

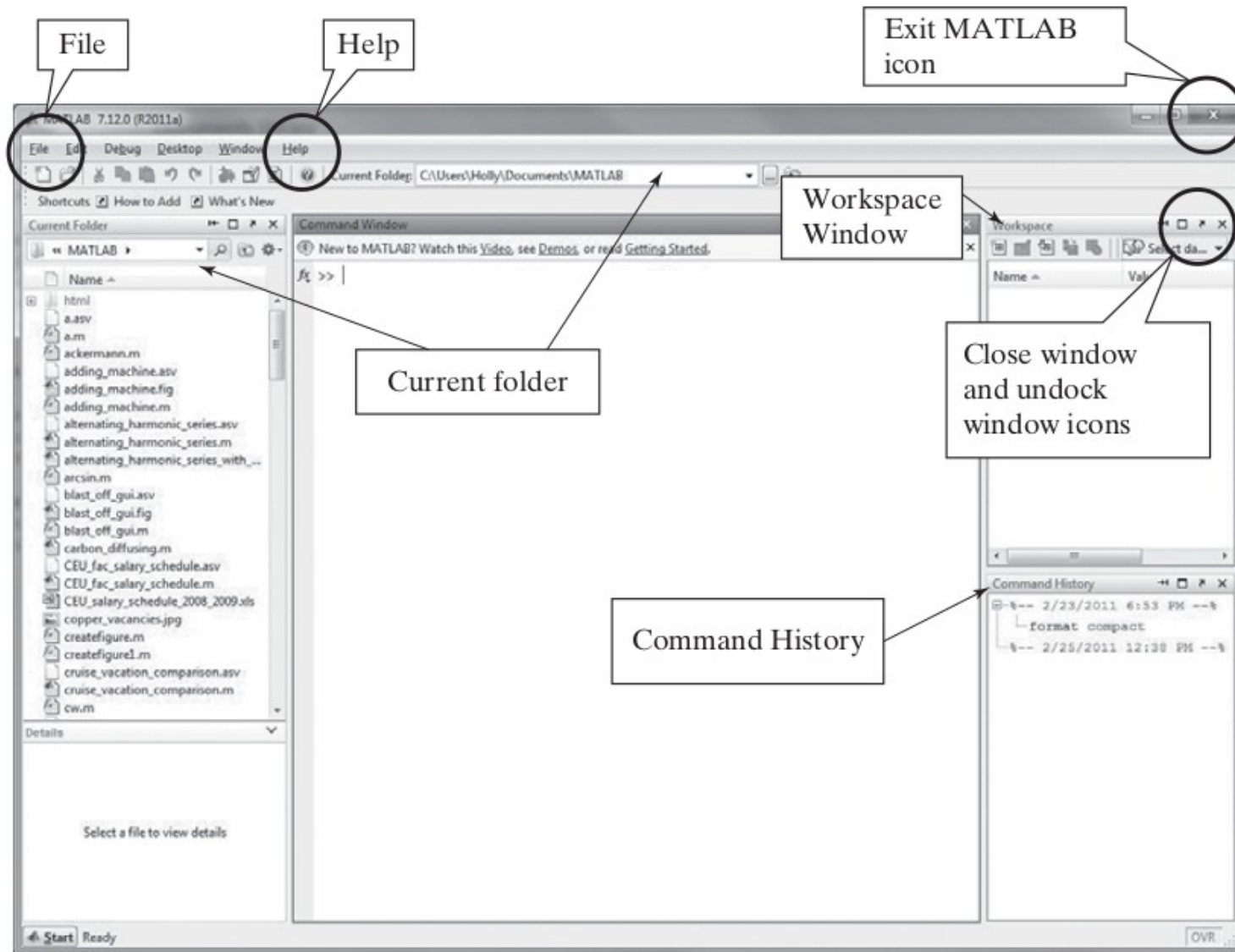
Basic MATLAB Operations



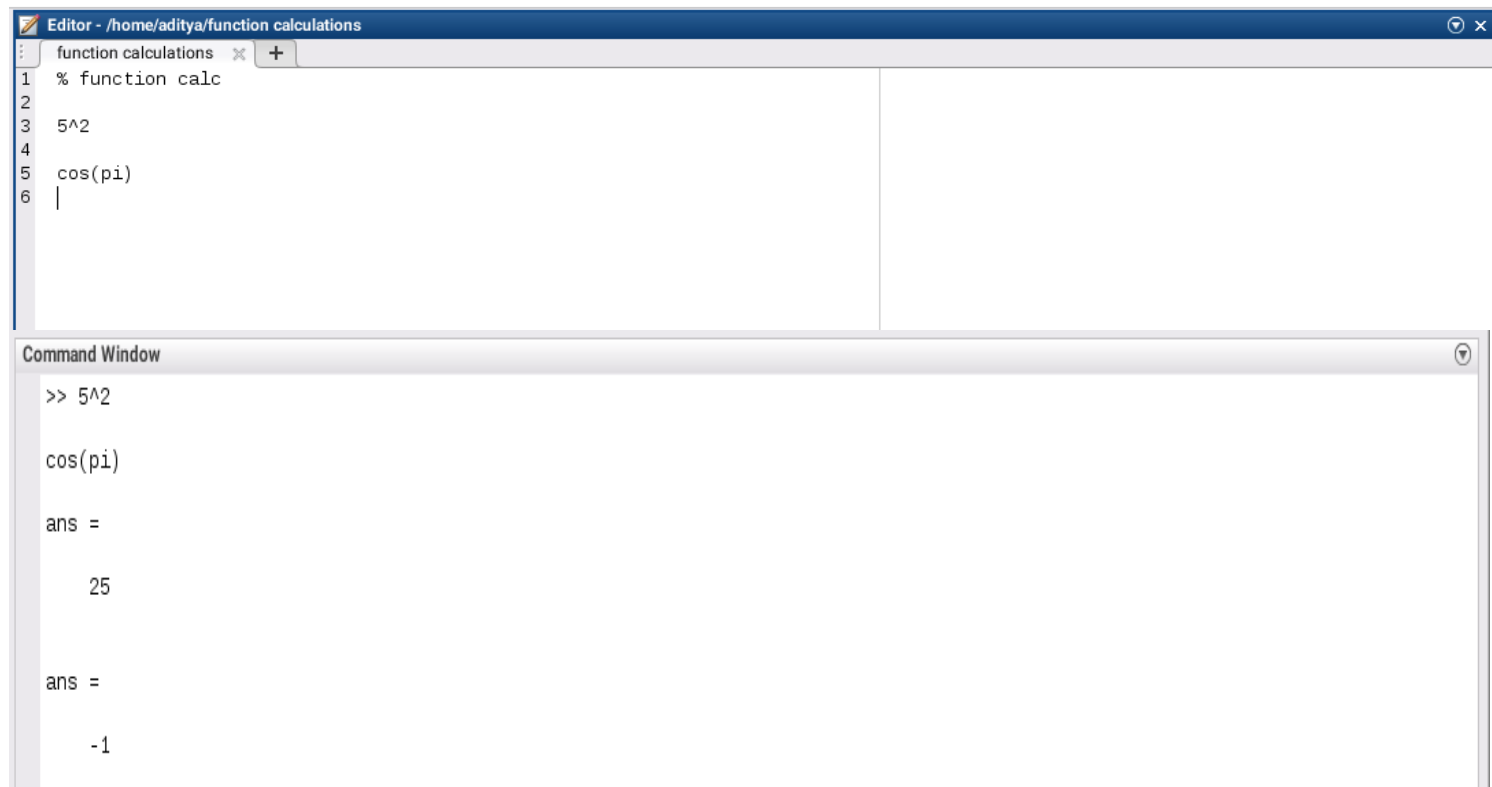
References

- MATLAB for Engineers by Holly Moore 5th Edition

MATLAB opening window



Basic calculations



The image shows a MATLAB environment with an Editor window and a Command Window. The Editor window, titled 'Editor - /home/aditya/function calculations', contains a script with the following code:

```
1 % function calc
2
3 5^2
4
5 cos(pi)
6 |
```

The Command Window, titled 'Command Window', shows the execution of the code from the script:

```
>> 5^2

cos(pi)

ans =

    25

ans =

   -1
```

The Command Window output shows the results of the calculations: 25 for 5^2 and -1 for $\cos(\pi)$.

How to create an array of elements?

- Explicit method

```
B = [1, 2, 3, 4]
```

```
B =  
    1    2    3    4
```

```
B = [1 2 3 4]
```

```
B =  
    1    2    3    4
```

```
C = [1 2 3 4; 10 20 30 40; 5 10 15 20]
```

```
C =  
    1    2    3    4  
   10   20   30   40  
    5   10   15   20
```

- Implicit method:

Ex: $b = 1:5$ or $b = [1:5]$

Above command returns a row matrix

$b = 1\ 2\ 3\ 4\ 5$ (Default increment is 1)

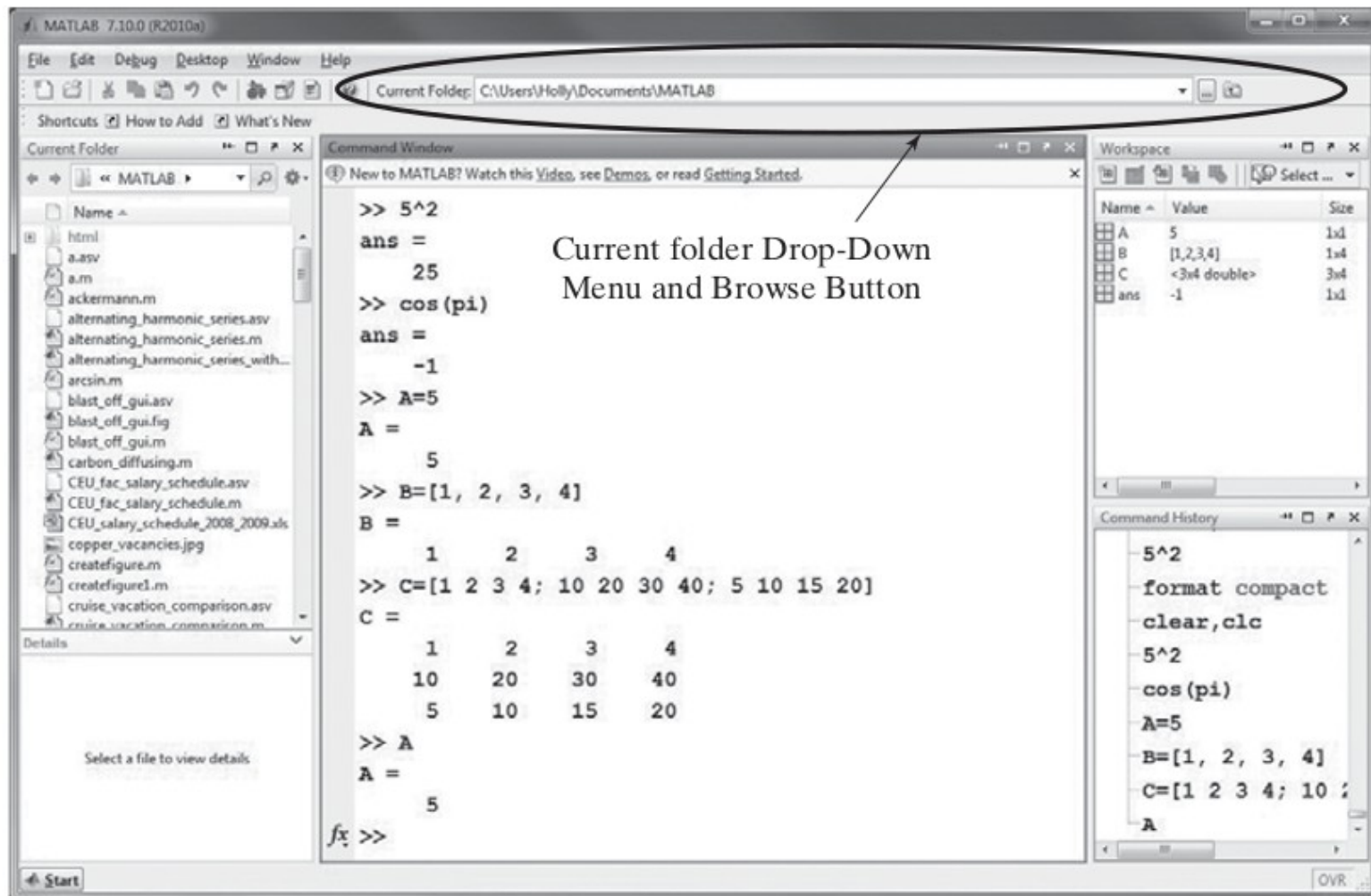
- Ex: $c = 1:2:5$

Above command returns a row matrix

$c = 1\ 3\ 5$

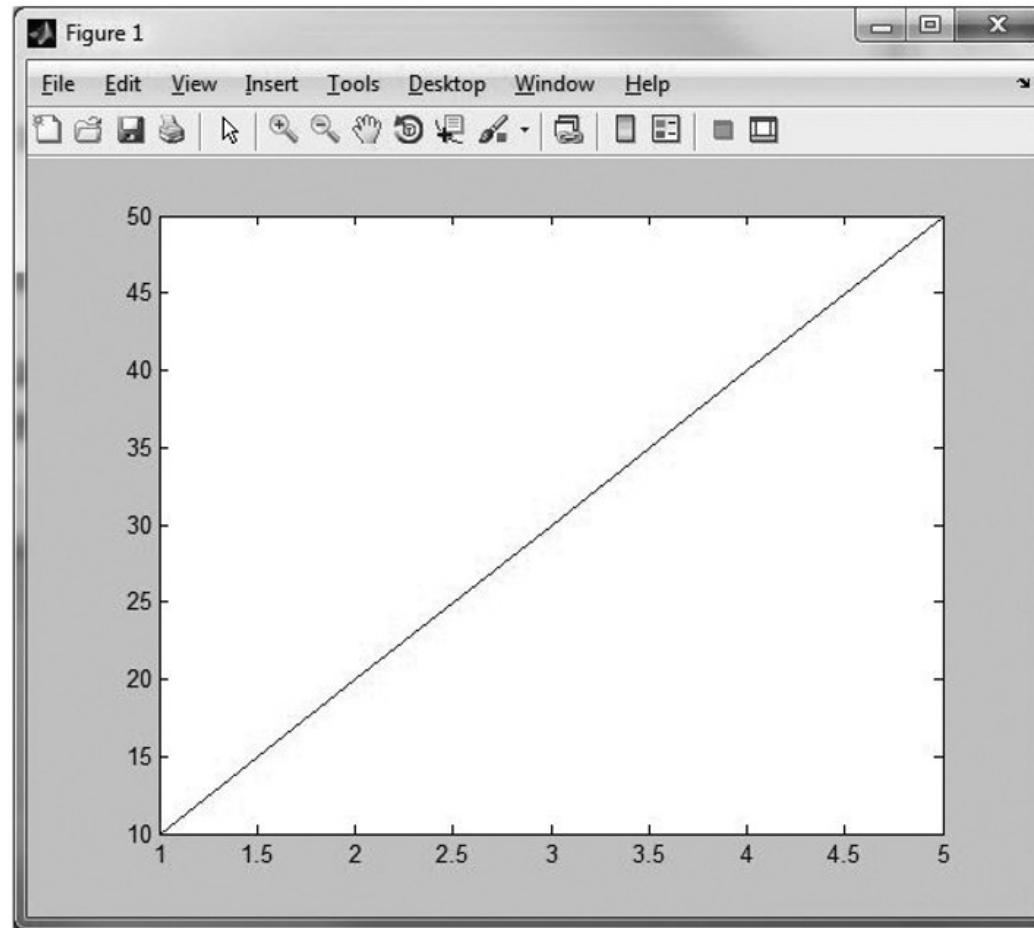
(Indicates the increment between the values 1 & 5 by a factor of 2)

Arrays



Plot Function

- $X=[1,2,3,4,5]$; $y=[10,20,30,40,50]$;
- To create a graph, use the plot command: `plot(x,y)`



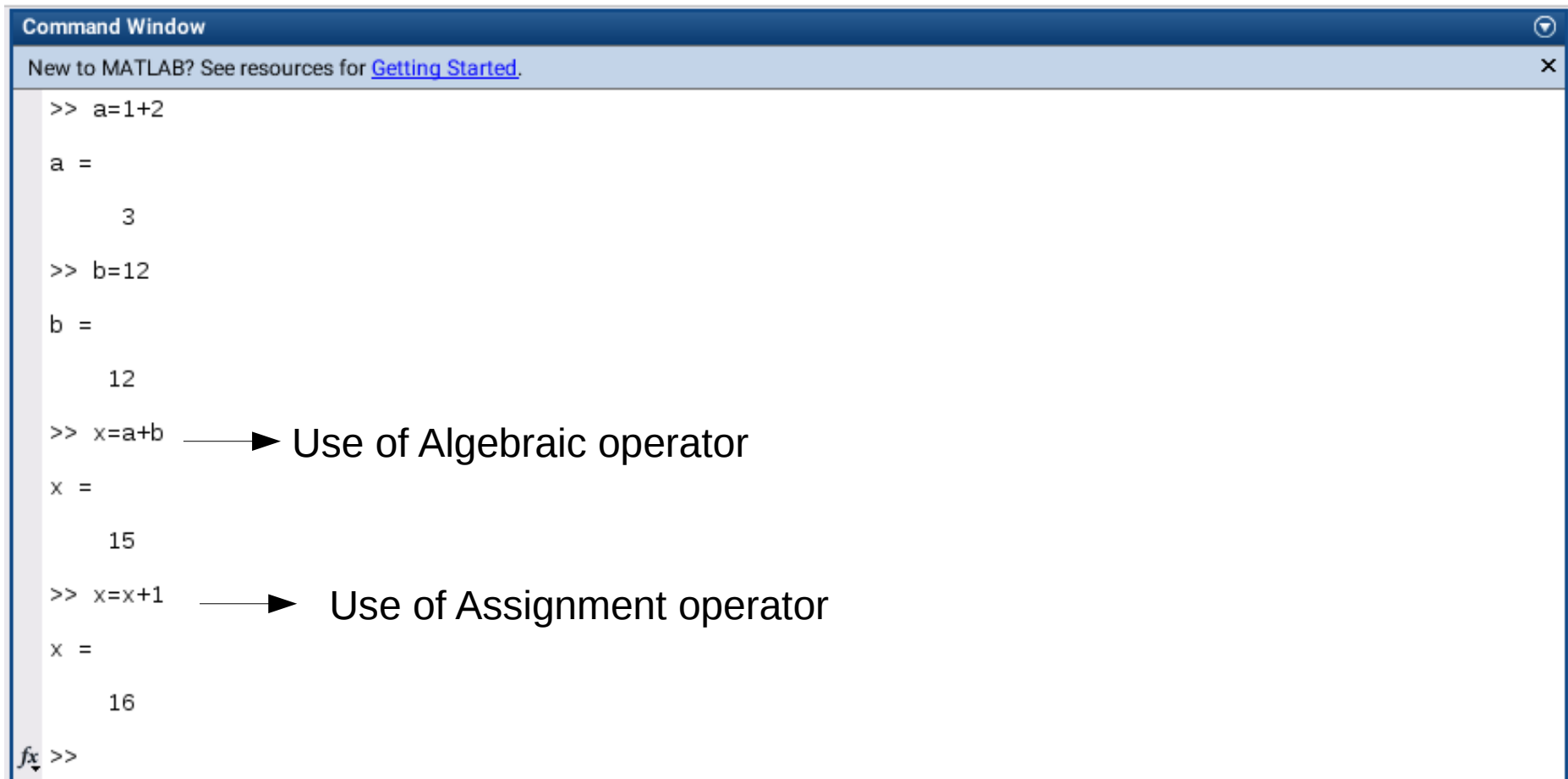
Matrices in MATLAB

$$A = [5] \quad B = [2 \ 5] \quad C = \begin{bmatrix} 1 & 2 \\ 5 & 7 \end{bmatrix}$$

Arithmetic operations between two scalars

Operation	Algebraic Syntax	MATLAB [®] Syntax
Addition	$a + b$	a + b
Subtraction	$a - b$	a - b
Multiplication	$a \times b$	a * b
Division	$\frac{a}{b}$ or $a \div b$	a / b
Exponentiation	a^b	a ^ b

Scalar operations



The image shows a screenshot of the MATLAB Command Window. The window has a title bar 'Command Window' and a subtitle bar 'New to MATLAB? See resources for [Getting Started.](#)'. The main area contains the following text:

```
>> a=1+2  
  
a =  
  
     3  
  
>> b=12  
  
b =  
  
    12  
  
>> x=a+b —————> Use of Algebraic operator  
  
x =  
  
    15  
  
>> x=x+1 —————> Use of Assignment operator  
  
x =  
  
    16  
  
fx >>
```

The text 'fx >>' is located at the bottom left of the Command Window, below the last line of code.

Order of operations

In all mathematical calculations, it is important to understand the order in which operations are performed. MATLAB[®] follows the standard algebraic rules for the order of operation:

- First perform calculations inside parentheses, working from the innermost set to the outermost.
- Next, perform exponentiation operations.
- Then perform multiplication and division operations, working from left to right.
- Finally, perform addition and subtraction operations, working from left to right.

Array operations

- **linspace** command: spacing between elements

Syntax: `d = linspace(1,10,3)`

Where 1 = initial value

10 = Final value

3 = Total number of values

Output : `d = 1 5.5 10`

Matrix multiplication

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
>> a=[1 2 3]
```

```
a =
```

```
     1     2     3
```

```
>> b=[4 5 6]
```

```
b =
```

```
     4     5     6
```

```
>> c=a*transpose(b)  —▶ Matrix multiplication
```

```
c =
```

```
    32
```

```
>> c=a.*b  —▶ Element by Element multiplication
```

```
c =
```

```
     4    10    18
```

```
>> d=a./b  —▶ Element by Element division
```

```
d =
```

```
    0.2500    0.4000    0.5000
```

```
>> e=a.^b  —▶ Exponentiation
```

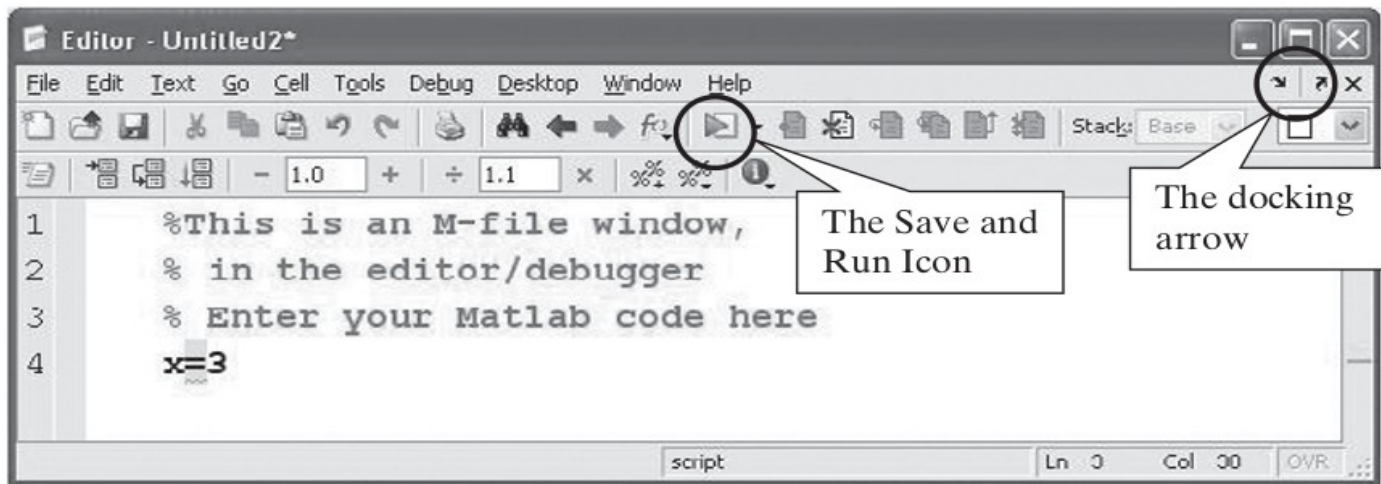
```
e =
```

```
     1    32   729
```

```
fx >>
```

Script M-files

- An M-file is an ASCII text file which can be edited and created with MATLAB Editor. To open the editing window, choose (File → New → Script)



Approaches to Executing a Script M-File from the Command Window

MATLAB® Command	Comments
<code>myscript</code>	Type the file name, for example myscript . The .m file extension is assumed.
<code>run myscript</code>	Use the run command with the file name.
<code>run('myscript')</code>	Use the functional form of the run command.

Built-In functions

abs(x)	Finds the absolute value of x .	abs(-3) ans = 3
sqrt(x)	Finds the square root of x .	sqrt(85) ans = 9.2195
nthroot(x,n)	Finds the real <i>n</i> th root of x . This function will not return complex results. Thus, $(-2)^{(1/3)}$ does not return the same result, yet both answers are legitimate third roots of -2.	nthroot(-2, 3) ans = -1.2599 (-2)^(1/3) ans = 0.6300 + 1.0911i
sign(x)	Returns a value of -1 if x is less than zero, a value of 0 if x equals zero, and a value of +1 if x is greater than zero.	sign(-8) ans = -1
rem(x,y)	Computes the remainder of x/y .	rem(25,4) ans = 1
exp(x)	Computes the value of e^x , where <i>e</i> is the base for natural logarithms, or approximately 2.7183.	exp(10) ans = 2.2026e + 004
log(x)	Computes $\ln(x)$, the natural logarithm of x (to the base <i>e</i>).	log(10) ans = 2.3026
log10(x)	Computes $\log_{10}(x)$, the common logarithm of x (to the base 10).	log10(10) ans = 1

Example:

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
>> x=[4 9 16];  
>> b=sqrt(x)
```

b =

```
     2     3     4
```

```
>> d=[1 2 3;4 5 6]
```

d =

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

```
>> f=size(d)
```

f =

```
     2     3
```

fx >>

Roots function:

```
Command Window
New to MATLAB? See resources for Getting Started.

>> % Roots of the equation x^3 -6*(x^2)+11*x-6
>> k=[1 -6 11 -6];
>> x=roots(k)

x =

    3.0000
    2.0000
    1.0000

fx >>
```