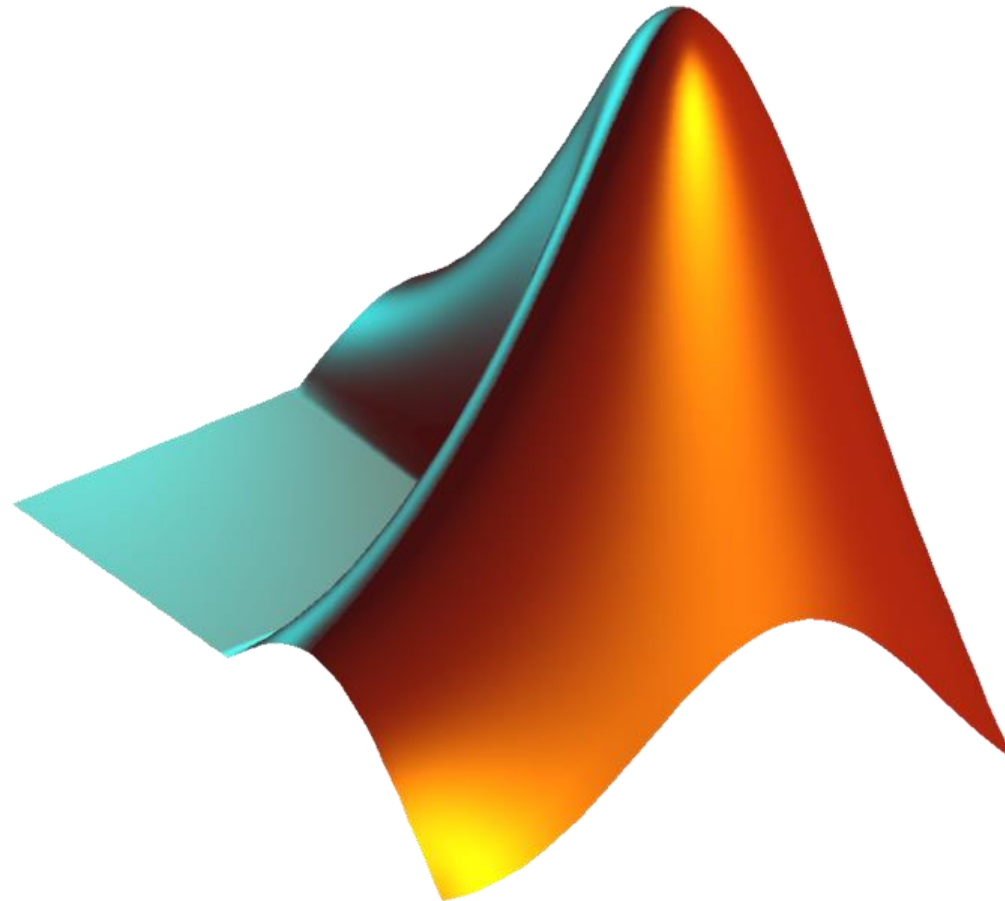




# MATLAB COURSE



مدرس دوره : احمد خیراندیش

Session: **1**

## نرم افزار متلب

توسعه نرم افزار متلب در اواخر سال ۱۹۷۰ توسط **Cleve Moler** (رئیس بخش علوم کامپیوتر دانشگاه نیومکزیکو)

آغاز شد و هدف **Moler** از این کار این بود که دانشجویان بدون نیاز به یادگیری زبان فرترن، نیازهای درسی خود را برطرف نمایند.

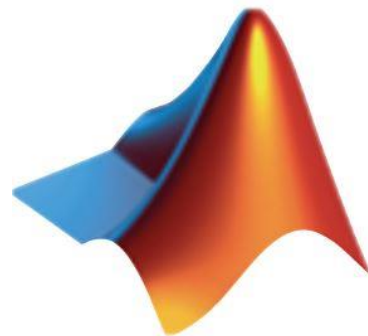


سرانجام در سال ۱۹۸۳، پیشنهاد تجاری سازی این نرم افزار توسط **Jack Little** در

ملاقاتش با **Moler** در دانشگاه استنفورد مطرح گردید و در نهایت در سال ۱۹۸۴

شرکت **MathWorks** تاسیس گردید. آنها متلب را مجدداً با زبان C بازنویسی کردند و


















کتابخانه های متعددی را به این نرم افزار افزودند.



# MathWorks®

www.eeweb.ir احمد خیراندیش

# Matlab Language Rank in 2016

Language Rank	Types	Spectrum Ranking
1. C		100.0
2. Java		98.1
3. Python		98.0
4. C++		95.9
5. R		87.9
6. C#		86.7
7. PHP		82.8
8. JavaScript		82.2
9. Ruby		74.5
10. Go		71.9
11. Swift		70.1
12. Arduino		69.9
13. Assembly		68.6
14. Matlab		68.5
15. Scala		66.9
16. HTML		65.6
17. Perl		58.5

## نسخه های مختلف ارائه شده :

Version <sup>[42]</sup>	Release name	Number	Bundled JVM	Year	Release date	Notes
MATLAB 1.0				1984		
MATLAB 2				1986		
MATLAB 3				1987		
MATLAB 3.5				1990		Ran on DOS but needed at least a 386 processor; version 3.5m needed math coprocessor
MATLAB 4				1992		Ran on Macintosh
MATLAB 4.2c				1994		Ran on Windows 3.1x, needed a math coprocessor
MATLAB 5.0	Volume 8			1996	December, 1996	Unified releases across all platforms
MATLAB 5.1	Volume 9			1997	May, 1997	
MATLAB 5.1.1	R9.1					

- 
- 
-

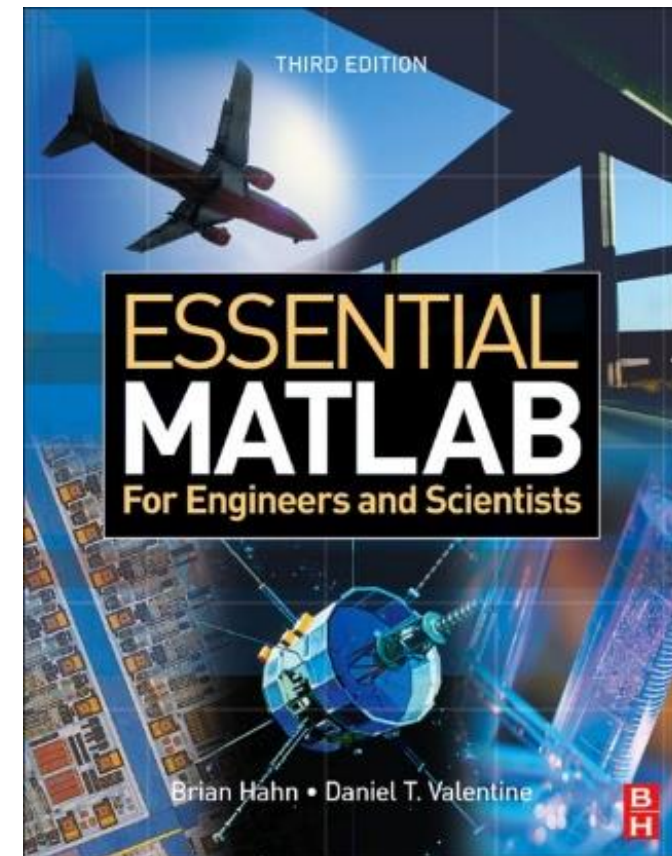
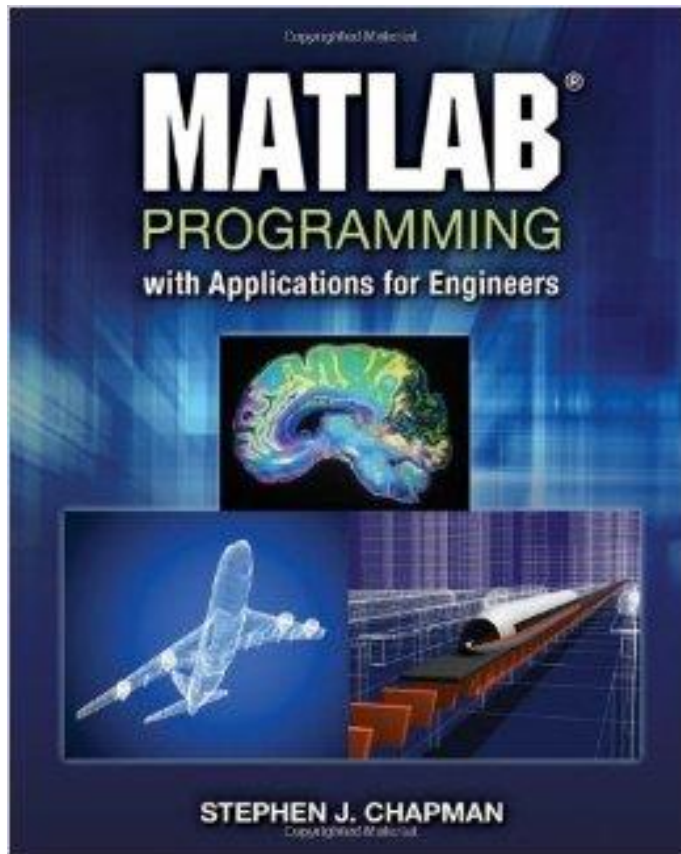


## نسخه های مختلف ارائه شده :



MATLAB 8.4	R2014b	32	1.7.0_11	2014	October 3, 2014	New class-based graphics engine (a.k.a. HG2); <sup>[68]</sup> tabbing function in GUI; <sup>[69]</sup> improved user toolbox packaging and help files; <sup>[70]</sup> new objects for time-date manipulations; <sup>[71]</sup> Git-Subversion integration in IDE; <sup>[72]</sup> big data abilities with MapReduce (scalable to Hadoop); <sup>[73]</sup> new py package for using Python from inside MATLAB, new engine interface to call MATLAB from Python; <sup>[74][75]</sup> several new and improved functions: webread (RESTful web services with JSON/XML support), tcpclient (socket-based connections), histcounts, histogram, animatedline, and others
MATLAB 8.5	R2015a	33	1.7.0_60	2015	March 5, 2015	Last release supporting Windows XP and Windows Vista
MATLAB 8.5	R2015aSP1		1.7.0_60		October 14, 2015	
MATLAB 8.6	R2015b	34	1.7.0_60		September 3, 2015	New MATLAB execution engine (a.k.a. LXE); <sup>[76]</sup> graph and digraph classes to work with graphs and networks; <sup>[77]</sup> MinGW-w64 as supported compiler on Windows; <sup>[78]</sup> Last version with 32-bit support
MATLAB 9.0	R2016a	35	1.7.0_60	2016	March 3, 2016	Live Scripts: interactive documents that combine text, code, and output (in the style of Literate programming); <sup>[79]</sup> App Designer: a new development environment for building apps (with new kind of UI figures, axes, and components); <sup>[80]</sup> pause execution of running programs using a Pause Button
MATLAB 9.1	R2016b	36	1.7.0_60		September 15, 2016	define local functions in scripts; <sup>[81]</sup> automatic expansion of dimensions (previously provided via explicit call to bsxfun); tall arrays for Big data; <sup>[82]</sup> new string type; <sup>[83]</sup> new functions to encode/decode JSON; official MATLAB Engine API for Java <sup>[32]</sup>

کاربرد های نرم افزار متلب:



**AEROSPACE**

**BIOINFORMATICS**

**CONTROL SYSTEM**

**DATABASE**

**ECONOMETRICS**

**FINACIAL**

**DSP SYSTEMS**

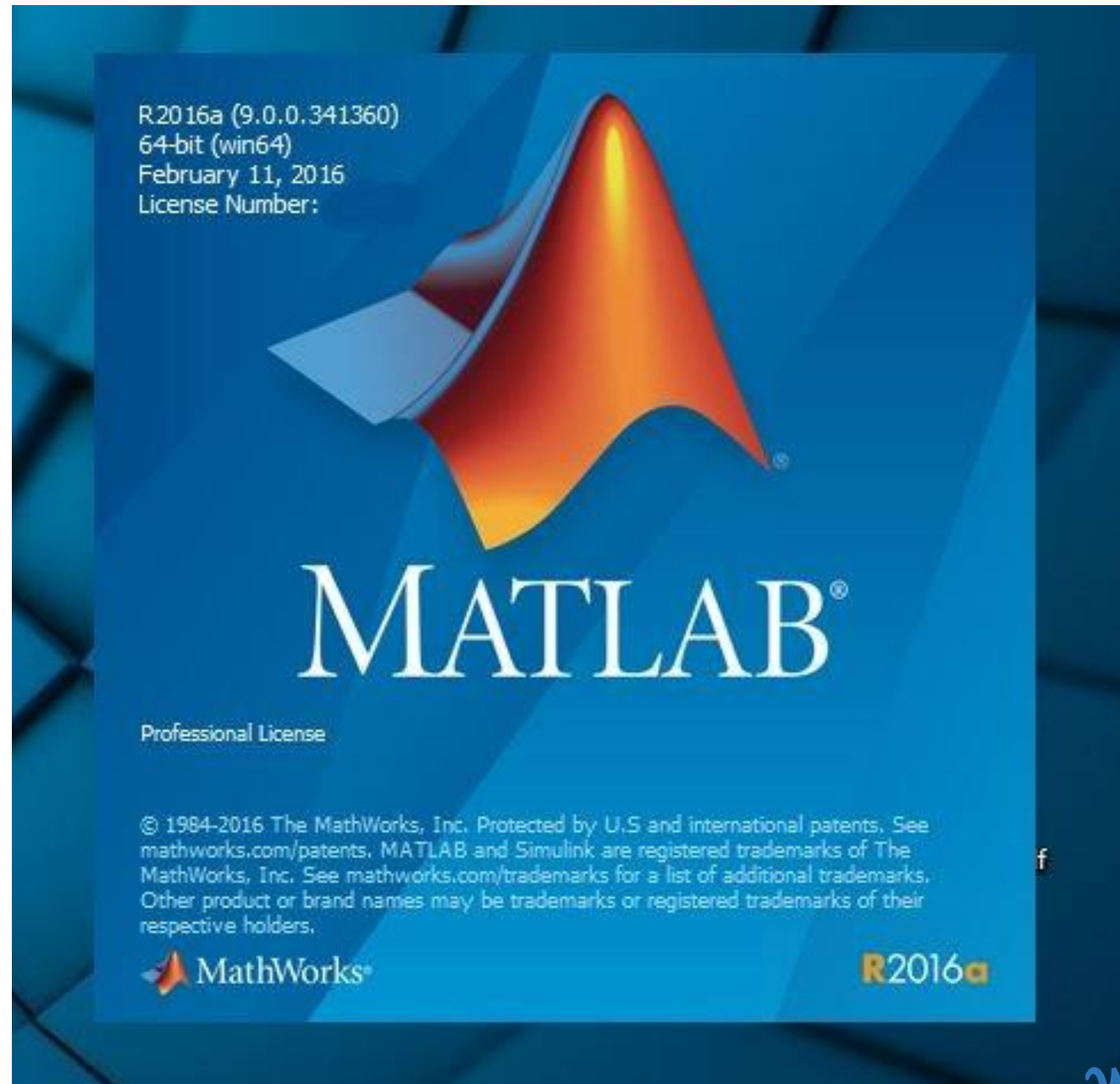
**HDL CODER**

**AEROSPACE**

**SIMULINK**

...





نرم افزار متلب محیطی برای برنامه نویسی سطح بالا جهت حل مسائل علمی و مهندسی به شمار می رود. و امکانات محاسبات متعدد، رسم نمودار و تصاویر و ... را در اختیار کاربر قرار داده و دارای کتابخانه های گسترده ای از توابع مختلف است که هر کدام از این کتابخانه ها برای پردازش و بررسی سیستم ها و داده ها در علوم مختلف طراحی شده اند.

هدف از این توابعی که در این کتابخانه ها هستند این است که به کاربر این اجازه را می دهد که بدون نیاز به نوشتن برنامه های طولانی، انواع مختلفی از شبیه سازی ها و پردازش ها و محاسبات را انجام دهند.

# پنجره های مختلف نرم افزار :

The screenshot shows the MATLAB R2016a desktop environment. The interface includes a top toolbar with tabs for HOME, PLOTS, and APPS. Below the toolbar is a ribbon with various tool groups. The main workspace is divided into several panes:

- Current Folder:** Located on the left, it displays a file explorer view of the current directory. A callout box points to it with the text: **دایرکتوری جاری** (مسیر ذخیره سازی فایل های تولید شده).
- Command Window:** Located in the center, it contains the MATLAB command prompt. A callout box points to it with the text: **پنجره ی فرمان**.
- Workspace:** Located on the right, it displays a table of variables in the workspace. A callout box points to it with the text: **پنجره ی فضای کاری** (ذخیره سازی متغیرها).
- Command History:** Located at the bottom right, it displays a list of commands entered in the Command Window. A callout box points to it with the text: **پنجره ی تاریخچه ی دستورات** (ذخیره سازی دستورات اجرا شده در پنجره ی فرمان).

**پنجره فرمان (Command Window)** : این پنجره مکانی است که کاربر باید دستورات مورد نظر را درون آن تایپ کند. مکان تایپ با دو علامت >> مشخص شده است و مکان نما در جلوی این علامت >> در حال چشمک زدن است که نشان می دهد ، نرم افزار آماده دریافت دستور جدید است.

**پنجره فضای کاری (Workspace Window)** : در این فضا تمامی متغیر هایی که در برنامه توسط دستورات ایجاد شده اند نشان داده می شود. در متلب متغیر ها را می توان به صورت برداری، ماتریسی تعریف نمود.



✓ برای اجرای هر دستور در **MATLAB** باید کلید **ENTER** پس از نوشتن دستور در پنجره ی **Command** (فرمان) زده شود .

✓ اگر نقطه ویرگول (**;**) در پایان هر فرمانی تایپ شود، خروجی فرمان نمایش داده نمی شود .

✓ اگر در شروع دستوری و یا هر کجا از کدمان علامت درصد (**%.)** استفاده شود ، آن خط یک عبارت توضیحی تلقی می گردد. لذا حتی در هنگام اجرا و یا **ENTER** زدن آن خط اجرا نمی شود و به اصطلاح به آن **(Comment)** می گویند .

✓ برای پاک کردن محتویات پنجره فرمان **(Command)** از دستور **clc** استفاده می کنیم .

✓ برای لغو اجرای فرمانی دکمه **ctrl + c** را فشار می دهیم.

help

با استفاده از **Help** نرم افزار متلب، می توان اطلاعات تکمیلی در مورد تابعی دلخواه بدست آورد برای اینکار باید در پنجره ی **Command** بصورت زیر عمل کنیم:

**help** Function Name

علامت (...)

در برخی موارد اگر خطوط کدمان بسیار طولانی باشد و ناچارا باید ادامه کد را در خط بعدی بنویسیم از علامت (...) در انتهای خط قبلی استفاده می نمایم.

نام دلخواه متغیرها

- نام متغیرها می تواند ترکیبی از حروف و اعداد باشد.
- طول نام متغیرها نباید از ۶۳ کاراکتر بیشتر باشد .
- متلب به بزرگی و کوچکی حروف حساس است.
- نباید از نام های توابع داخلی خود متلب استفاده کرد.(مثلا :  
(sin , cos , ...
- اسامی متغیرها حتما باید با یک حرف شروع شوند ، و پس  
از آن از اعداد و حروف دیگر و یا Under line استفاده کرد .

## نمادهای عملیات محاسباتی

جمع	+
تفریق	-
ضرب	*
تقسیم از راست	/
تقسیم از چپ	\
توان	^



نرم افزار متلب دارای برخی متغیرهای از پیش تعریف شده ای می باشد.

- از متغیرهای پیش فرض ،به عنوان یک متغیر جدید نباید استفاده کرد چون مقدار جدید جایگزین مقدار قبلی می گردد .

متغیرهای پیش فرض

ans	نمایش مقدار محاسبه شده به وسیله ی یک عبارت
pi	نمایش عدد ۳.۱۴
eps	کوچکترین مقداری که وقتی به عددی اضافه شود عددی بزرگتر از آن تولید کند.
inf	نمایش بی نهایت
i	به صورت رادیکال 1- نیز می توان بیان نمود
j	همانند i
NaN	هیچ عددی برای آن قابل تعریف نیست. (۰/۰)
clock	نمایش زمان جاری (سال،روز،ساعت)
date	نمایش تاریخ جاری در قالب رشته ی کاراکتری

## دستورات مدیریت متغیرها

clear	حذف تمامی متغیرها از حافظه
clear x,y,z	حذف فقط متغیرهای x,y,z
who	نمایش متغیرهای جاری در Workspace
whos	نمایش لیستی از متغیرهای جاری به همراه اطلاعات تکمیلی

## توابع مقدماتی خود نرم افزار متلب

abs(x)	محاسبه قدر مطلق
sqrt(x)	محاسبه ریشه دوم
round(x)	گرد کردن به نزدیک ترین عدد صحیح
fix(x)	گرد کردن به نزدیک ترین عدد صحیح به سمت صفر
floor(x)	گرد کردن به نزدیکترین عدد صحیح به سمت منفی بی نهایت
ceil(x)	گرد کردن به نزدیک ترین عدد صحیح به سمت مثبت بی نهایت
sign(x)	علامت یک متغیر را مشخص می کند (سه مقدار ممکن: ۱- و ۱+ و ۰)
rem(x,y)	باقیمانده تقسیم $x$ بر $y$
log(x)	محاسبه $\ln x$
log10(x)	محاسبه $\log x$ در مبنای ۱۰
exp(x)	محاسبه $e^x$ ( $e=2.7183$ )

## توابع مقدماتی خود نرم افزار متلب

$\sin(x)$	محاسبه ی سینوس
$\cos(x)$	محاسبه ی کسینوس
$\tan(x)$	محاسبه ی تانژانت
$\text{asin}(x)$	معکوس سینوس
$\text{acos}(x)$	معکوس کسینوس
$\text{atan}(x)$	تانژانت معکوس
$\text{atan2}(y,x)$	تانژانت معکوس $x/y$
$\sinh(x)$	سینوس هیپربولیک
$\cosh(x)$	کسینوس هیپربولیک
$\tanh(x)$	تانژانت هیپربولیک
$\text{asinh}(x)$	معکوس سینوس هیپربولیک



## عملگرهای مختلط

conj(x)	محاسبه مزدوج مختلط متغیر x $x = a + ib \rightarrow x = a - ib$
real(x)	مقدار حقیقی عدد مختلط
imag(x)	مقدار موهومی عدد مختلط
abs(x)	قدر مطلق اندازه ی عدد مختلط
angle(x)	زاویه یک عدد مختلط $Q = \text{atan2}(\text{imag}(x), \text{real}(x))$
complex(a,b)	ایجاد یک عدد مختلط با قسمت حقیقی a و بخش موهومی b

**آرایه ها (Arrays):** لیستی از اعداد مرتب شده در سطرها و یا ستون ها است .

**آرایه یک بعدی :** دارای یک سطر یا یک ستون از اعداد

**آرایه دو بعدی :** دارای **یک** سطر و **یک** ستون از اعداد ← **ماتریس**

**بردار سطری :** در این حالت عناصر با کاما و یا فضای خالی (space) از هم جدا می شوند :

**$X=[1 \ 5 \ 6 \ 8] \quad == \quad X=[1,5,6,8]$**

**بردار ستونی :** در این حالت عناصر با سمیکالن از هم جدا می شوند :

**$X=[1$**

**$5$**

**$== \quad X=[1;5;6;8]$**

**$6$**

**$8]$**

## سه تابع معروف برای اینکار وجود دارد :

$X = a : c : b$	مقادیر بین <b>a</b> و <b>b</b> را با گام <b>c</b> ایجاد می کند .
$X = \text{inspace}(a, b, c)$	مقادیر بین <b>a</b> و <b>b</b> را به <b>c</b> قسمت تقسیم می کند.
$X = \text{logspace}(a, b, c)$	ایجاد آرایه ی لگاریتمی مثل تابع بالا ولی خروجی آن برپایه $10^k$ بوده که <b>k</b> خروجی بخش شبه <b>inspace</b> می باشد.

$X = (0 : 0.1 : 4)$

$X = \text{inspace}(0, 4, 40)$

$X = \text{logspace}(0, 4, 40)$

**ماتریس (Matrix):** آرایه ای دو بعدی از اعداد در سطر و ستون می باشد .

✓ ماتریس از چندین سطر متوالی تشکیل شده که مقادیر هر سطر با سطر بعدی بوسیله سمیکالن از هم جدا شده اند .

✓ عناصر ماتریس ممکن است اعداد حقیقی یا اعداد مختلط باشند .

$$X=[1 \ 2 \ 3 ; 4 \ 5 \ 6]$$

$$X=[3*\sin(z)+7 \ 8 ; 3i \ 5i-1]$$



## آدرس دهی آرایه ها

1	4	7
2	5	8
3	6	9

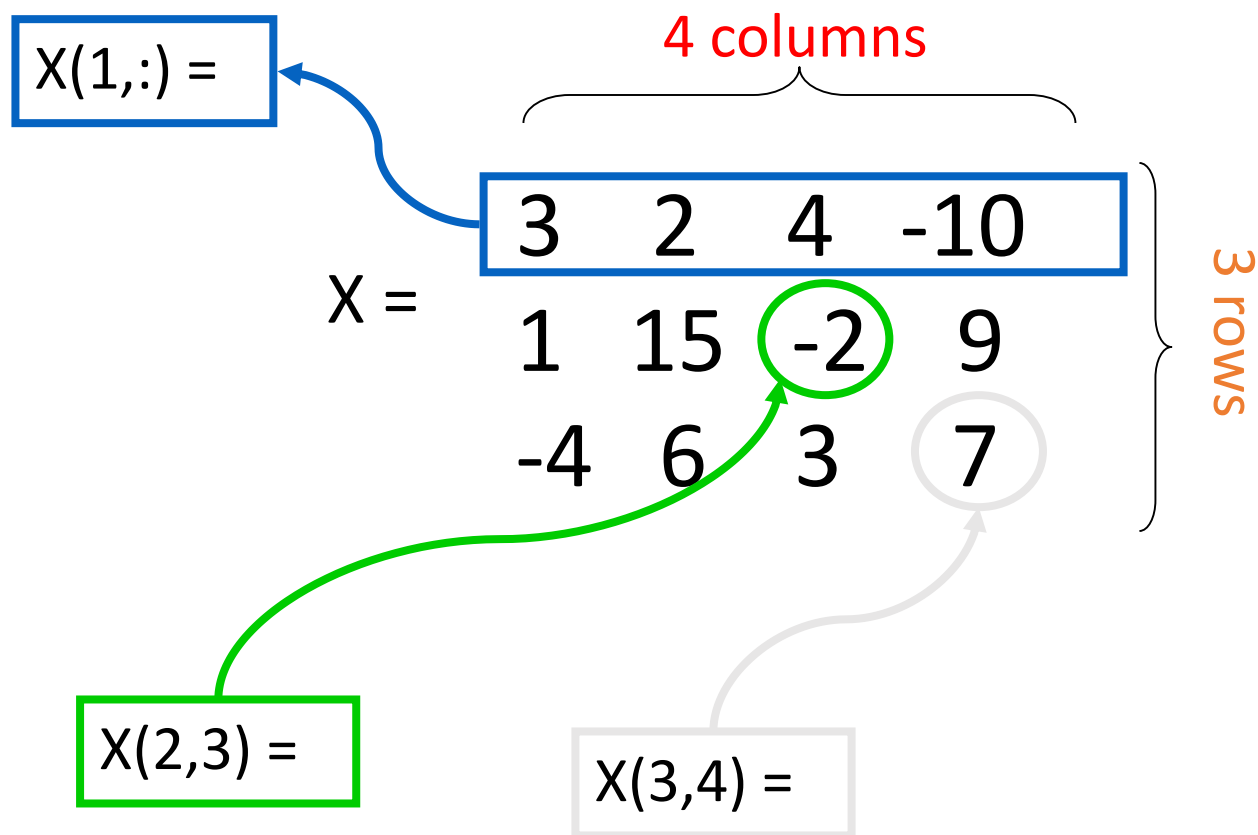


1, 1	1, 2	1, 3
2, 1	2, 2	2, 3
3, 1	3, 2	3, 3

$X(3) = 3$  اولین عنصر سطر سوم

$X(3,2) = 6$  سطر سوم، ستون دوم

## دسترسی به المان ها



## آدرس دهی به طیف وسیعی از عناصر یک ماتریس

$X(:)$	به تمام عناصر بردار $X$ اشاره دارد .
$X(m : n)$	المان های $m$ تا $n$ ام بردار $X$
$X(m : a : n)$	المان های $m$ تا $n$ ام بردار $X$ با طول گام $a$

$X=[1 \ 5 \ 6 ; 7 \ 8 \ 10]$

$X(1:2) = 1 \ , \ 7$

$X(1,:) = 1 \ 5 \ 6$

باید یک آرایه یک بعدی تعریف کنیم نه ماتریس

`values = 10:-1:2`

`values = [10, 9, 8, 7, 6, 5, 4, 3, 2]`

## دسترسی به المان ها

با استفاده از بردارهایی از اندیسهای سطرها و ستونها:

$$A([2\ 3],[1\ 2]) \longleftrightarrow A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & -5 & 6 \\ 5 & 6 & 7 \end{bmatrix}$$

$$B=A([3\ 2],[2\ 1])$$
$$B=[A(3,2),A(3,1);A(2,2);A(2,1)] \longleftrightarrow B = \begin{bmatrix} 6 & 5 \\ -5 & 4 \end{bmatrix}$$

## دسترسی به المان ها

دسترسی به یک سطر یا ستون کامل :

$$A(1,:) \longleftrightarrow A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & -5 & 6 \\ 5 & 6 & 7 \end{bmatrix}$$

دسترسی به قسمتی از یک سطر یا ستون :

$$\begin{array}{l} A(1:2,:) \\ A([1 \ 2],:) \end{array} \longleftrightarrow A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & -5 & 6 \\ 5 & 6 & 7 \end{bmatrix}$$

$$v=1:5 \longleftrightarrow v = [1 \ 2 \ 3 \ 4 \ 5]$$

$$w=1:2:5 \longleftrightarrow w = [1 \ 3 \ 5]$$



## دسترسی به المان ها در ماتریس ها

$X( :, n )$	به تمام عناصر ستون $n$ ام از ماتریس $X$
$X( n, : )$	تمام عناصر در سطر $n$ ام
$X( :, m : n )$	عناصر ستون های $m$ تا $n$ ماتریس
$X( m : n, : )$	عناصر سطرهای $m$ تا $n$
$X( m : n, p : q )$	عناصر مشترک سطرهای $m$ تا $n$ و ستون های $p$ تا $q$ ماتریس

`plot (x , y , ' Style' )`

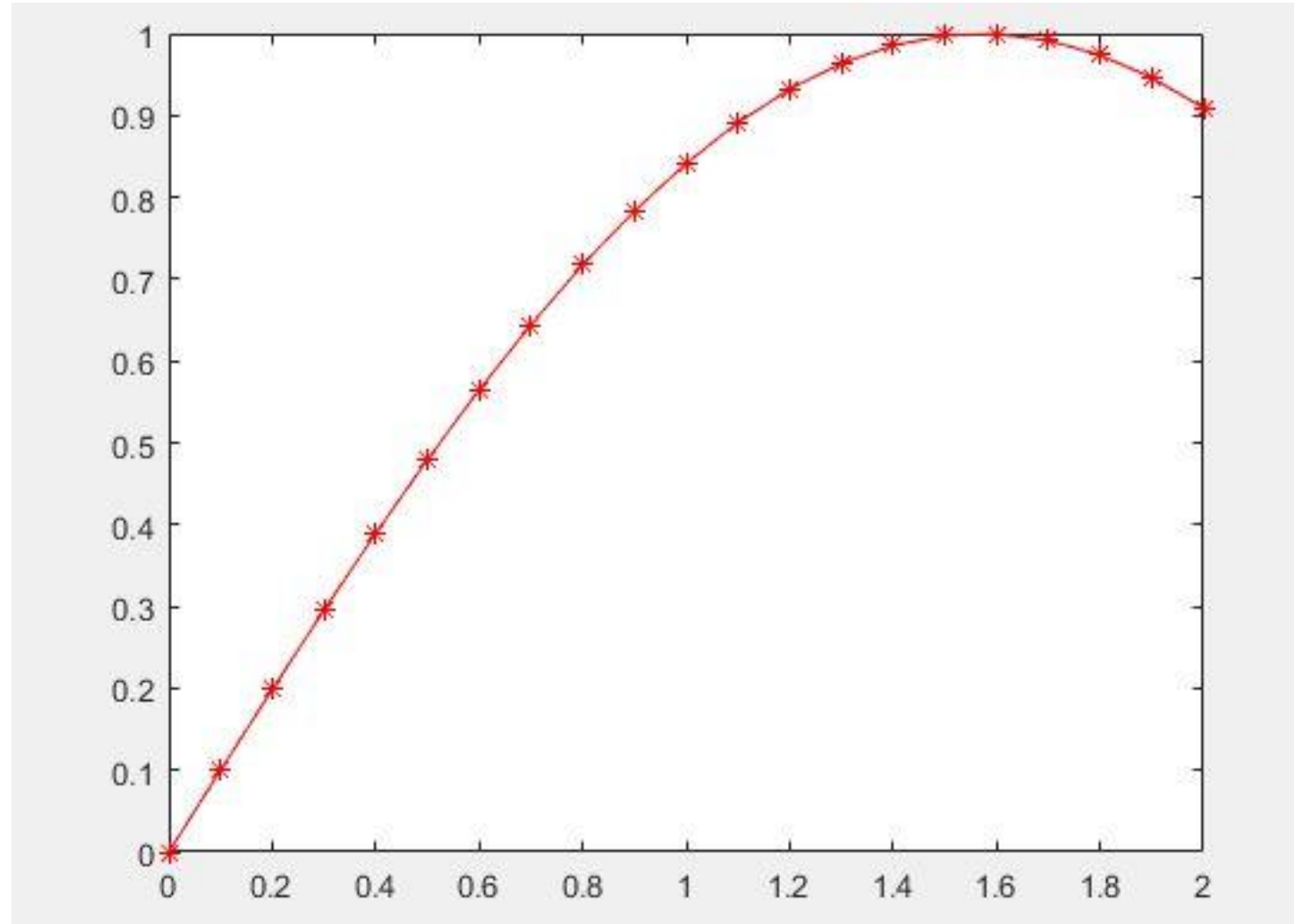
که **x** و **y** بردارهای شامل مختصات نقطه‌ای می باشد .

فرمان اصلی برای  
تولید رسم دو بعدی

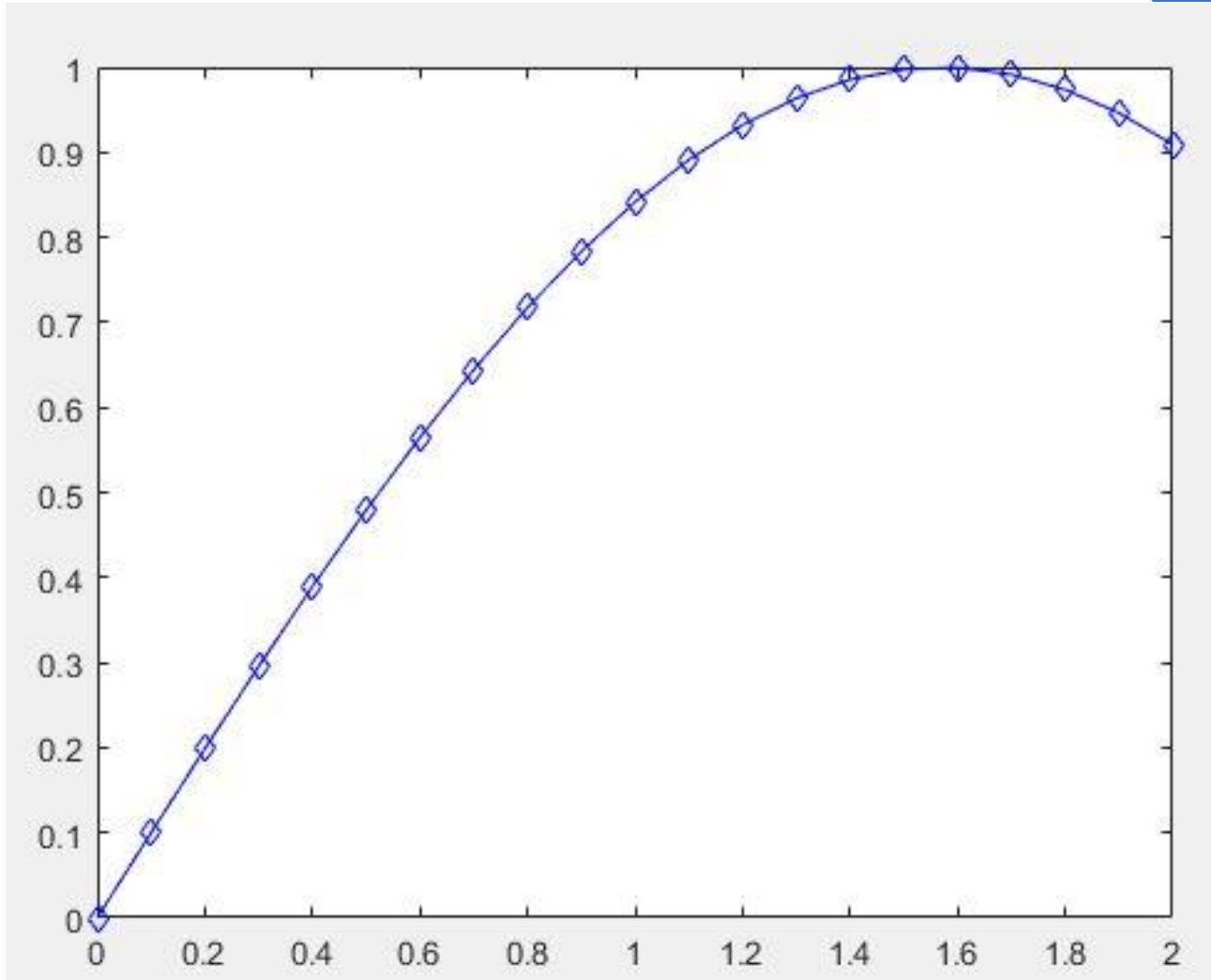
- ✓ شکل های ترسیم شده در پنجره ای به نام **figure** نمایش داده می شوند ، هر بار استفاده از این دستور سبب خواهد شد تا شکل قبلی پاک شده و شکل جدید جایگزین شکل قبلی گردد .
- ✓ انتخاب **style** شامل یک سری آرگومان های اختیاری مانند رنگ ، نوع خط و نوع علامت است .

انتخاب نوع علامت		انتخاب نوع خط		انتخاب نوع رنگ	
علامت جمع	+	توپر	-	زرد	y
دایره	O	خط تیره	--	ارغوانی	m
نشان ستاره دار	*	۲ نقطه	:	فیروزه ای	c
علامت ضرب	x	خط نقطه	-.	قرمز	r
علامت نقطه	.			سبز	g
انواع مثلث	^v<>			آبی	b
مربع	s			سفید	w
لوزی	d			سیاه	k
ستاره پنج پر	p				
ستاره شش پر	h				

```
x=(1 : 0.1 : 2)  
y=sin(x)  
plot(X , Y , ' -r* ' )
```

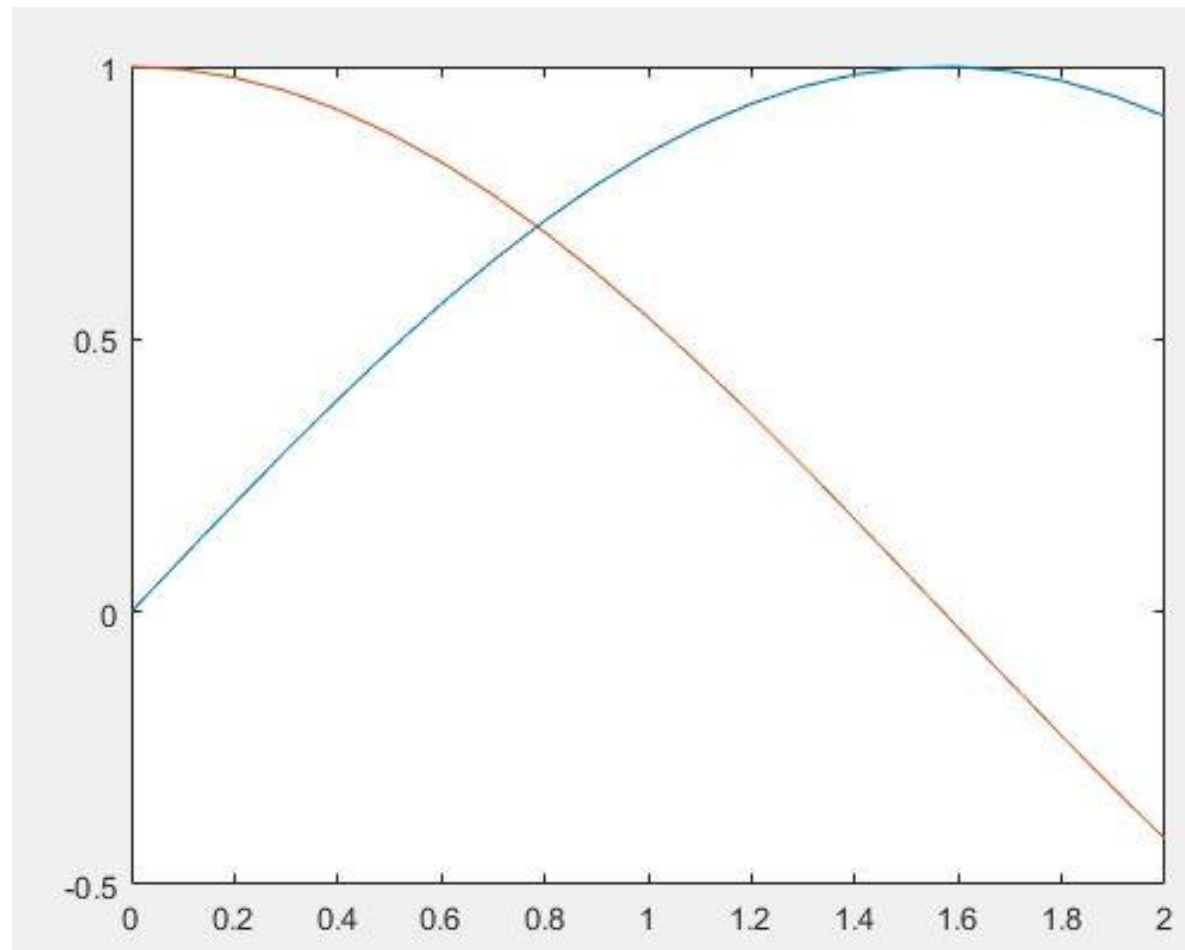


```
x=(1 : 0.1 : 2)  
y=sin(x)  
plot(X , Y , ' -bd ' )
```



## رسم چند ضابطه در یک پنجره Figure

```
x=(1 : 0.1 : 2)  
y=sin(x)  
z=cos(x)  
plot(x , y , x , z )
```



به کمک دستور plot



## رسم چند ضابطه در یک پنجره Figure

```
hold on
```

```
x=(1 : 0.1 : 2)
```

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

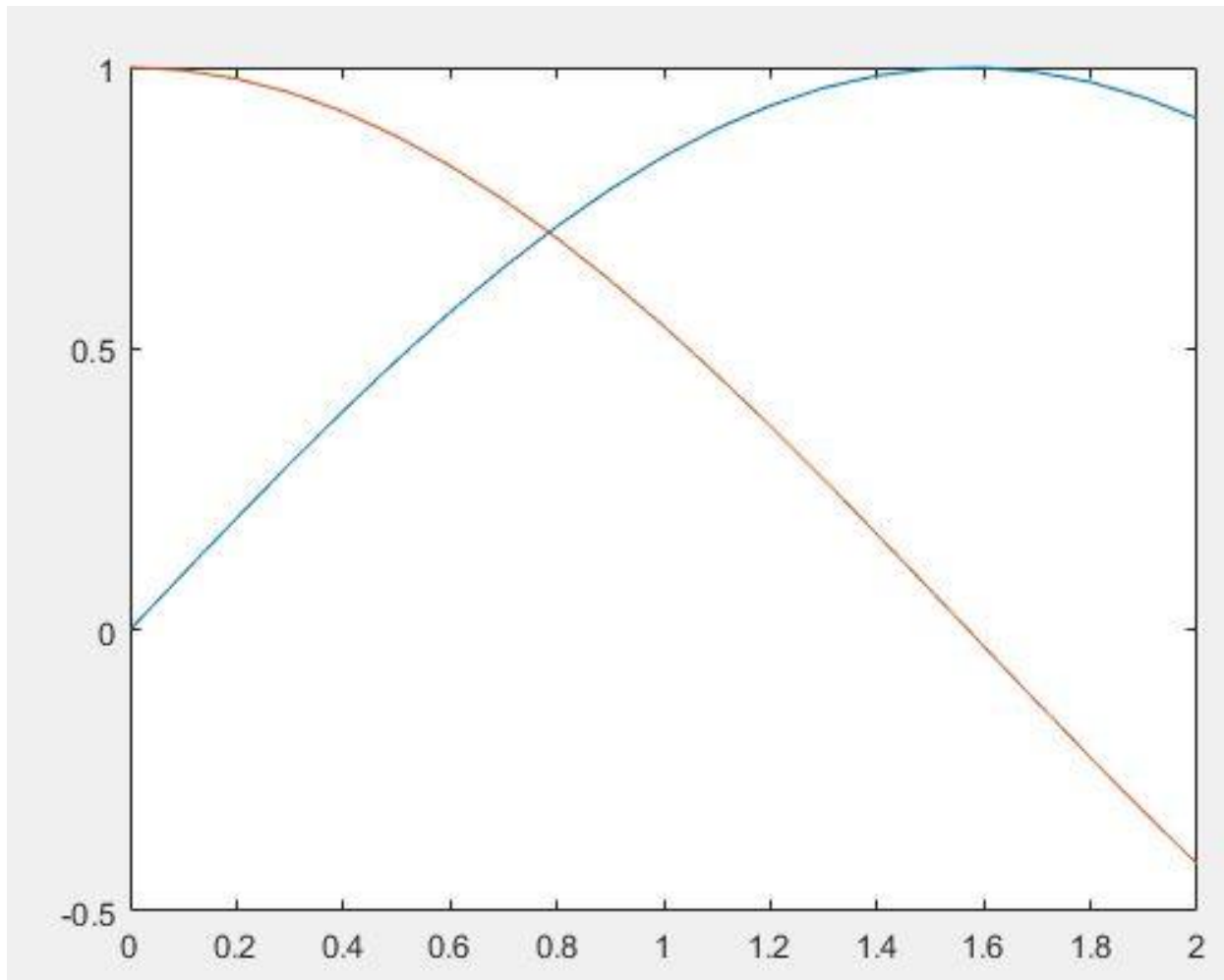
```
z=cos(x)
```

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

```
plot(x , z )
```

```
hold off
```

به کمک دستور  
hold



## رسم چند ضابطه در چند پنجره

```
x=(1 : 0.1 : 2)
```

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

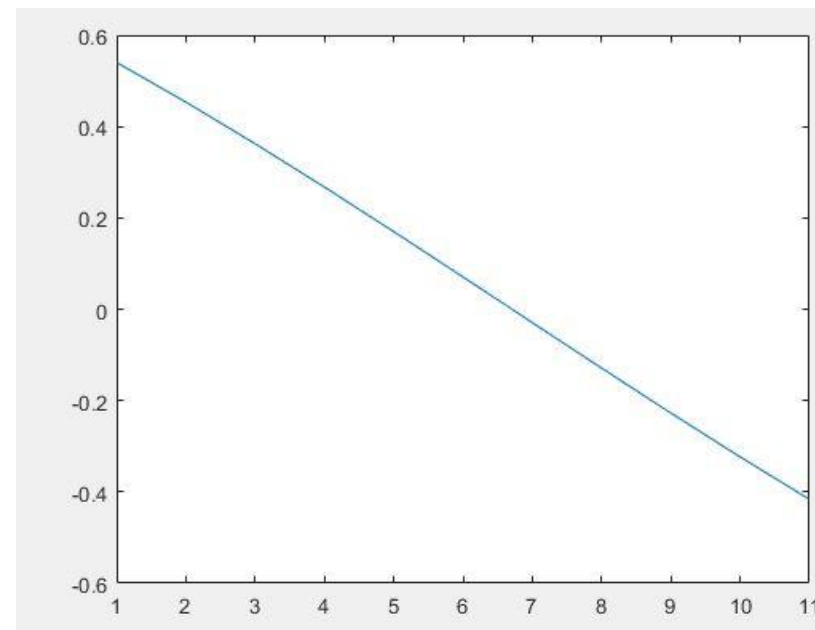
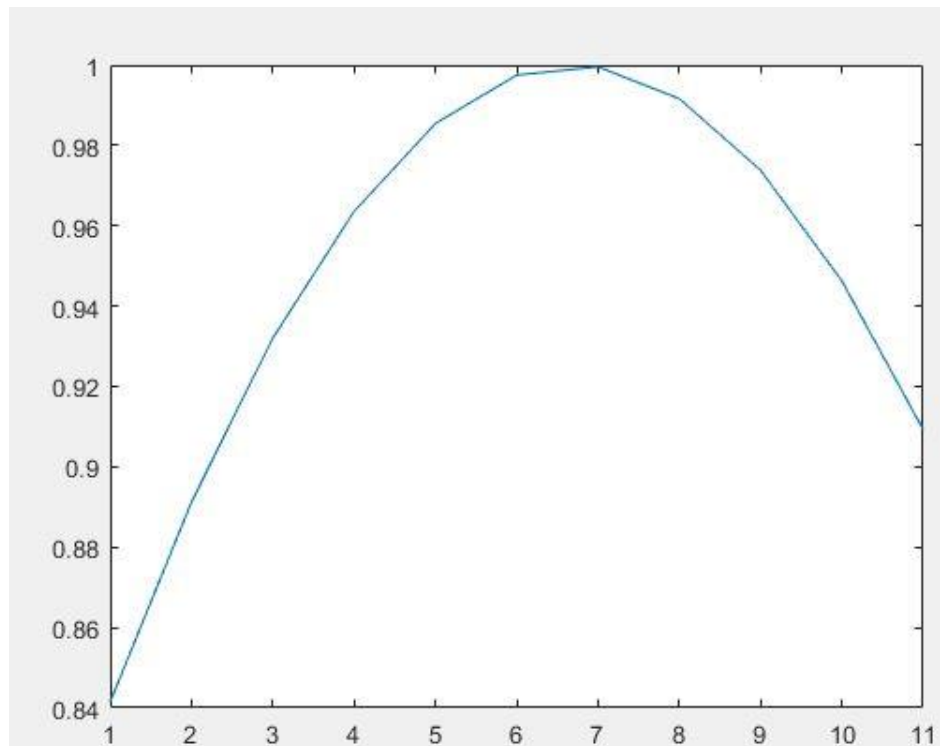
```
z=cos(x)
```

```
figure(1)
```

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

```
figure(2)
```

```
plot(x , z )
```



به کمک دستور  
figure

## برخی از دستورات پر کاربرد در رسم نمودار

title	قرار دادن عنوان برای شکل
xlabel	برچسب زدن روی محور افقی
ylabel	برچسب زدن روی محور عمودی
grid on	فعال شدن خطوط شبکه برای نمودار جاری
grid off	غیر فعال شدن خطوط شبکه برای نمودار جاری
box on	فعال شدن خطوطی که محورها را در بر گرفته اند
box off	غیر فعال شدن خطوطی که محورها را در بر گرفته اند
Text	می توان متن یا برچسب را در محل مورد نظر خود، روی سطح نمودار قرار داد مختصات گوشه ی سمت چپ مرکزی متن = $x, y$ $\text{Text}(x, y, 'string')$

<code>gtext('string')</code>	می توان با دستور <code>gtext</code> مختصات روی نمودار را بدست آورد و متنی را در داخل آن نوشت. برای اینکار کافی است در محل مورد دلخواه کلیک کنیم.
<code>axis on</code>	تمام حالت‌های مربوط به برچسب گذاری، علائم و پس زمینه محورهای مختصات را فعال می کند.
<code>axis off</code>	تمام حالت‌های مربوط به برچسب گذاری، علائم و پس زمینه محورهای مختصات را غیرفعال می کند.
<code>hold on</code>	نمودارهای جاری پاک نمی شوند و باعث می شود شکل های بعدی روی هم بیفتند
<code>hold off</code>	لغو حالت <code>hold on</code>
<code>close all</code>	بستن تمام پنجره های Figure باز شده
<code>clf</code>	محتویات پنجره ی Figure جاری را پاک می کند
<code>clf reset</code>	همان کار <code>clf</code> را تمجام می دهد، همچنین تنظیمات را به حالت اولیه باز می گرداند

ایجاد راهنما بر روی نمودار	Legend('string1' , 'string2' , 'string3', ... , n)
n محل قرارگیری راهنما بر روی پنجره figure	

محل قرارگیری راهنما	کد محل قرارگیری
شمال شرق و خارج از محورها	-1
داخل محورها (حالت بهینه)	0
شمال شرق	1
شمال غرب	2
جنوب غرب	3
جنوب شرق	4

این دستور پنجره ی figure را همانند یک ماتریس  $m * n$  ، به  $m$  سطر و  $n$  ستون تقسیم می کند و ناحیه ی  $p$  ام را به عنوان ناحیه ی فعال ، جهت رسم ، انتخاب می کند.

`subplot(m , n , p)`

subplot

```
x=linspace(0 , 3,100);  
y=sin(pi *x);  
subplot(2,2,1);  
plot(x,y)  
Z=cos(pi*x);  
subplot(2,2,3:4);  
plot(x,z)
```

