** to MATLAB online in addition to desktop version

## 6. Getting started with MATLAB

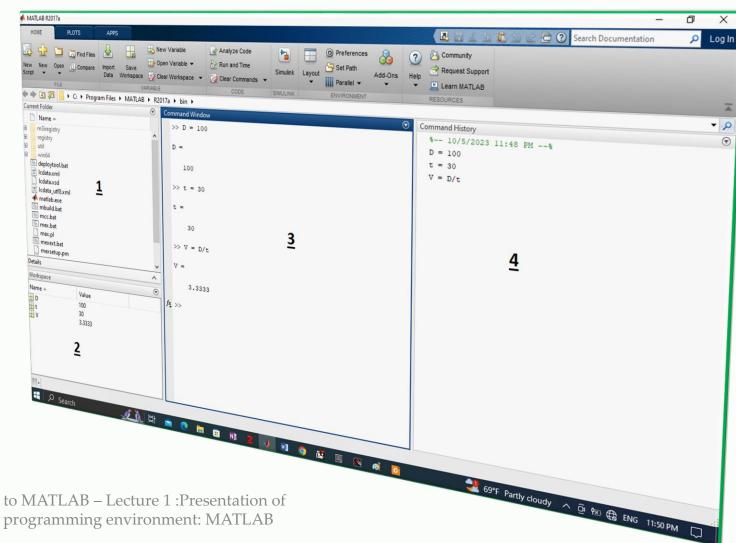
- 1. Current folder (File browser): Displays current, the saved and the performed working directory
- 2. Workspace: Refers to the environment where create, store, and manage variables and data during MATLAB session.



Introduction to MATLAB – Lecture 1: Presentation of computing programming environment: MATLAB

## 6. Getting started with MATLAB

- 3. Command window: is where we can enter commands and see their results. It is the primary interface for interacting with MATLAB
- 4. Command history : allows to view and access the history of commands we have entered during our current session



Introduction to MATLAB – Lecture 1: Presentation of computing programming environment: MATLAB

#### 7. Basic arithmetic

MATLAB uses the standard arirhmetic operators:

```
addition (+),
subtraction (-),
multiplication (*),
division (/ or \)
power (^)
```

```
>> 5 + 3
ans =
     8
ans =
     * 3
ans =
     15
>> 5 / 3
ans =
     1.6667
ans =
      125
```

### 8. Priority of commands

The arithmetic operators have the following priority:

- 1. Brackets first;
- 2. Power next;
- 3. Multiplication and division next. Left to right in case of competition;
- 4. Addition and subtraction next.

  Left to right in case of competition.

## 8. Priority of commands

$$2 + 2 / 2 * 3 ^ 2 = 11$$

$$(2 + 2 / 2 * 3) ^ 2 = 25$$

$$(2 + 2 / 2) * 3 ^ 2 = 27$$

#### 9. Variables

Variables in MATLAB are defined by an **identifier name**, which must respect the following rules:

- 1. The variable name must begin with a letter and can contain letters, digits and underscore character (\_);
- 2. MATLAB is case sensitive. So, variable named **Var\_1** is different from variable named **var 1**;
- 3. MATLAB has **reserved keywords**, which cannot be used as an identifier name (**sqrt**, **exp**, ...)

#### 9. Variables

Variables in MATLAB amust respect the following

- 1. The variable name **1** letters, digits and un
- 2. MATLAB is case sens from variable named
- 3. MATLAB has **reserv** identifier name (**sqr**

>> 5\*var

```
>> A = 5
A =
>> Var 1 = A +1
Var 1 =
>> 5var
 5var
Invalid expression. Check for missing multiplication
operator, missing or unbalanced delimiters, or other
syntax error. To construct matrices, use brackets
instead of parentheses.
 Did you mean:
```

#### 9. Variables

Variables in MATLAB amust respect the following

- 1. The variable name **1** letters, digits and un
- 2. MATLAB is case sens from variable named
- 3. MATLAB has **reserv** identifier name ( **sqr**

```
>> var_1
Unrecognized function or variable 'var_1'.
Did you mean:
```

>> Var 1

 $Var_1 =$ 

>> A-1

ans =

## 10. Some reserved keywords and commands

MATLAB has a list of reserved words and commands. Some of such keywords are :

Functions: sin, cos, sqrt, exp, ...

Constants: pi, i, j, inf ...

Commands: who, whos, ...

## 11. Reusing previous commands

To rerun previous commands do one of the following:

- Press the up arrow key (↑) until the command you want appears at the prompt, and then press Enter
- Double-click an entry or entries in the Command History window, or select an entry and press Enter.
- To extend the selection to include multiple commands, press Shift+↑.

# 12. Clearing screen and variables and quitting MATLAB

To **remove items** from workspace **and freeing up system memory**, the following commands are used:

```
clc : clear command window.
clear : removes all variables from the workspace.
clear VARIABLES : does the same thing as the previous command.
clear GLOBAL : removes all global variables.
clear all : removes all variables
clear VAR1 VAR2 : clears the specified variables.
```

To close MATLAB using the command window we type quit or exit.

#### **Practice**