

# COMPUTER APPLICATIONS

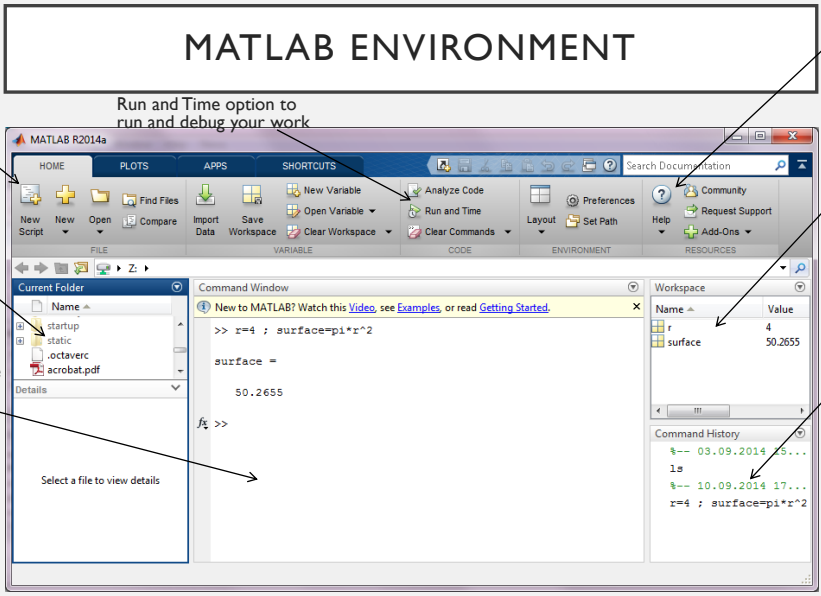
## An Introduction to MatLab

Lecturer Ahmed Wael

## OVERVIEW

- MATLAB is a high performance language for technical computing.
- MATLAB is also a modern programming language environment.
- The origin of the word “MATLAB” is: “MATrix LABoratory”

# MATLAB ENVIRONMENT



New Script to open the "EDITOR" window

Current Folder sub window lists the files in your directory

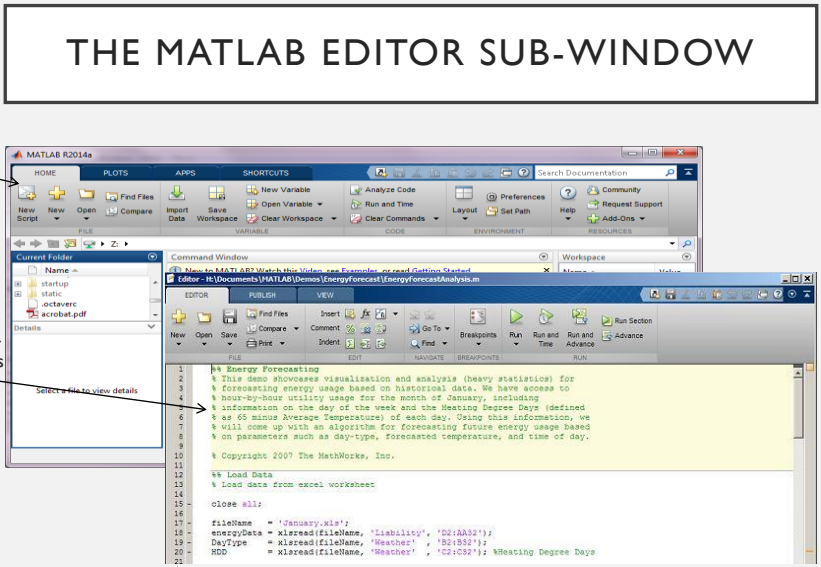
Command Window is when you type commands and see results of your work

Help window used to provide more information

Workspace lists all variables the user created in the Command Window

Command History lists recent variables and functions user is created to make easy back of them

# THE MATLAB EDITOR SUB-WINDOW



(1) New Script will open Script Editor

The editor window used for writing and editing programs

## MATLAB HELP OPTION

- To view the online documentation, select MATLAB Help from help menu or write MATLAB Help directly in the command window.
- To know a specific or full information about a command in MATLAB, in command window type the following:
 

```
>> help command %It search exacts function name.
```
- Example
 

```
>> help sqrt      % It gives you details about the squared root function
```

## VARIABLES AND BASIC ARITHMETIC

- The syntax of MATLAB variables assignment is:
 

```
>> variable Name = a value or an expression
```
- Example
 

```
>> x = 2;          % Defining a variable x with value of 2.
>> y = 'a';       % Defining a variable y equal to string.
>> z = x+3;       % Defining a variable z equal to expression.
```
- Variable names MUST start with a letter and can be followed by letters, digits and “\_”

## VARIABLES AND BASIC ARITHMETIC

- The usual arithmetic operations available in MATLAB are as follow:

Addition	+
Subtraction	-
Multiplication	*
Division	/
exponentiation	^

## VARIABLES AND BASIC ARITHMETIC

- Example: In the MATLAB command window write a program to run the following equation for  $y = 2$  and  $z = 8$ .

$$x = 3yz^3 + \sqrt{\frac{y}{z}}$$

```
>> y = 2;
>> z = 8;
>> x = (3 * y * z^3) + (y / z)^0.5;
```

- There is a priority in arithmetic operations. MATLAB perform parentheses first, division and multiplication and finally addition and subtraction

## NUMBER FORMATS

- To control how a number is displayed, you use the “**format**” command: changing the format has no effect on MATLAB’s internal calculations. The format list are as follow:

command	output	
<code>format short</code>	44.8219	this is the default
<code>format short e</code>	4.4822e+01	that is, $4.4833 \times 10^1$ , note rounding
<code>format long</code>	44.821869662029940	16 places (double precision)
<code>format long e</code>	4.482186966202994e+01	
<code>format bank</code>	44.82	as if it were money
<code>format rat</code>	14343/320	a rational approximation

## COMMANDS FOR MANAGING THE VARIABLES

- To eliminate variables or to obtain information about variables can be obtained by entering the command in the COMMAND WINDOW.
- The list of such commands are listed below:

Command	Description
<code>clear</code>	Removes all variables from the memory.
<code>clear x, y, z</code>	Clears/removes only variables <b>x</b> , <b>y</b> and <b>z</b> from the memory.
<code>who</code>	Lists the variables currently in the workspace.
<code>whos</code>	Displays a list of the variables currently in the memory and their size together with information about their bytes and class.

## BUILT-IN FUNCTIONS OF MATLAB

- MATLAB offers many predefined mathematical functions for technical computing which contains a large set of mathematical functions.
- Typing `>> help elfun` and `>> help specfun` calls up full lists of elementary and special function respectively.

<code>cos(x)</code>	Cosine	<code>abs(x)</code>	Absolute value
<code>sin(x)</code>	Sine	<code>sign(x)</code>	Signum function
<code>tan(x)</code>	Tangent	<code>max(x)</code>	Maximum value
<code>acos(x)</code>	Arc cosine	<code>min(x)</code>	Minimum value
<code>asin(x)</code>	Arc sine	<code>ceil(x)</code>	Round towards $+\infty$
<code>atan(x)</code>	Arc tangent	<code>floor(x)</code>	Round towards $-\infty$
<code>exp(x)</code>	Exponential	<code>round(x)</code>	Round to nearest integer
<code>sqrt(x)</code>	Square root	<code>rem(x)</code>	Remainder after division
<code>log(x)</code>	Natural logarithm	<code>angle(x)</code>	Phase angle
<code>log10(x)</code>	Common logarithm	<code>conj(x)</code>	Complex conjugate

## EXAMPLE

```
>> a = 5; x = 2; y = 8;
>> y = exp(-a)*sin(x)+10*sqrt(y)
y =
    28.2904
```

## VECTORS AND MATRIX

- An array is a list of numbers arranged in row and/or columns.
- A one dimensional array is a row or a columns of numbers and a two dimensional array has a set of numbers arranged in rows and columns.
- In MATLAB, the ROW vectors can be written as:

```
>> x = [ 1 2 3 4 5 6]
```

```
x =
```

```
1 2 3 4 5 6
```

```
>> x = [1,2,3]
```

```
x =
```

```
1
```

```
2
```

```
3
```

## COLON AND Linspace

- If the entries of a vector are an arithmetic sequence, you can use COLON OPERATOR or Linspace command to build the vector.
- Example: a vector (r) can be defined as follow:

```
>> r = [1 2 3 4];           % one row vector of elements from 1 to 4
```

Or 

```
>> r = [1:4];           % one row vector of elements from 1 to 4
```

Or 

```
>> r = linspace(1,4,4); % one row vector of elements from 1 to 4
```

## EXAMPLE

- Define the vector ( $r$ ) starts from 0 to 20 by step of 2.

```
>> r = [0:2:20];
```

- What do you think the output could be!!!!

```
r =  
0 2 4 6 8 10 12 14 16 18 20
```

## ANOTHER EXAMPLE

- What will be the output of this program:

```
>> a = [pi : pi/2 : 4*pi];  
>> m = 3*cos(a);
```

- The output of this program is:

```
m =  
0 3 0 3 ...
```