

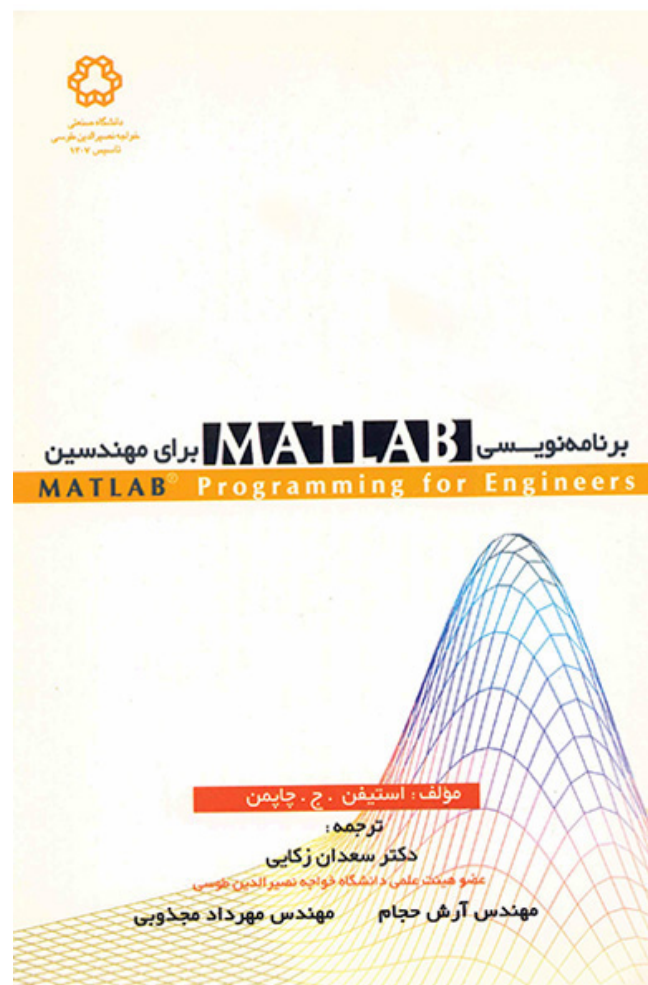
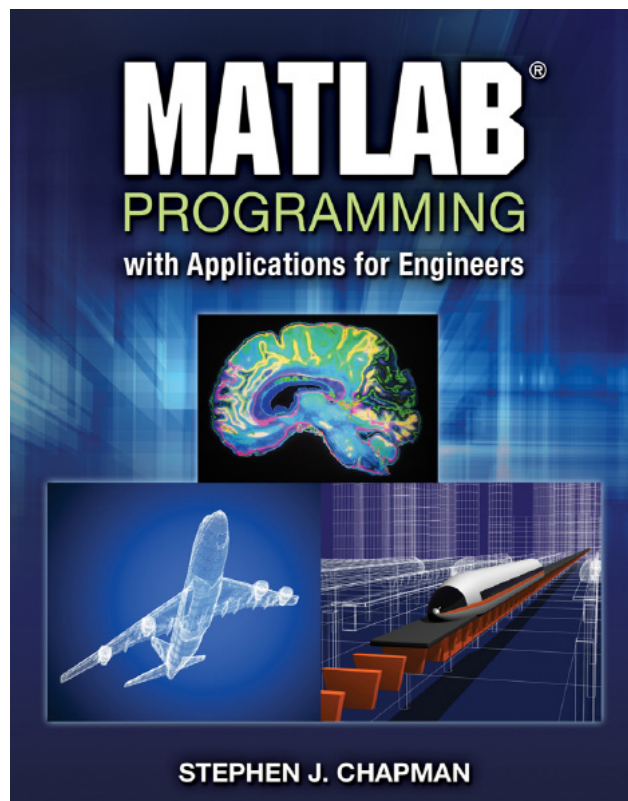
مقدمه ای بر برنامه نویسی با MATLAB

سعید سهیلی
استادیار گروه مکانیک
دانشگاه آزاد اسلامی مشهد

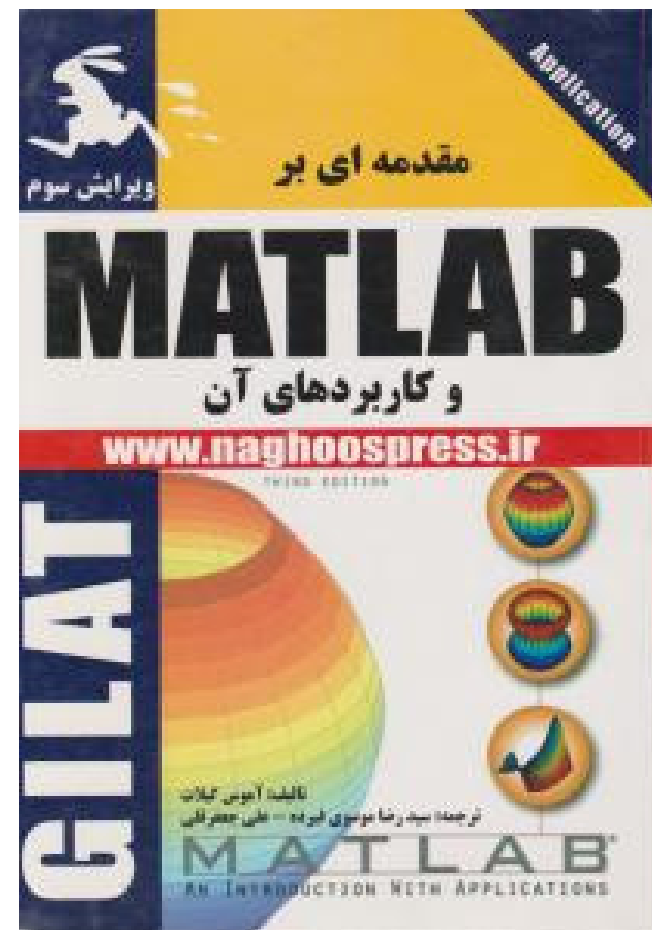
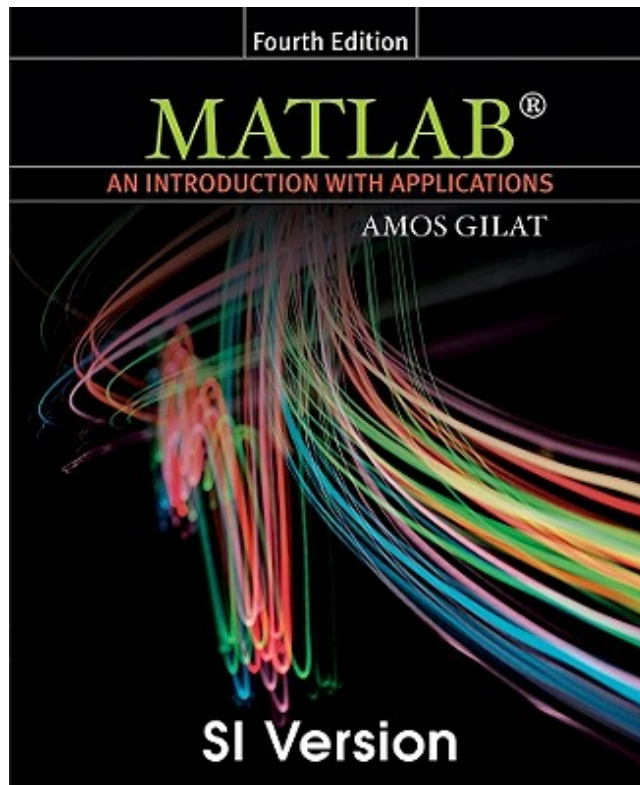
بسمہ اربعہ

مراجع

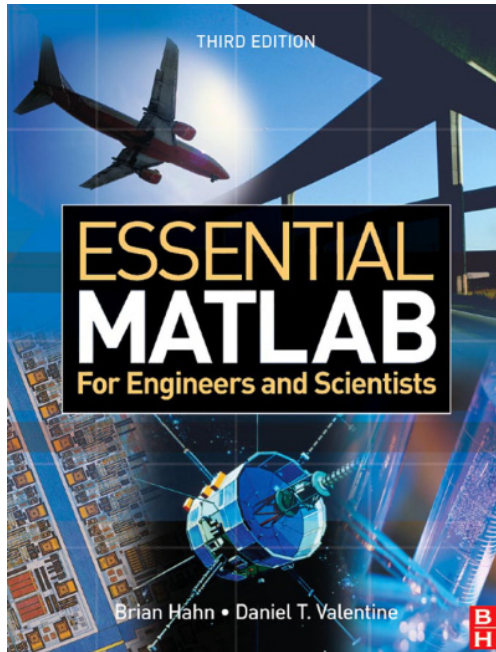
- مرجع اصلی درس



• مرجع کمکی



• سایر مراجع



Advanced
Mathematics
and Mechanics
Applications Using

MATLAB®

Third Edition

Howard B. Wilson
University of Alabama

Louis H. Turcotte
Rose-Hulman Institute of Technology

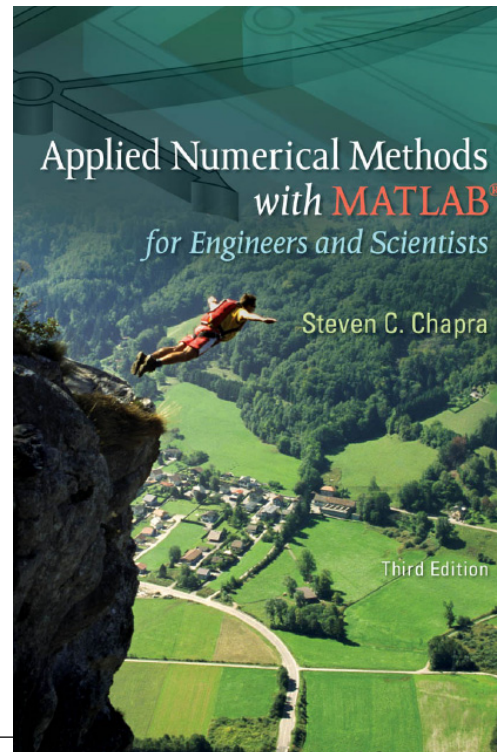
David Halpern
University of Alabama



CHAPMAN & HALL/CRC

A CRC Press Company
Boca Raton London New

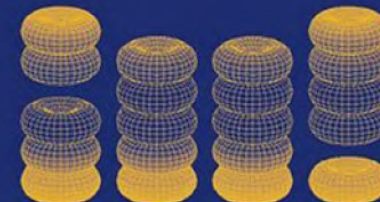
by Chapman & Hall/CRC



WILEY

Applied Numerical
Methods Using
MATLAB®

Won Y. Yang
Wenwu Cao, Tae-Sang Chung, and John Morris



WWW.
LIVE EXAMPLES

سرفصل های درسی

- مقدمه ای بر MATLAB
- اصول MATLAB
- اصول برنامه نویسی: عبارات شاخه ای
- اصول برنامه نویسی: حلقه ها
- توابع تعریف شده توسط کاربر
- مباحث پیشرفته توابع تعریف شده توسط کاربر
- کاربردهایی در مهندسی مکانیک و عمران

ارزشیابی

- تمرین‌های تحویلی و حضور و غیاب: ۱۰٪
- کوئیز و ارزشیابی کلاسی: ۳۰٪
- میان ترم: ۳۰٪
- پایان ترم: ۳۰٪
- دو بخش تستی و تشریحی
- در پایان ترم قسمت تشریحی نمره بیشتری دارد
- نوشتن برنامه های کامپیوتری کامل

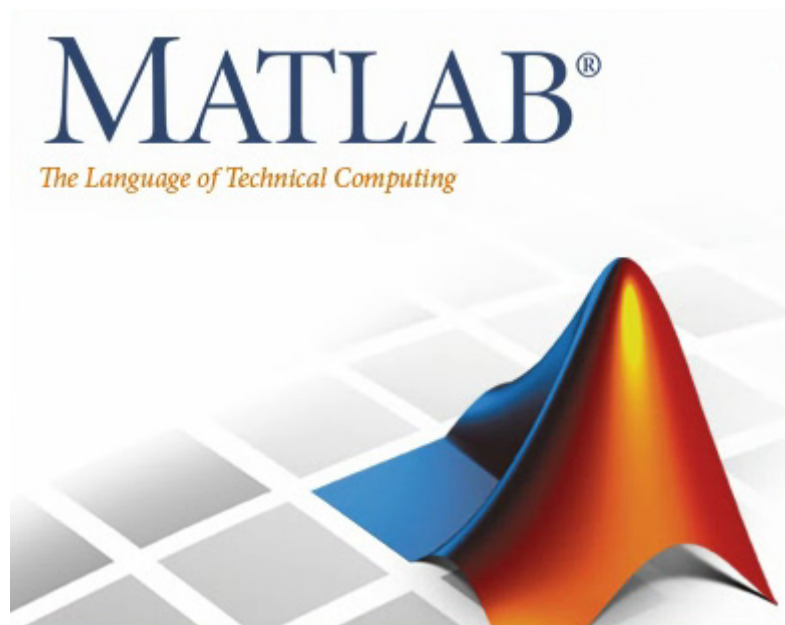
Email: soheili78@yahoo.com

فصل اول

مقدمه ای بر MATLAB

• MATLAB مخفف MATrix LABoratory

- یک برنامه کامپیوتری طراحی شده برای انجام محاسبات علمی و مهندسی است.
- در ابتدا به عنوان برنامه ای جهت انجام ریاضیات ماتریسی طراحی شده بود.
- MATLAB حل مسائل فنی را بسیار ساده تر از دیگر زبان های برنامه نویسی مانند Fortran و C کرده است.



- در سال ۲۰۱۸ بیش از ۳ میلیون کاربر
- قیمت این نرم افزار در سال ۲۰۲۰ برابر با
؟؟؟ دلار حدوداً؟؟ تومان

• Matlab سه جنبه اصلی دارد

1. نرم افزار ریاضی کاربردی: شامل انواع توابع ریاضی و فنی برای انجام

امور ریاضی و مهندسی

انواع جعبه ابزارها برای رشته های مختلف علوم و مهندسی

2. زبان برنامه نویسی: امکان ایجاد برنامه کامپیوتری دلخواه کاربر با

پشتوانه دستورات و ابزارهای متنوع (زبان نسل چهارم)

3. شبیه سازی سیستم های دینامیکی: محیطی تحت عنوان Simulink

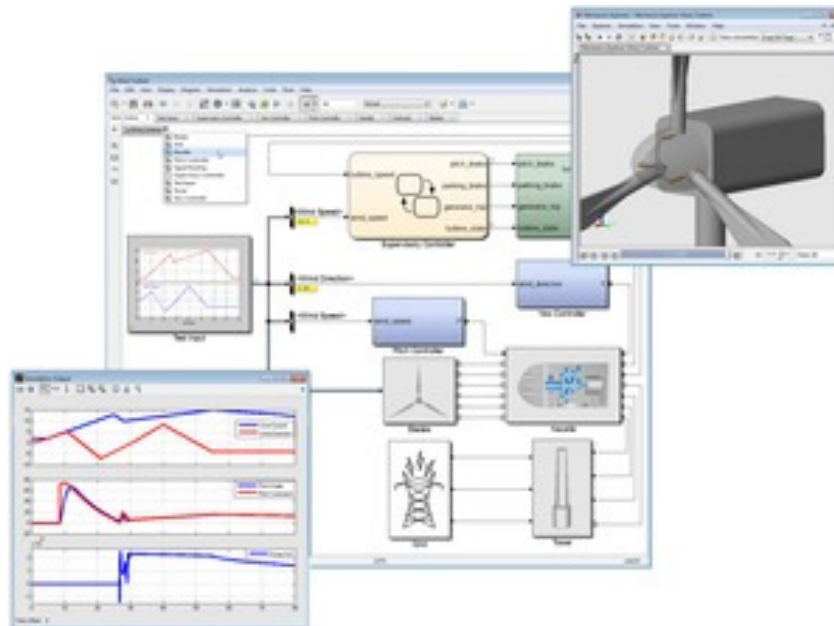
امکان شبیه سازی سیستم های دینامیکی را فراهم نموده است.

مزیت های MATLAB برای برنامه نویسی

1. راحتی استفاده: مانند بسیاری از نسخه های Basic یک زبان ترجمه شده است و استفاده از آن بسیار ساده است.
2. توابع از پیش تعریف شده: دارای کتابخانه گسترده از پیش تعریف شده برای بسیاری از کاربردهای علمی و مهندسی.
بسیاری از جعبه ابزارها (Toolbox) برای حل مسائل پیچیده در زمینه های خاص به کار گرفته می شوند. به عنوان مثال جعبه ابزار پردازش سیگنال، سیستم های کنترل، مخابرات، پردازش تصویر، مدلسازی سیستم های دینامیکی و ...
در پایگاه www.mathworks.com مجموعه وسیعی از برنامه های هدایی کاربران به اشتراک گذاشته شده است.

3. قابلیت ترسیمی: حاوی بسیاری از دستورات ترسیمی است. به یکی از برجسته ترین ابزارهای نمایش داده های فنی تبدیل شده است.
4. محیط گرافیکی برای کاربران (GUI): اجازه به برنامه نویس برای ایجاد محیط گرافیکی جهت استفاده آسان دیگر کاربران
5. Simulink: محیط برنامه نویسی گرافیکی برای شبیه سازی سیستم های

دینامیکی



نسخه های Matlab

- هر ساله دو نسخه از نرم افزار matlab عرضه می شود.
- نسخه های جدید حجم زیادی را نیاز دارند (بالغ بر 20Gb) و البته امکانات فراوانی هم دارند.
- برای درس برنامه نویسی استفاده از نسخه های قدیمی هم قابل قبول است.
- می توان از نسخه ۲۰۱۲ یا ۲۰۱۶ استفاده نمود ← حتما حتما روی کامپیوتر

نصب شود

- اپلیکیشن موبایلی Matlab mobile هم تا حدی قابل استفاده است.
- سایت زیر نیز تا قبل از نصب روی کامپیوتر می تواند مفید باشد.

www.octavonline.???

MATLAB میز کار

MATLAB R2015a

HOME PLOTS APPS EDITOR PUBLISH VIEW

FILE VARIABLE CODE ENVIRONMENT

Current Folder: /Users/sergiu/Desktop/.Softpedia test

Editor - SoftpediaAnimation.m

```

1 - degree = 6;
2 - order = 1;
3 - amplitude = 0.5;
4 - radius = 5;
5
6 - Ymn = legendre(degree,cos(theta(:,1)));
7 - Ymn = Ymn(order+1,:);
8 - yy = Ymn;
9
10 - for kk = 2: size(theta,1)
11 -     yy = [yy Ymn];
12 - end
13
14 - yy = yy.*cos(order*phi);
15
16 - order = max(max(abs(yy)));
17 - rho = radius + amplitude*yy/order;
18
19 - r = rho.*sin(theta); % convert to Cartesian coordinates
20 - x = r.*cos(phi);
21 - y = r.*sin(phi);
22 - z = rho.*cos(theta);
23

```

Command Window

```

>> SoftpediaTest
>> semilogx(lastAccessed1)
>> semilogy(lastAccessed1)
>> histogram(lastAccessed1)
>> loglog(lastAccessed1)
>> comet(lastAccessed1)
>> stem(lastAccessed1)
>> stairs(lastAccessed1)
>> barh(lastAccessed1)
>> loglog(lastAccessed1)
>>

```

Variables - baseDomain

Variable	Value
0.5000	0.5000
60x60 sparse do...	60x60 sparse do...
2079x1 cell	2079x1 cell
2078x1 cell	2078x1 cell
2079x1 double	2079x1 double
2078x1 double	2078x1 double
6	6
2079x1 double	2079x1 double
2078x1 double	2078x1 double
60x60 sparse do...	60x60 sparse do...
2079x1 cell	2079x1 cell
2078x1 cell	2078x1 cell
2079x1 double	2079x1 double
2078x1 double	2078x1 double
2079x1 double	2079x1 double
2078x1 double	2078x1 double
2079x1 double	2079x1 double
2078x1 double	2078x1 double
1x30 double	1x30 double
2079x1 double	2079x1 double
2078x1 double	2078x1 double
2079x1 cell	2079x1 cell
2078x1 cell	2078x1 cell
1	1
2079x1 cell	2079x1 cell
2078x1 cell	2078x1 cell
5	5
60x3 double	60x3 double
2079x1 cell	2079x1 cell
2078x1 cell	2078x1 cell

Ln 34 Col 1

Current Directory Browser shows a list of the files in the current directory

Launch the **Help Browser**

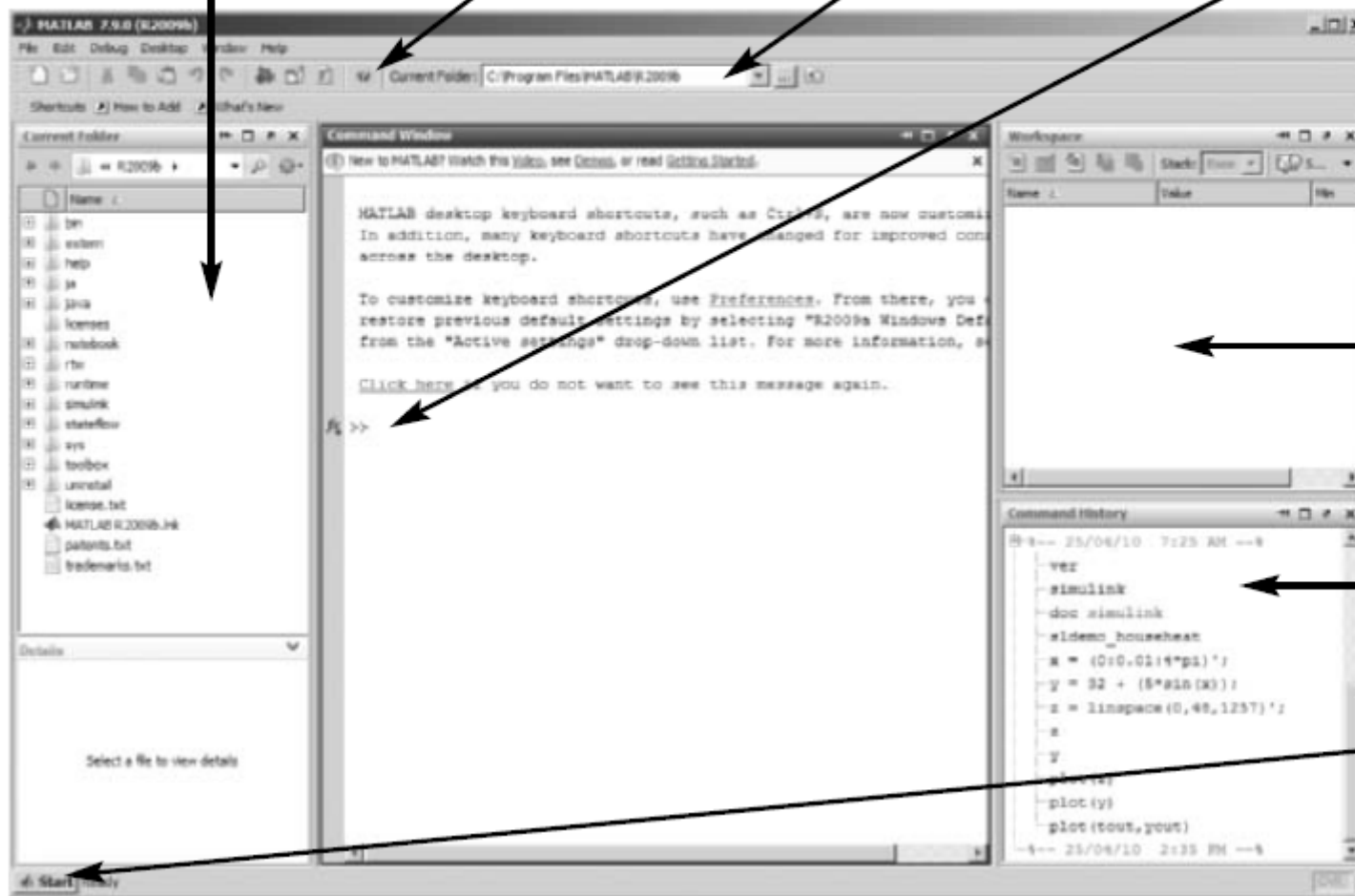
This control allow a user to view or change the current directory

MATLAB Command Window

Workspace Browser shows variables defined in workspace

Command History window displays previous commands

Start Button launches toolboxes, MATLAB tools, etc.

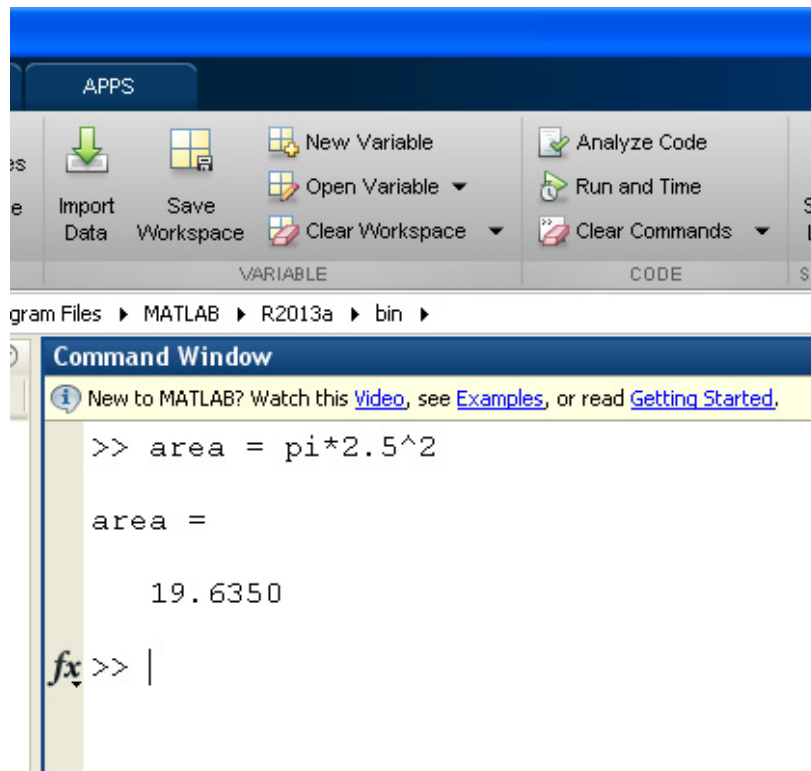


پنجره های Matlab

Window	Purpose
Command Window	Main window, enters variables, runs programs.
Figure Window	Contains output from graphic commands.
Editor Window	Creates and debugs script and function files.
Help Window	Provides help information.
Command History Window	Logs commands entered in the Command Window.
Workspace Window	Provides information about the variables that are stored.
Current Folder Window	Shows the files in the current folder.

پنجره فرمان

- دستورات در این پنجره وارد شده و پاسخ آن به صورت آنی گرفته می شود.



- مثال: محاسبه مساحت دایره

به شعاع ۲.۵ متر

- پاسخ در متغیری (آرایه 1*1)

به نام area ذخیره شده

• عملیات ریاضی

<u>Operation</u>	<u>Symbol</u>	<u>Example</u>
Addition	+	$5 + 3$
Subtraction	-	$5 - 3$
Multiplication	*	$5 * 3$
Right division	/	$5 / 3$
Left division	\	$5 \setminus 3 = 3 / 5$
Exponentiation	^	$5 \wedge 3$ (means $5^3 = 125$)

• ترتیب انجام عملیات

<u>Precedence</u>	<u>Mathematical Operation</u>
First	Parentheses. For nested parentheses, the innermost are executed first.
Second	Exponentiation.
Third	Multiplication, division (equal precedence).
Fourth	Addition and subtraction.

```
output = 3*(2+5)*3;  
output = 3*7*3;
```

```
output = 3^2^3;
```

```
output = 9^3;
```

```
output = 729;
```

```
output = 3^(2^3);
```

```
output = 3^8;
```

```
output = 6561;
```

• از MATLAB به عنوان یک ماشین حساب قدرتمند نیز می توان استفاده نمود.

```
>> 7+8/2
ans =
    11
>> (7+8)/2
ans =
    7.5000
>> 4+5/3+2
ans =
    7.6667
>> 5^3/2
ans =
    62.5000
>> 27^(1/3)+32^0.2
ans =
     5
>> 27^1/3+32^0.2
ans =
    11
>> 0.7854-(0.7854)^3/(1*2*3)+0.785^5/(1*2*3*4*5)...
- (0.785)^7/(1*2*3*4*5*6*7)
ans =
    0.7071
>>
```

Type and press Enter.
8/2 is executed first.

Type and press Enter.
7+8 is executed first.

5/3 is executed first.

5^3 is executed first, /2 is executed next.

1/3 is executed first, 27^(1/3) and 32^0.2 are executed next, and + is executed last.

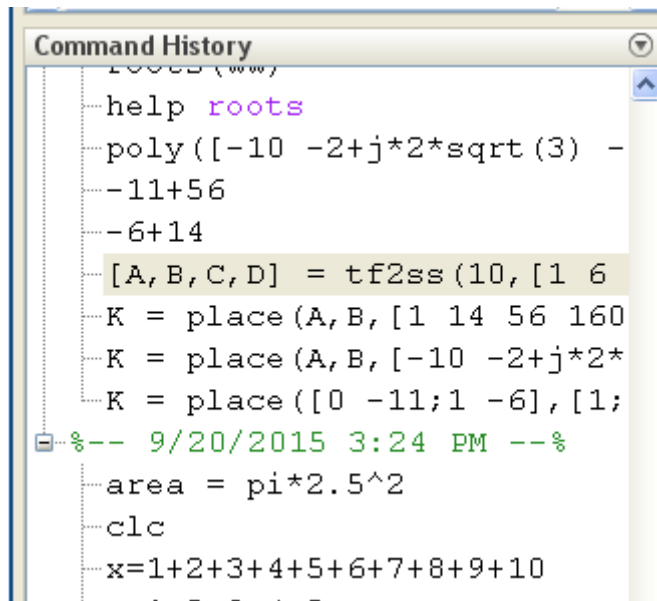
27^1 and 32^0.2 are executed first, /3 is executed next, and + is executed last.

Type three periods ... (and press Enter) to continue the expression on the next line.

The last expression is the first four terms of the Taylor series for $\sin(\pi/4)$.

پنجره تاریخچه فرمان

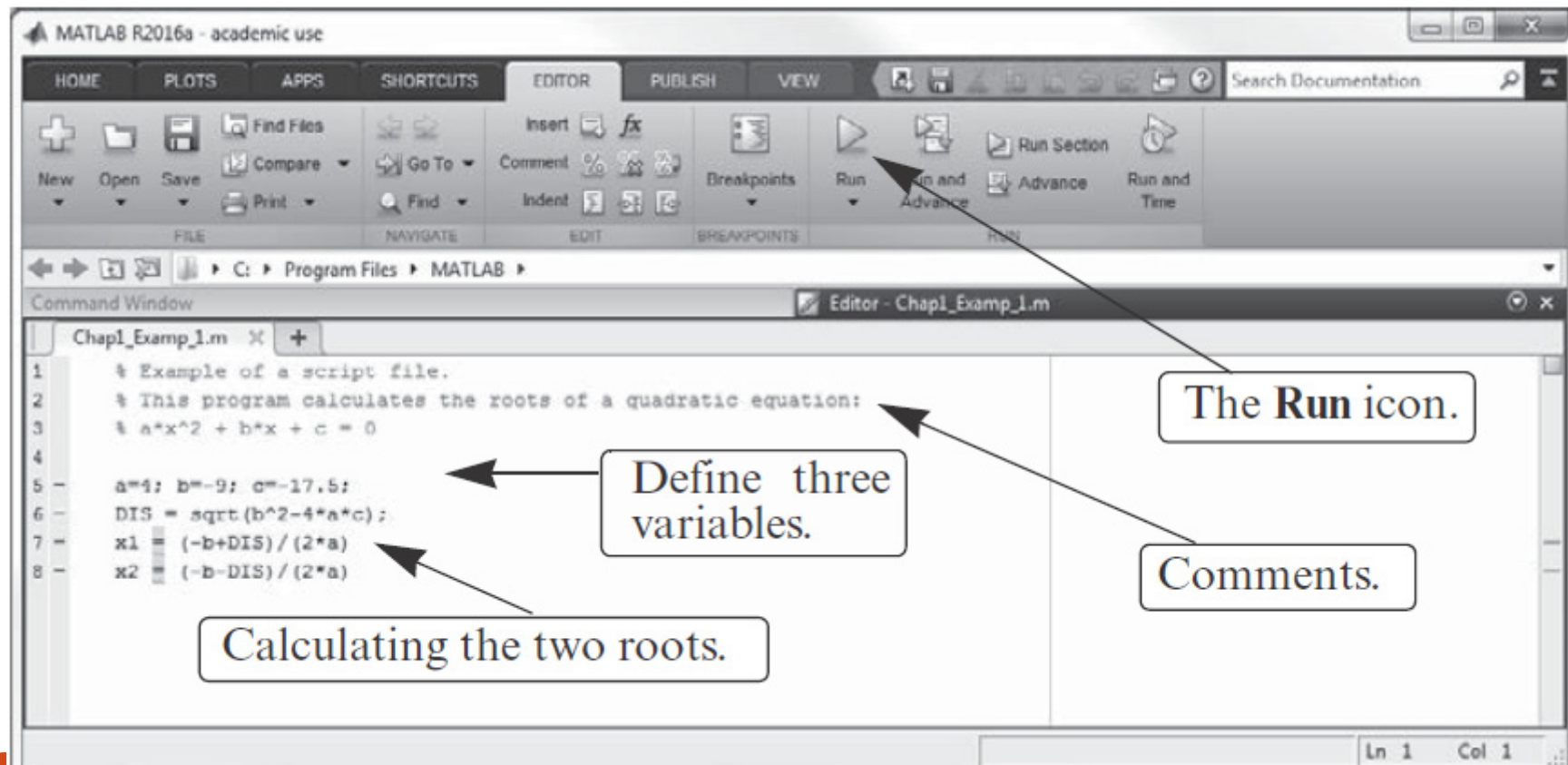
- Command history window از لیست دستورات قبلی تشکیل شده است.



```
Command History
help roots
poly([-10 -2+j*2*sqrt(3) -
-11+56
-6+14
[A,B,C,D] = tf2ss(10,[1 6
K = place(A,B,[1 14 56 160
K = place(A,B,[-10 -2+j*2*
K = place([0 -11;1 -6],[1;
%-- 9/20/2015 3:24 PM --%
area = pi*2.5^2
clc
x=1+2+3+4+5+6+7+8+9+10
```

پنجره تصحیح / عیب یابی

- این پنجره در اصل یک ویرایشگر متون برنامه است و برای ساخت یا اصلاح m-file استفاده می شود (مناسب برنامه نویسی).



Editor - D:\Program Files\MATLAB\R2013a\toolbox\mpc\mpcdemos\mpccstr.m

EDITOR PUBLISH VIEW

New Open Save Find Files Compare Print Insert Comment Indent Go To Find Breakpoints Run Run and Time Run and Advance Advance

snake_oscillator4.m x mpccstr.m x

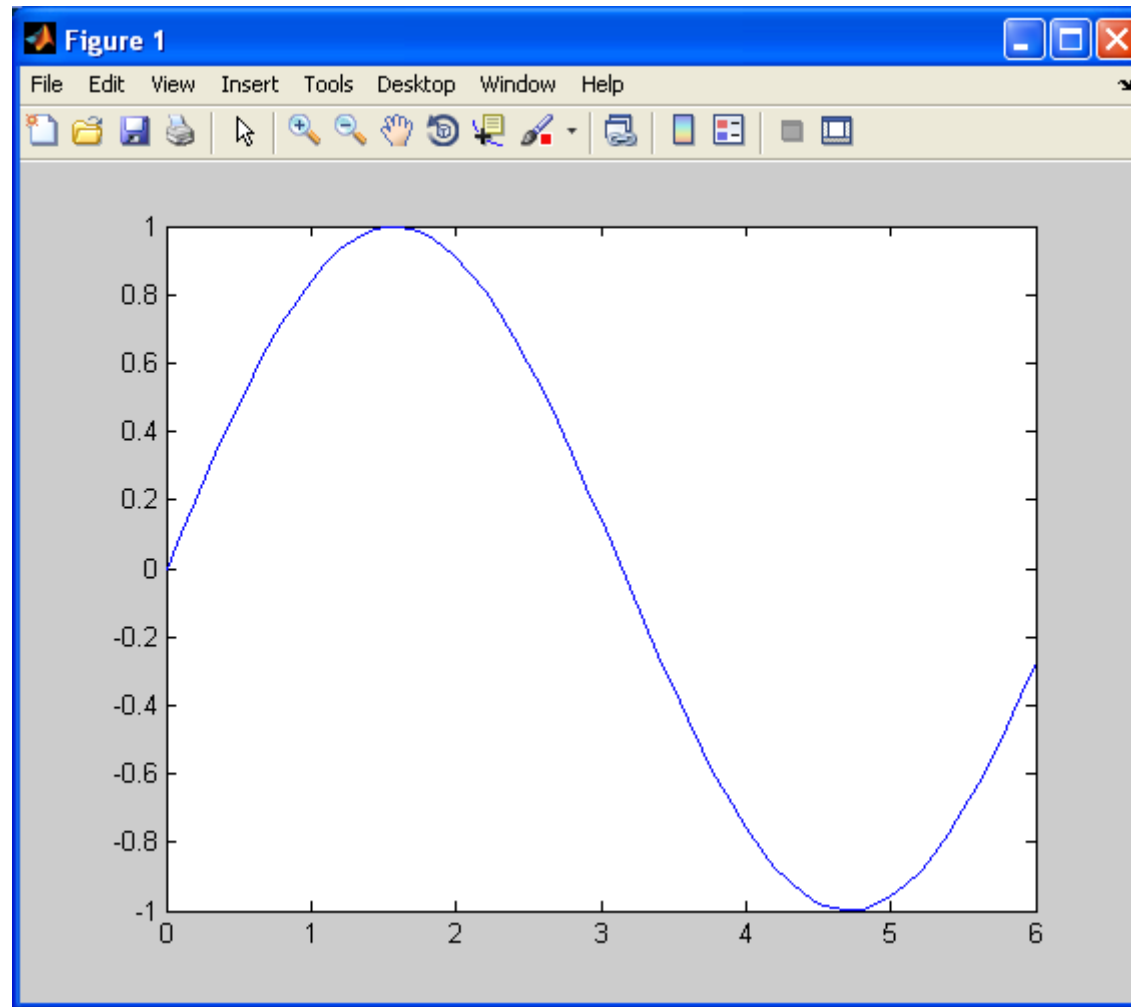
This file can be published to a formatted document. For more information, see the publishing [video](#) or [help](#).

```
7 % block is used to illustrate a scheduling based strategy to solve a
8 % nonlinear control problem with multiple operating conditions.
9 %
10 % The reactor is modeled by 'CSTR_INOUT.mdl'. For background, see S.
11 % D.E. et al., "Process Dynamics and Control", 2nd Ed., 2004, Wiley,
12 % pp.34-36.
13
14 % Copyright 1990-2012 The MathWorks, Inc.
15 % $Revision: 1.1.8.8 $ $Date: 2012/03/05 23:01:55 $
16
17 %% Open-Loop Model: Linearize the Nonlinear CSTR Model
18 - if ~mpcchecktoolboxinstalled('simulink')
19 -     disp('Simulink(R) is required to run this example.')
20 -     return
21 - end
22 - if ~mpcchecktoolboxinstalled('slcontrol')
23 -     disp('Simulink Control Design(R) is required to run this example')
24 -     return
25 - end
26
27 -
```

پنجره تصاویر

- برای نشان دادن تصاویر و اشکال گرافیکی استفاده می شود.
- نوشتن فرامین زیر در ویرایشگر متلب و اجرای آن، تابع سینوس را در پنجره تصویر رسم می کند.

```
% sin_x.m: This M-file calculates and plots the  
% function sin(x) for 0 <= x <= 6.  
x = 0:0.1:6  
y = sin(x);  
plot(x,y);
```

نحوه ترسیم به زبان ساده

- مرحله ۱) تقسیم نقاط بازه مورد نظر

```
>> x = 0:0.1:6
```

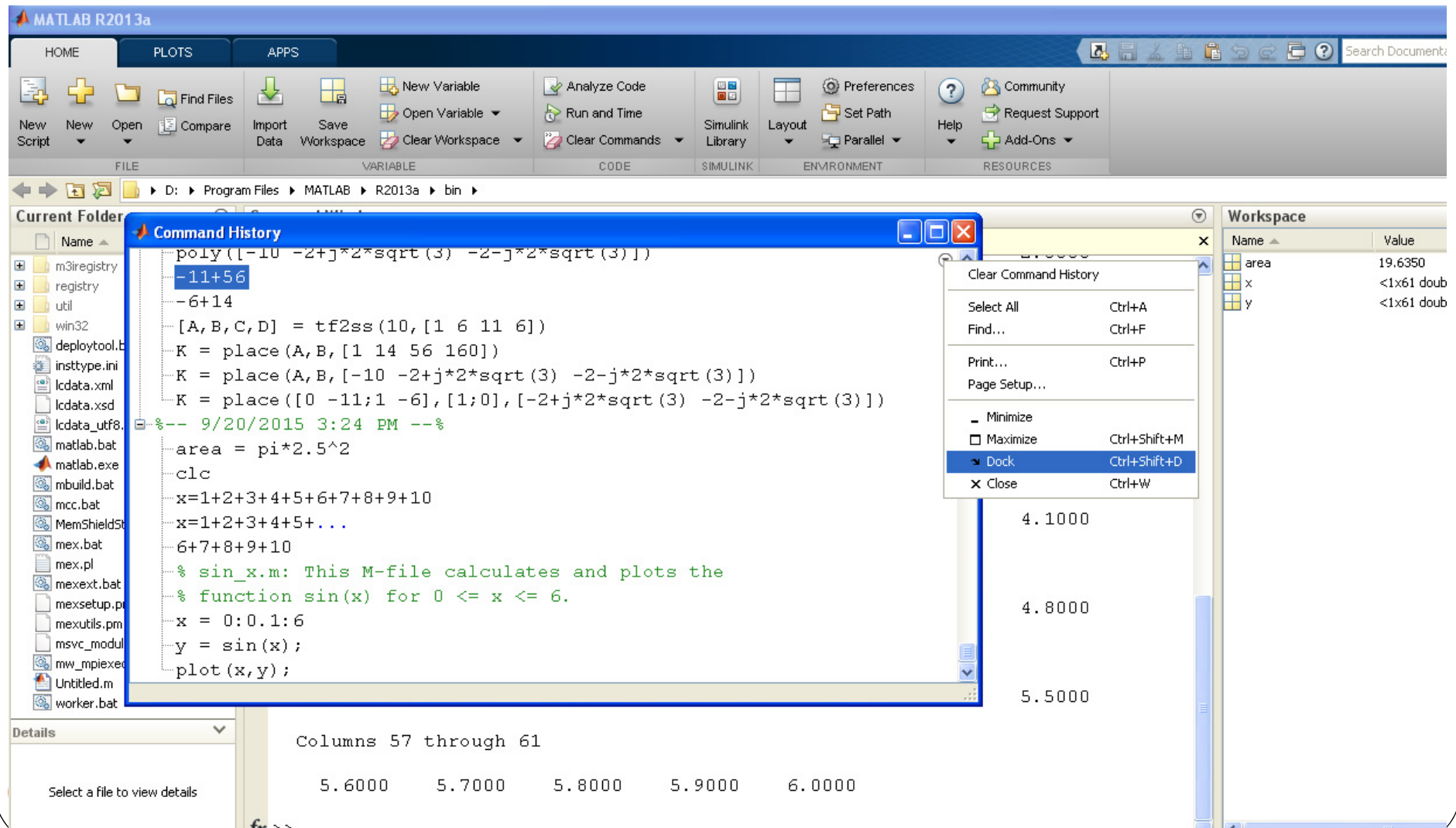
- مرحله ۲) محاسبه مقدار تابع در نقاط

```
>> y = sin(x)
```

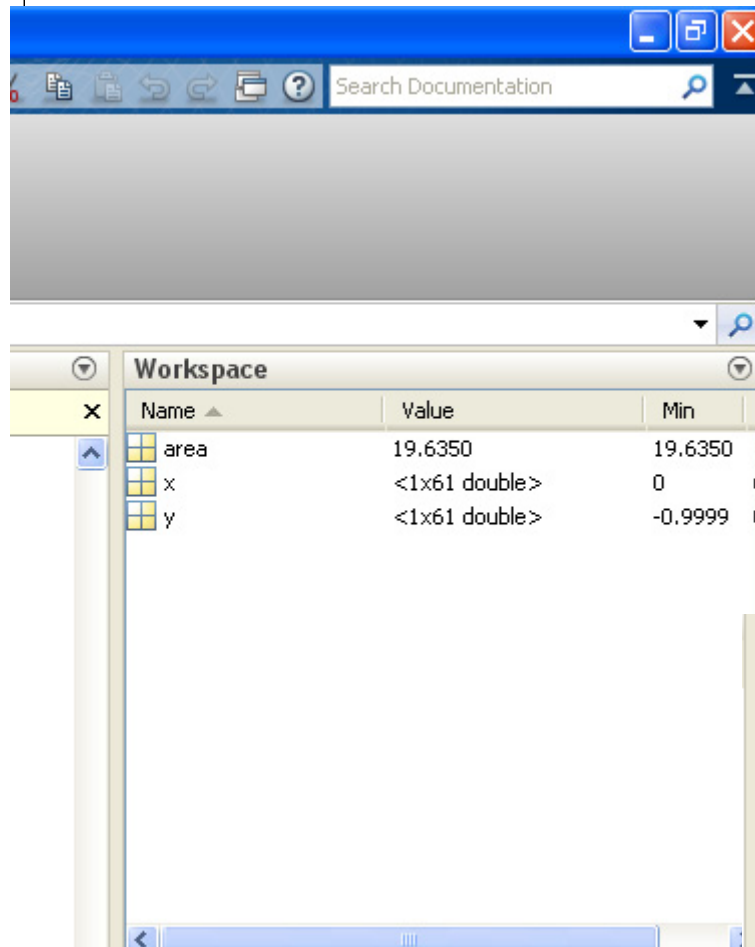
- مرحله ۳) ترسیم نقاط در صفحه

```
>> plot(x,y)
```

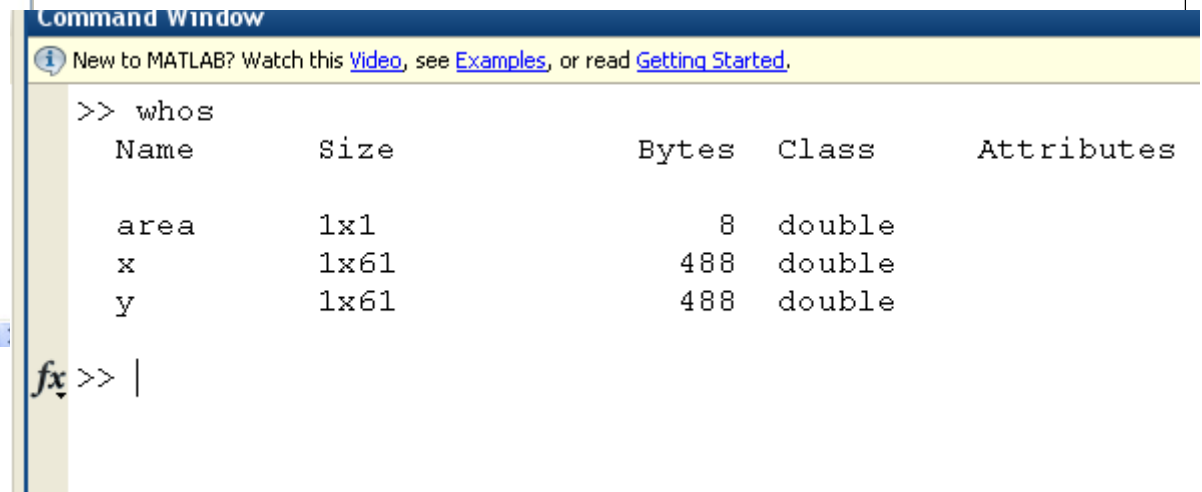
- می توان پنجره های مختلف را از محیط متلب جدا (undock) و به صورت یک پنجره مستقل استفاده نمود.



فضای کاری (MATLAB workspace)



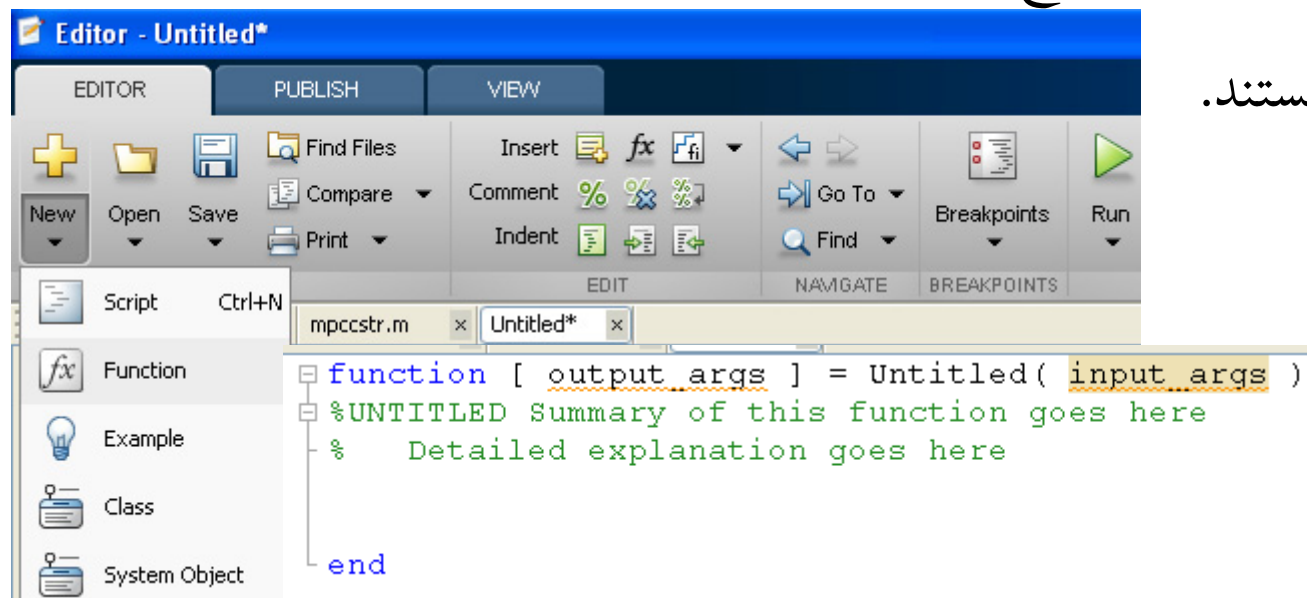
- شامل تمام متغیرها و آرایه هایی است که در پنجره فرمان یا script file اجرا شده است
- با فرمان whos تمام متغیرهای workspace لیست می شوند



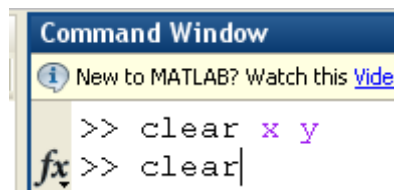
- متغیرهای تعریف شده در workspace در تمام قسمت های متلب مشترک هستند

- در آینده خواهیم دید که فقط توابع (functions) دارای workspace

مختص به خودشان هستند.

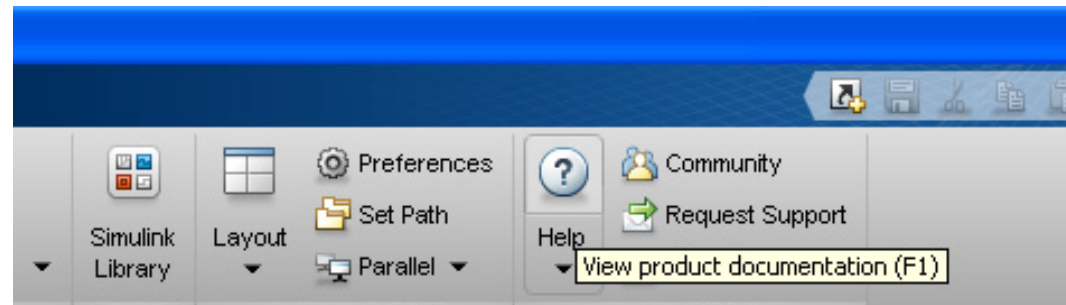


- با دستور clear می توان متغیرها را از فضای کاری حذف نمود.



استفاده از Help

1. انتخاب help از نوار ابزار و جستجوی عبارت مورد نظر



2. تایپ نام دستور بعد از عبارت help

```
>> help plot
plot    Linear plot.
       plot(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
       then the vector is plotted versus the rows or columns of the matrix,
       whichever line up. If X is a scalar and Y is a vector, disconnected
       line objects are created and plotted as discrete points vertically at
       X.

       plot(Y) plots the columns of Y versus their index.
       If Y is complex, plot(Y) is equivalent to plot(real(Y),imag(Y)).
       In all other uses of plot, the imaginary part is ignored.
```

3. استفاده از فرمان lookfor و نوشتن عبارتی برای جستجو

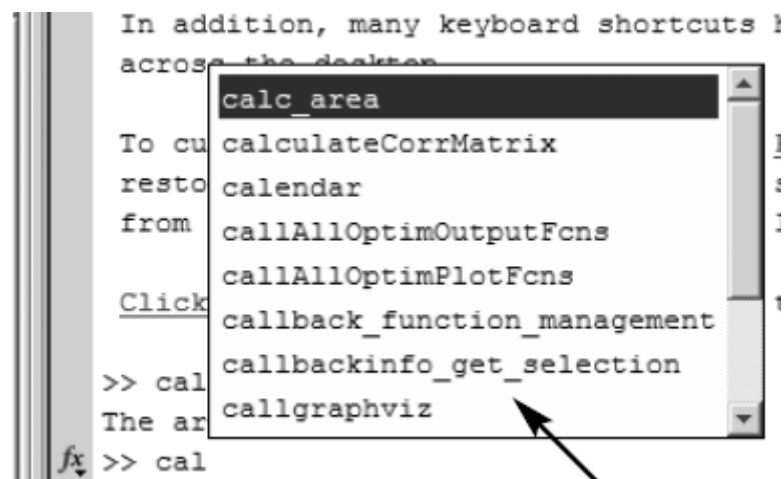
```
>> lookfor plotting
xpquad
sfunxy
aero_radplot
plot_adapt_coefs
sf_pool_plotter
commdoc_const
exresp
plotbode
plotnic
plotnyq
respdemo
aero_radplot_dsp
hfigslpc
hfigsstats
lmsxy
portfolioexamples_plot
getff
utidplot
mlayers
mapex3ddome
mpcstair
sfunxy2
```

- Superquadrics plotting demonstration.
- S-function that acts as an X-Y scope using MAT
- RADPLOT Radar Data Processing Tracker plotting
- Callback function for plotting the current ada
- GUI for plotting and interaction for the sf_po
- Plotting Signal Constellations
- Example response, used by plotting control rou
- Plots Bode responses given the plotting data.
- Plots Nichols responses given the plotting dat
- Plots Nyquist responses given the plotting dat
- Plotting System Responses
- Radar Tracking plotting function
- Setup and cleanup scope window for plotting da
- Set up figures for plotting data in dspStats d
- S-function that acts as an X-Y scope using MAT
- Specialized plotting function for portfolioexa
- Selects the frequency function (for own plotti
- Utility code used for plotting data.
- Control plotting of display structure elements
- Plotting a 3-D Dome as a Mesh Over a Globe
- Turns an array into a stair-step function for
- S-function that acts as an X-(double Y) scope

چند دستور مهم

- با تایپ دستور demo بسیاری از قابلیت های نرم افزار نمایش داده می شود.
- پنجره فرمان با دستور clc پاک می شود
- متغیرهای درون workspace با دستور clear پاک می شود.
- در ابتدای هر m-file یا قبل از شروع هر محاسبه جداگانه از دستور clear استفاده شود.
- Ctrl+c برای قطع اجرای یک برنامه

- با نوشتن چند کاراکتر یک دستور و زدن کلید **tab** می توان آن را کامل نمود



- برخی از نام ها توسط Matlab استفاده می شوند و نمی توان به عنوان نام متغیر استفاده نمود.

```
break case catch classdef continue else elseif end for  
function global if otherwise parfor persistent return  
spmd switch try while
```

مقادير خاص

Function	Purpose
<code>pi</code>	Contains π to 15 significant digits.
<code>i, j</code>	Contain the value i ($\sqrt{-1}$).
<code>Inf</code>	This symbol represents machine infinity. It is usually generated as a result of a division by 0.
<code>NaN</code>	This symbol stands for not-a-number. It is the result of an undefined mathematical operation, such as the division of zero by zero.
<code>clock</code>	This special variable contains the current date and time in the form of a six-element row vector containing the year, month, day, hour, minute, and second.
<code>date</code>	Contains the current data in a character string format, such as 24-Nov-1998.
<code>eps</code>	This variable name is short for “epsilon.” It is the smallest difference between two numbers that can be represented on the computer.
<code>ans</code>	A special variable used to store the result of an expression if that result is not explicitly assigned to some other variable.

مسیر جستجو در MATLAB

- با وارد نمودن نام در پنجره فرمان، به صورت زیر جستجو می شود.

1. به نام وارد شده به صورت یک متغیر می نگرد

2. بررسی می کند که نام وارد شده یک دستور یا تابع موجود باشد

3. بررسی می کند که نام وارد شده نام m-file در شاخه کنونی

(current directory) است

4. بررسی می کند که نام وارد شده در یکی از شاخه های search path باشد.



- هیچگاه مقدار یک متغیر از پیش تعریف شده را در MATLAB تغییر ندهید. این کار باعث بروز خطاهای ظریف غیرقابل شناسایی می شود.
- در واقع با این کار بستری برای رخداد یک فاجعه آماده می شود

Command Window

New to MATLAB? Watch this [Video](#), see [Examples](#), or read [Getting Started](#).

```
>> pi
ans =
    3.1416
>>
>> a = pi * 2.5^2
a =
    19.6350
>>
>> pi = -3;
>>
>> a = pi * 2.5^2
a =
   -18.7500
>>
>> clear pi
>>
>> a = pi * 2.5^2
a =
    19.6350
fx>> |
```

تعریف متغیرهای اسکالر

- علامت = اپراتور انتساب است و معنی برابری را در نرم افزار نمی دهد.

Variable_name = A numerical value, or a computable expression

```
>> x=15
```

```
x =  
    15
```

```
>> x=3*x-12
```

```
x =  
    33
```

```
>>
```

The number 15 is assigned to the variable x.

MATLAB displays the variable name and its assigned value.

A new value is assigned to x. The new value is 3 times the previous value of x minus 12.

```
>> a=12
```

Assign 12 to a.

```
a =  
    12
```

```
>> B=4
```

Assign 4 to B.

```
B =  
    4
```

```
>> C=(a-B)+40-a/B*10
```

Assign the value of the expression on the right-hand side to the variable C.

```
C =  
    18
```

• استفاده از سمی کالون ;

```
>> a=12;
```

```
>> B=4;
```

```
>> C=(a-B)+40-a/B*10;
```

The variables a, B, and C are defined but are not displayed, since a semicolon is typed at the end of each statement.

```
>> C
```

```
C =  
    18
```

The value of the variable C is displayed by typing the name of the variable.

- انتساب می تواند در یک خط انجام شود.

```
>> a=12, B=4; C=(a-B)+40-a/B*10
```

```
a =  
    12
```

```
C =  
    18
```

The variable B is not displayed because a semi-colon is typed at the end of the assignment.

- با تعریف متغیر می توان از آن در ورودی توابع استفاده نمود.

```
>> x=0.75;
```

```
>> E=sin(x)^2+cos(x)^2
```

```
E =
```

```
    1
```

```
>>
```


توضیحات تکمیلی در مورد متغیرها

- سه نوع متغیر می توان تعریف نمود.

```
>> a=4; % class double
>> fname='Robert'; % class char
>> temperature=101.2; % class double
>> isDone=true; % class logical
```

- استفاده از ; در انتهای فرامین

```
>> a=4.2
a =
    4.2000
>> a=5.5;
>>
```

- فاصله در نام متغیر مجاز نیست بجای آن از _ استفاده شود. (خط تیره اشتباه است)

`calc_energy`

- MATLAB به بزرگ و کوچک بودن نام متغیرهای حساس است

`name, NAME, and Name`

- متغیر نوع `double` مقادیر اسکالر و اعشاری و مختلط

```
var = 10.5
```

```
var = 10 + 10i
```

- متغیر نوع `char` کاراکتر و حروف را در خود جای می دهد

```
comment = 'This is a character string'
```

- متغیر نوع `logical` درست (1) و غلط (0) بودن را در خود جای می دهد

```
dis = true
```

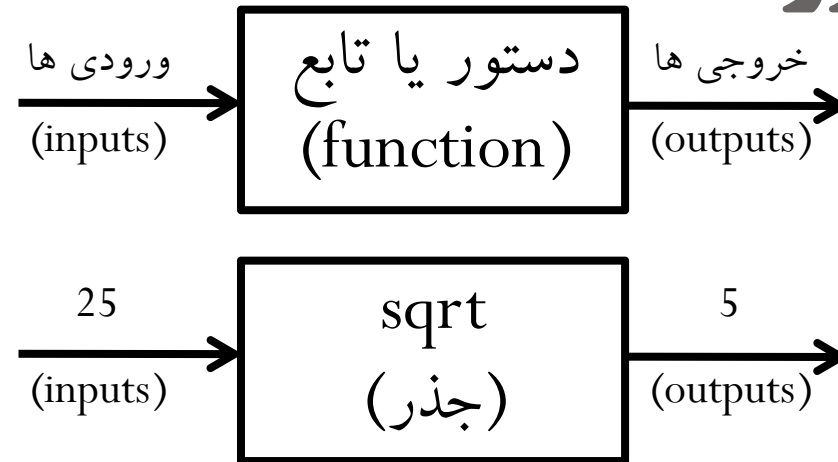
نمایش داده های خروجی

- با دستور `format` می توان نحوه نمایش اعداد را تغییر داد

Format Command	Results	Example ¹
<code>format short</code>	4 digits after decimal (default format)	12.3457
<code>format long</code>	14 digits after decimal	12.34567890123457
<code>format short e</code>	5 digits plus exponent	1.2346e+001
<code>format short g</code>	5 total digits with or without exponent	12.346
<code>format long e</code>	15 digits plus exponent	1.234567890123457e+001
<code>format long g</code>	15 total digits with or without exponent	12.3456789012346
<code>format bank</code>	“dollars and cents” format	12.35
<code>format hex</code>	hexadecimal display of bits	4028b0fcd32f707a
<code>format rat</code>	approximate ratio of small integers	1000/81
<code>format compact</code>	suppress extra line feeds	
<code>format loose</code>	restore extra line feeds	
<code>format +</code>	only signs are printed	+

¹ The data value used for the example is 12.345678901234567 in all cases.

اجرای دستورات



• نحوه اجرای دستورات در Matlab

```
>> (ورودی ها) نام تابع = [خروجی ها]
```

```
>> [ b] = sqrt(25)
```

```
b = 5
```

خروجی تابع جذر به متغیر b نسبت داده می شود

برخی از دستورات پر کاربرد

Function	Description	Example
<code>sqrt (x)</code>	Square root.	<pre>>> sqrt(81) ans = 9</pre>
<code>nthroot (x,n)</code>	Real n th root of a real number x . (If x is negative n must be an odd integer.)	<pre>>> nthroot(80,5) ans = 2.4022</pre>
<code>exp (x)</code>	Exponential (e^x).	<pre>>> exp(5) ans = 148.4132</pre>
<code>abs (x)</code>	Absolute value.	<pre>>> abs(-24) ans = 24</pre>
<code>log (x)</code>	Natural logarithm. Base e logarithm ($\ln$).	<pre>>> log(1000) ans = 6.9078</pre>
<code>log10 (x)</code>	Base 10 logarithm.	<pre>>> log10(1000) ans = 3.0000</pre>
<code>factorial (x)</code>	The factorial function $x!$ (x must be a positive integer.)	<pre>>> factorial(5) ans = 120</pre>

Function	Description	Example
<code>sin(x)</code> <code>sind(x)</code>	Sine of angle x (x in radians). Sine of angle x (x in degrees).	<pre>>> sin(pi/6) ans = 0.5000</pre>
<code>cos(x)</code> <code>cosd(x)</code>	Cosine of angle x (x in radians). Cosine of angle x (x in degrees).	<pre>>> cosd(30) ans = 0.8660</pre>
<code>tan(x)</code> <code>tand(x)</code>	Tangent of angle x (x in radians). Tangent of angle x (x in degrees).	<pre>>> tan(pi/6) ans = 0.5774</pre>
<code>cot(x)</code> <code>cotd(x)</code>	Cotangent of angle x (x in radians). Cotangent of angle x (x in degrees).	<pre>>> cotd(30) ans = 1.7321</pre>

The inverse trigonometric functions are `asin(x)`, `acos(x)`, `atan(x)`, `acot(x)` for the angle in radians; and `asind(x)`, `acosd(x)`, `atand(x)`, `acotd(x)` for the angle in degrees. The hyperbolic trigonometric functions are `sinh(x)`, `cosh(x)`, `tanh(x)`, and `coth(x)`. Table 1-4 uses `pi`, which is

توابع رند کردن

Rounding functions

`ceil(x)`

Rounds x to the nearest integer towards positive infinity:

`ceil(3.1) = 4` and `ceil(-3.1) = -3`.

`fix(x)`

Rounds x to the nearest integer towards zero: `fix(3.1) = 3`

and `fix(-3.1) = -3`.

`floor(x)`

Rounds x to the nearest integer towards minus infinity:

`floor(3.1) = 3` and `floor(-3.1) = -4`.

`round(x)`

Rounds x to the nearest integer.

یک محاسبه ساده

- بررسی کنید که معادله ریاضی زیر درست است ($x = \pi/5$)

$$\cos^2 \frac{x}{2} = \frac{\tan x + \sin x}{2 \tan x}$$

```
>> x=pi/5;
```

Define x.

```
>> LHS=cos(x/2)^2
```

Calculate the left-hand side.

```
LHS =  
0.9045
```

```
>> RHS=(tan(x)+sin(x))/(2*tan(x))
```

Calculate the right-hand side.

```
RHS =  
0.9045
```


یک برنامه ساده

- با معرفی شتاب گرانش، جرم، ارتفاع و سرعت به محاسبه انرژی جنبشی، انرژی پتانسیل و انرژی کل پردازد.

$$E_{kinetic} = \frac{1}{2}mv_y^2$$

$$E_{potential} = mgy$$

$$E_{total} = E_{kinetic} + E_{potential}$$

- محیط editor باز شود و برنامه نوشته شود.
- پس از پایان نوشتن برنامه با اسمی مناسب (مثلاً calcEnergy.m) ذخیره شود. پسوند فایل های برنامه نویسی m است (m-file)

- قدم اول) معرفی ورودی های مساله

```
g=9.81;  
m=0.01;  
y=6.0;  
vy=5.2;
```

- قدم دوم) انجام محاسبات

```
Ekin=0.5*m*vy^2;  
Epot=m*g*y;  
Etot=Ekin+Epot;
```

- قدم سوم) نمایش نتایج

```
disp('Ekin');  
disp(Epot);  
disp(Etot);
```

برنامه قبل به صورت کاملتر

```
%% calcParticleEnergy.m
%   Calculate potential, kinetic, and total energy
%   of a point particle
%   Author: G. Galilei



---



%% set physical parameters
g=9.81;      % acceleration due to gravity (m/s^2)
m=0.01;      % mass (kg)
y=6.0;       % height(m)
vy=5.2;      % velocity(m/s)
```

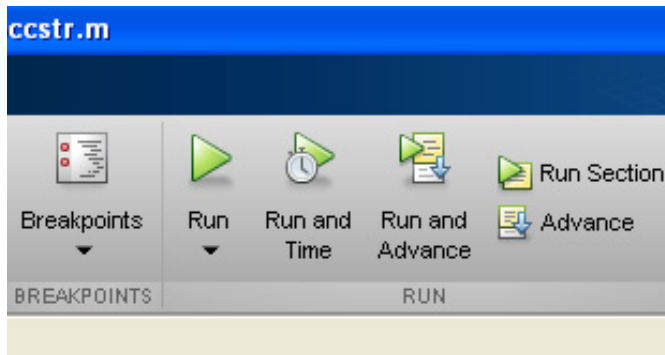
```
%% calculate energies (in Joules)
Ekin=0.5*m*vy^2;
Epot=m*g*y;
Etot=Ekin+Epot;

%% display results

disp(['  Calculation of particle energy ']);
disp(['-----']);
disp(['mass(kg)=          ',num2str(m)]);
disp(['height(m)=         ',num2str(y)]);
disp(['velocity (m/s)=    ',num2str(vy)]);
disp([' ']);
disp(['Kinetic Energy(J)   = ',num2str(Ekin)]);
disp(['Potential Energy(J) = ',num2str(Epot)]);
disp(['Total Energy(J)    = ',num2str(Etot)]);
```

نحوه اجرای برنامه

- پس از ذخیره برنامه در محیط editor (مثلاً با نام calcEnergy.m) از دو طریق می توان آن را اجرا نمود:



1. فشردن کلید Run در محیط editor

2. نوشتن نام فایل در پنجره command

```
>> calcEnergy
```

محاسبه ریشه های چند جمله ای درجه دو

- قدم اول) معرفی ورودی های مساله

$a = 4; b = -9; c = -17.5;$

- قدم دوم) انجام محاسبات

$DIS = \sqrt{b^2 - 4*a*c};$

$x1 = (-b + DIS)/(2*a);$

$x2 = (-b - DIS)/(2*a);$

- قدم سوم) نمایش نتایج

$\text{disp}(x1);$

$\text{disp}(x2);$

