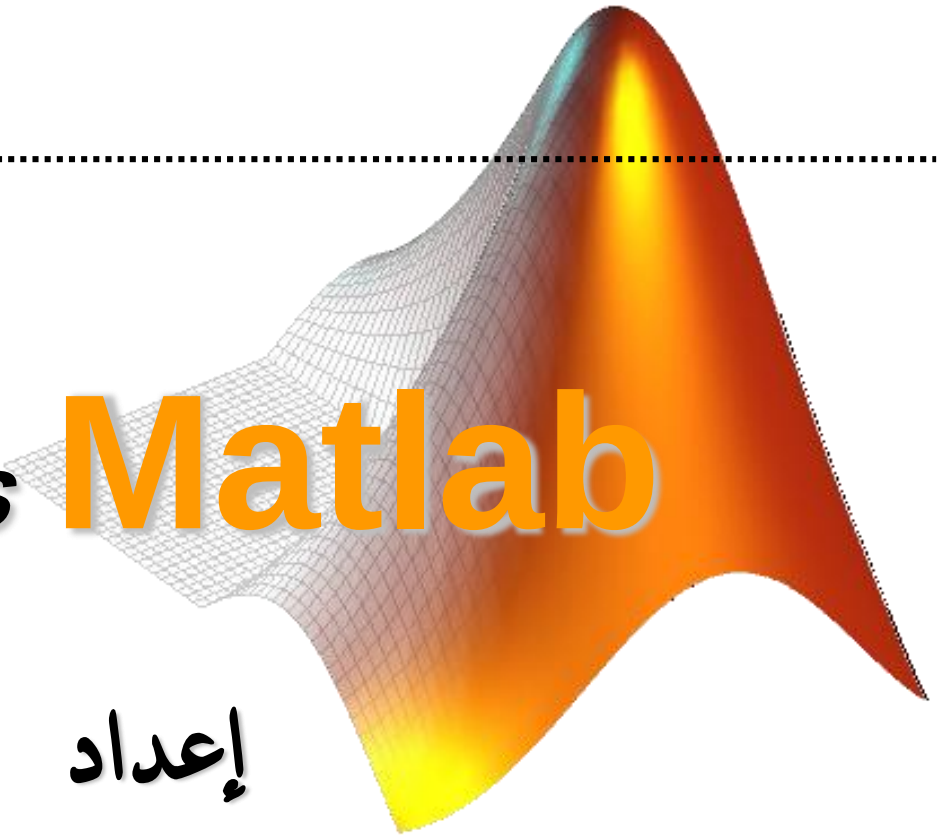
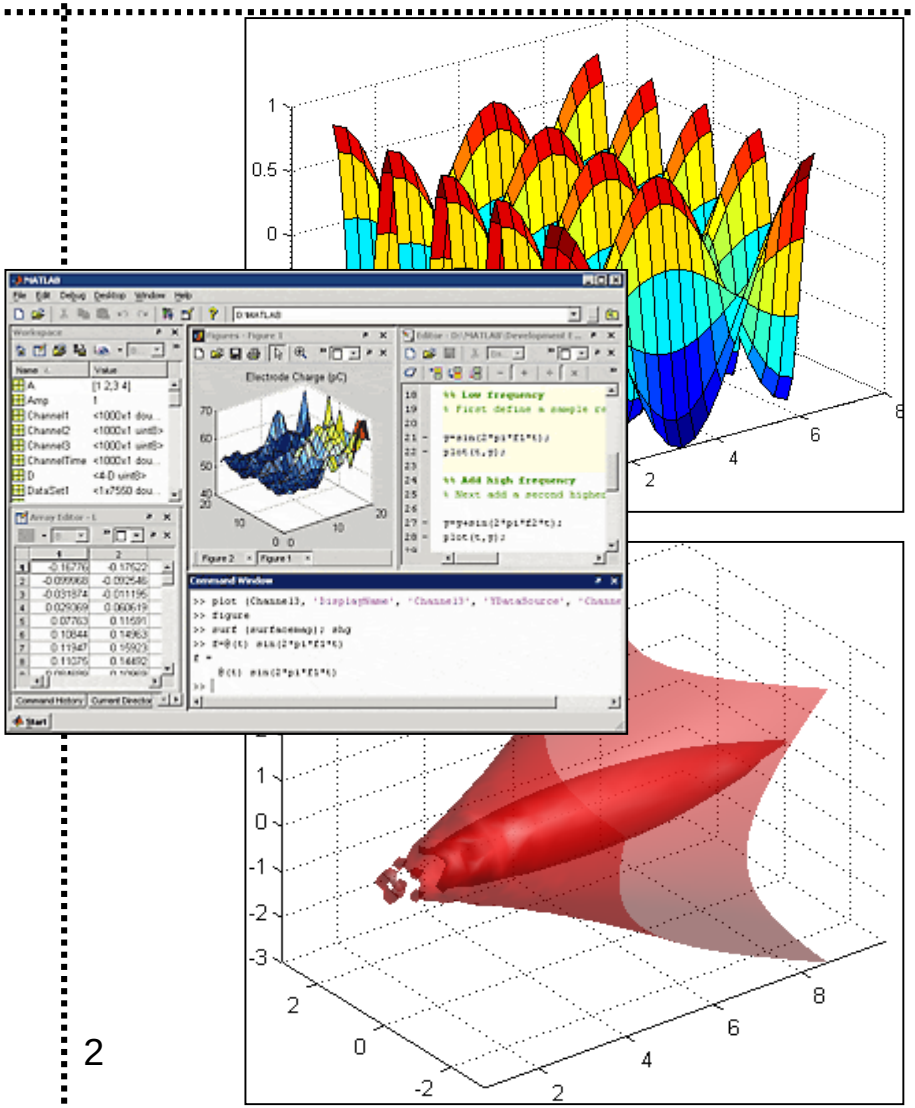


Mathworks Matlab

إعداد

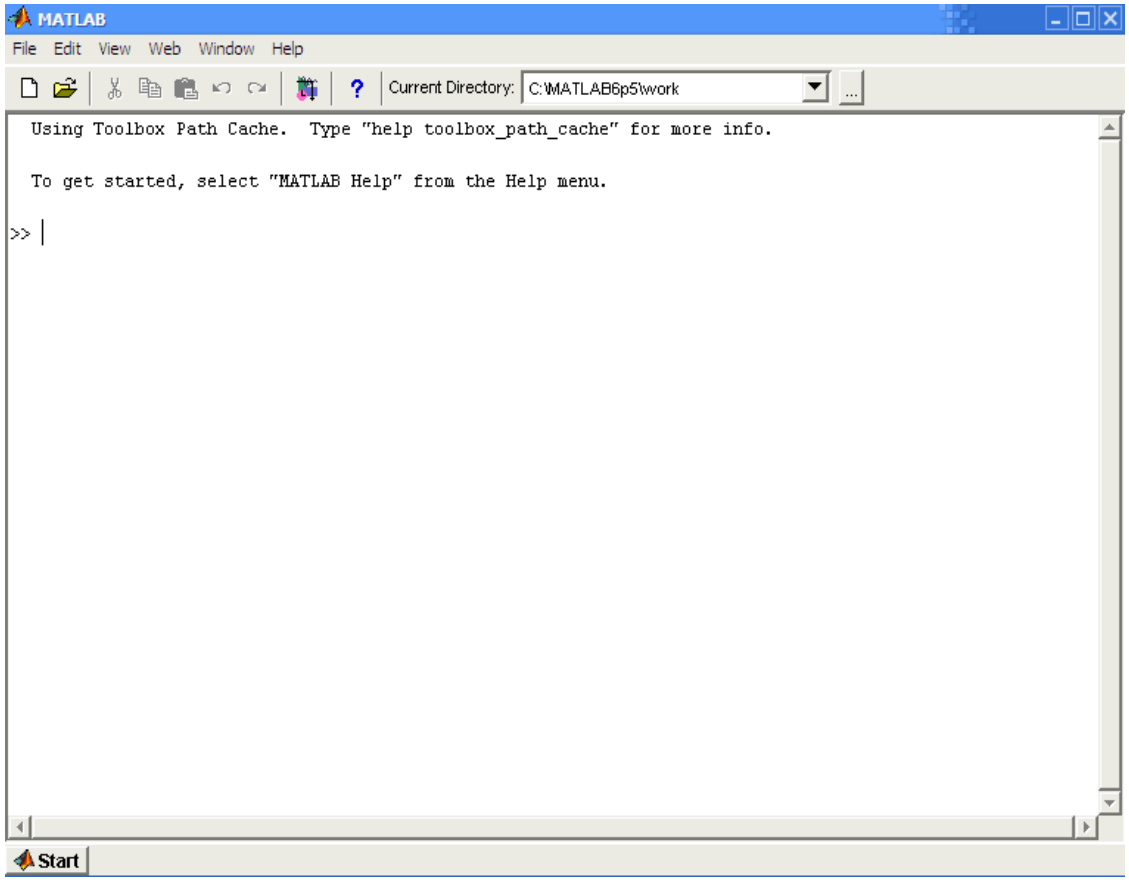
م.يوسف دعبول

ما هو Matlab؟



- الاسم مشتق من Matrix Laboratory
- برنامج ولغة لتطبيق العمليات الرياضية للحسابات العلمية والهندسية
- مصمم لعمليات المصفوفات
- صمم في بداية السبيعنات
- مشهور بقدراته الخاصة بالرسومات والأدوات المختلفة التطبيقات

استخدام Matlab



- بكتابة الأوامر عن طريق نافذة الأوامر (>>)
- بتنفيذ M-files
- باستخدام Simulink

Matlab نافذة

Matlab ذاكرة

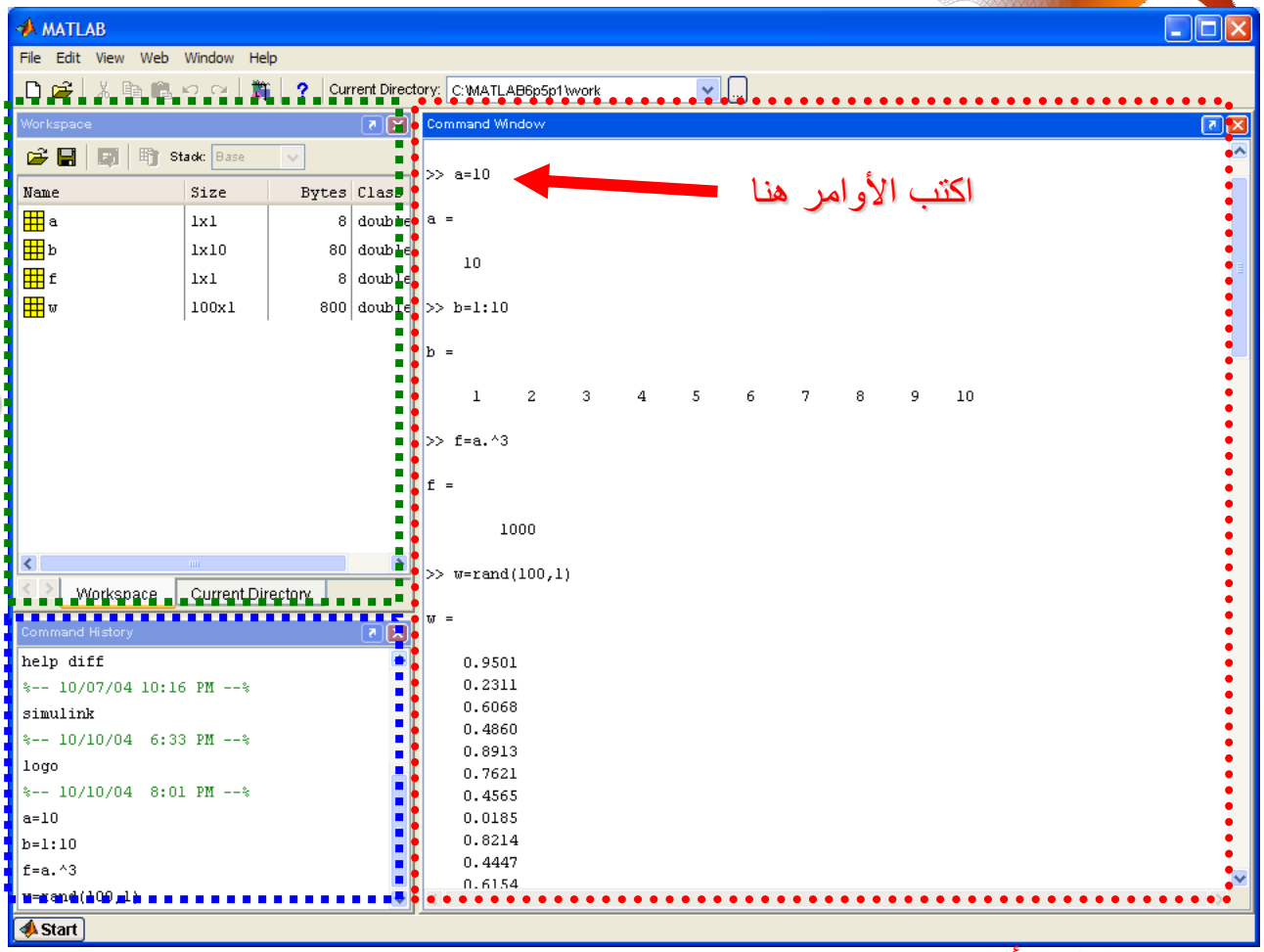
Workspace
Variables/Current Directory

(اصدارات Matlab الجديدة)

الأوامر المدخلة سابقاً

Command History

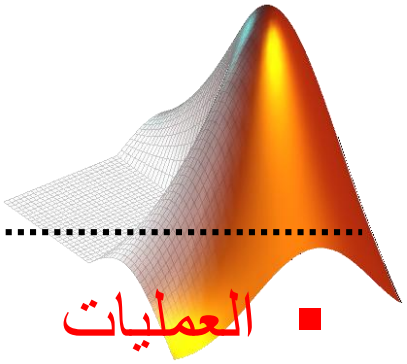
(اصدارات Matlab الجديدة)



نافذة الأوامر Command Window

(all Matlab versions)

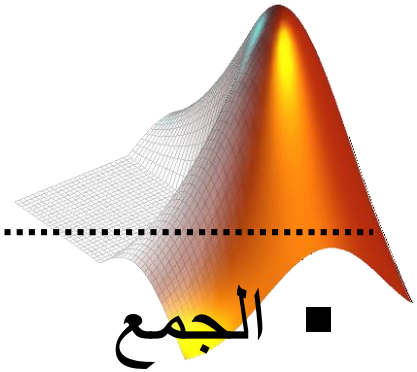
Matlab أجزاء



- Operators
- Matrices
- Plotting
- M-files
- Flow control

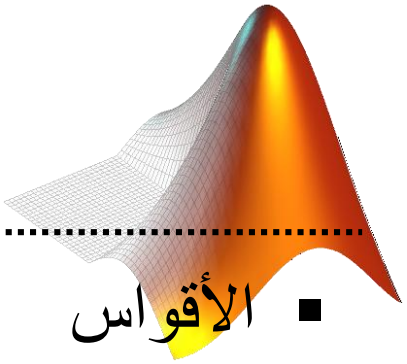
- العمليات
- المصفوفات
- الرسومات
- الملفات
- التحكم بالتنفيذ

العمليات الحسابية البسيطة



▪ الجمع	**+**	a+b
▪ الطرح	**-**	a-b
▪ الضرب	*****	a*b
▪ القسمة (من اليسار)	**/**	a/b
▪ القسمة (من اليمين)	****	a\b (b÷a)
▪ الأس	**^**	a^b

أولوية العمليات

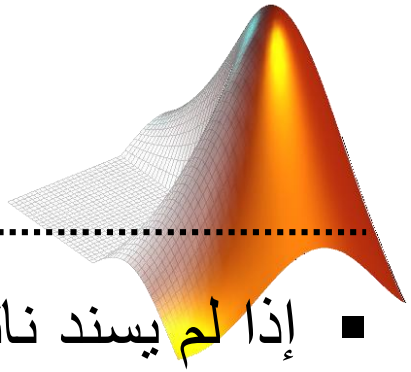


- الأقواس ( )
- الأس ^ L➔R
- الضرب/القسمة * / L➔R
- الجمع/الطرح + - L➔R

- **مثال**

```
>> 5/2*3+4
ans=
11.5
```

نتائج العمليات

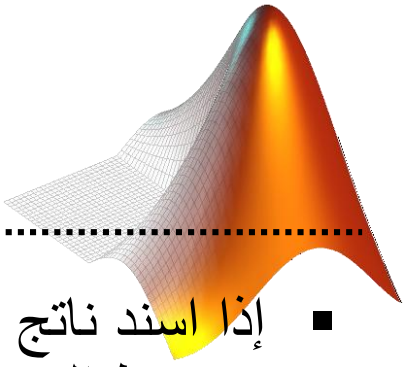


- إذا لم يسند ناتج العملية الحسابية إلى متغير، يقوم *Matlab* باسنادها الى المتغير ans

- لمعرفة ناتج ans، اكتب
>>ans

```
>> 6*5
ans =
30
>> 17/5
ans =
3.4000
>> ans
ans =
3.4000
```

اسناد النتائج



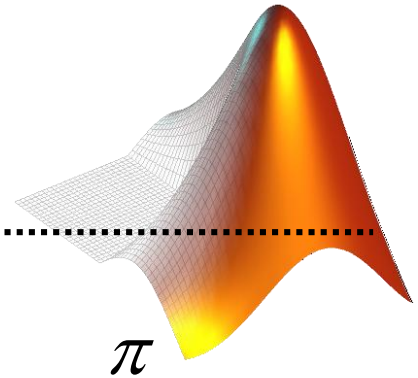
```
>> x=6*5
x=
     30
>> y=17/5;
>> y
y=
    3.4000
>> z=x+y
z=
    33.4000
>> who
Your variables are:
x    y    z
```

- إذا اسند ناتج العملية إلى متغير، سيحتفظ المتغير بالقيمة الأخيرة إلى أن يتم تعديلها
- استخدام الفاصلة المنقوطة (;) يمنع من إظهار ناتج العملية، بينما عدم استخدامها يظهر الناتج
- يفضل اسناد النواتج الى متغيرات
- من الممكن كتابة عدة أوامر بسطر واحد:

```
>> x=6*5;y=17/5;z=x+y
```

- لعرض قائمة المتغيرات بالذاكرة اكتب >>who

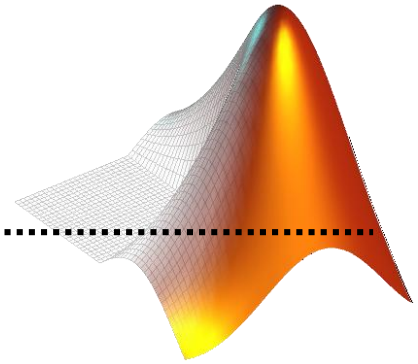
بعض دوال Matlab



- `exp(x)` e^x
- `sin(x)` $\sin x$
- `asin(x)` $\sin^{-1} x$
- `log(x)` $\ln x$
- `log10(x)` $\log x$
- `sqrt(x)` $\sqrt{x}$
- `abs(x)` $|x|$

- `pi` π
- `i,j` imaginary unit
- `NaN` Not-a-Number
- `Inf` ∞

عمليات Matlab

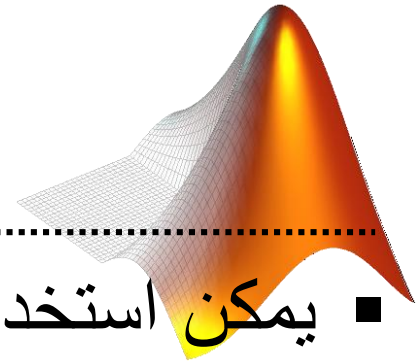


- مثال: يمكن حساب الجذر التربيعي لرقم بعدة طرق:

$$y = \sqrt{x}$$

```
>>y=x^0.5;
>>y=x^(1/2);
>>y=sqrt(x);
```

أسماء المتغيرات



- يمكن استخدام الحروف، والأرقام، و _ لتسمية المتغيرات:

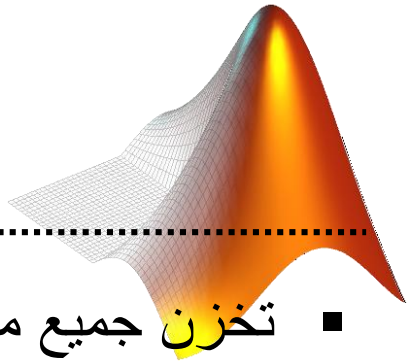
```
>>case_34=6.45;
>>case34=6.45;
```

- لا بد من بدء اسم المتغير بحرف، ولا يمكن وضع مسافات أو علامات أخرى ضمن اسم المتغير :

34case, ge204!, Name Here ✘

case34, ge204, Name_Here ✓

متغيرات Matlab



- تخزن جميع متغيرات كأرقام حقيقية وبحجم ٨ بايت لكل قيمة
- لا فرق بين الأرقام الصحيحة والحقيقية:

```
>> p = 3;
>> p = 3.0;
```

نفس الأمر!

- يمكن تخزين أرقام أو حروف كقيم:

```
>>number_123=7;
>>text04='This is GE204';
```

- يوجد فرق بين الحروف الكبيرة والصغيرة:

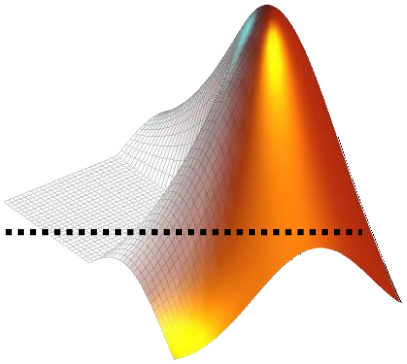
```
>> a = 4;
>> A = 10;
```

A≠a

- يمكن مسح المتغير من الذاكرة:

```
>> clear a
```

Matlab Help



```
>> help

HELP topics:

matlab\general       -  General purpose commands.
matlab\ops           -  Operators and special characters.
matlab\lang          -  Programming language constructs.
matlab\elmat         -  Elementary matrices and matrix manipulation.
matlab\elfun         -  Elementary math functions.
matlab\specfun       -  Specialized math functions.
matlab\matfun        -  Matrix functions - numerical linear algebra.
matlab\datafun       -  Data analysis and Fourier transforms.
matlab\audio         -  Audio support.
matlab\polyfun       -  Interpolation and polynomials.
matlab\funfun        -  Function functions and ODE solvers.
matlab\sparfun       -  Sparse matrices.
matlab\graph2d       -  Two dimensional graphs.
matlab\graph3d       -  Three dimensional graphs.
matlab\specgraph     -  Specialized graphs.
matlab\graphics      -  Handle Graphics.
matlab\uitools       -  Graphical user interface tools.
matlab\strfun        -  Character strings.
matlab\iofun         -  File input/output.
matlab\timefun       -  Time and dates.
matlab\datatypes     -  Data types and structures.
matlab\verctrl       -  Version control.
matlab\winfun        -  Windows Operating System Interface Files (DDE/ActiveX)
matlab\demos         -  Examples and demonstrations.
toolbox\local        -  Preferences.
simulink\simulink    -  Simulink
simulink\blocks      -  Simulink block library.
simulink\simdemos    -  Simulink 4 demonstrations and samples.
simdemos\aerospace   -  Simulink: Aerospace model demonstrations and samples.
simdemos\automotive  -  Simulink: Automotive model demonstrations and samples.
simdemos\simfeatures -  Simulink: Feature demonstrations and samples.
simdemos\simgeneral  -  Simulink: General model demonstrations and samples.
simdemos\simnew      -  Simulink: New features model demonstrations and samples.
simulink\dee         -  Differential Equation Editor
stateflow\stateflow  -  (No table of contents file)
stateflow\sfdemos    -  Stateflow demonstrations and samples.
stateflow\coder      -  (No table of contents file)
control\control      -  Control System Toolbox.
control\ctrlguis     -  Control System Toolbox -- GUI support functions.
control\ctrlobsolete -  Control System Toolbox -- obsolete commands.
control\ctrlutil     -  (No table of contents file)
control\ctrldemos    -  Control System Toolbox -- Demos.
mutools\commands     -  Mu-Analysis and Synthesis Toolbox.
mutools\subs         -  Mu-Analysis and Synthesis Toolbox.
toolbox\robust       -  Robust Control Toolbox.
toolbox\sb2sl        -  SB2SL (converts SystemBuild to Simulink)
signal\signal        -  Signal Processing Toolbox.
signal\fdatoolgui    -  Filter Design & Analysis Tool (GUI)
signal\sptoolgui     -  Signal Processing Toolbox GUI
```

- اكتب

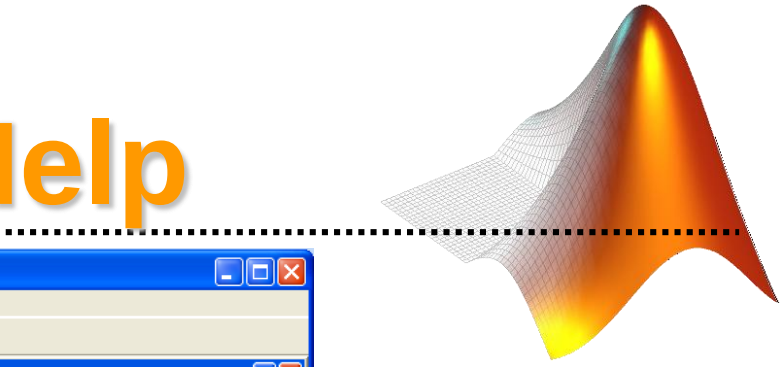
>>help

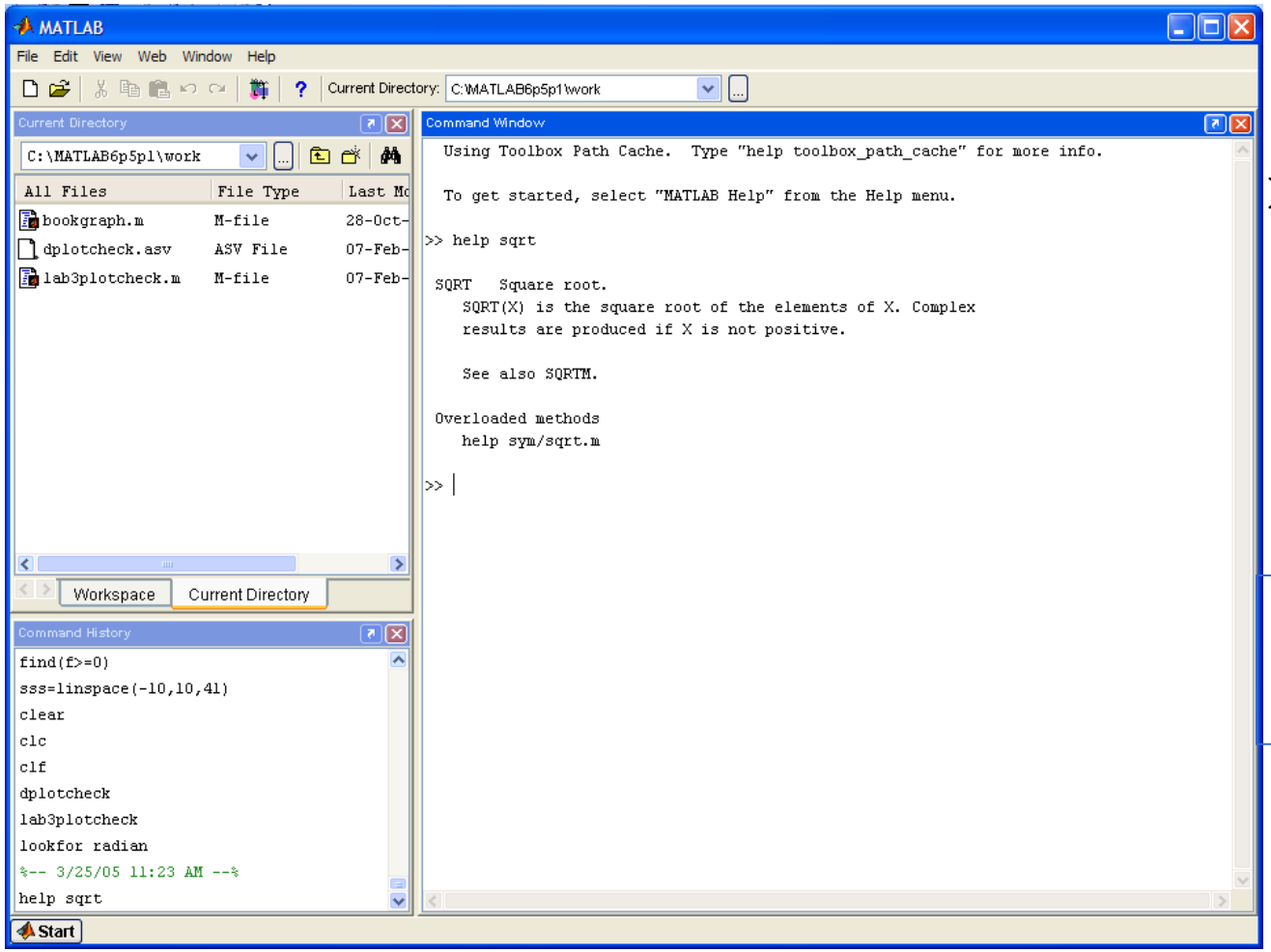
لمعرفة المزيد

عن أوامر ودوال

Matlab

Matlab Help

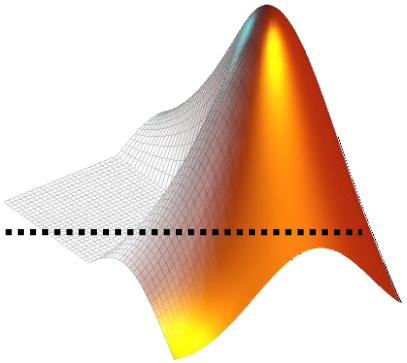





```
>> help sqrt
```

square **r**oot function help

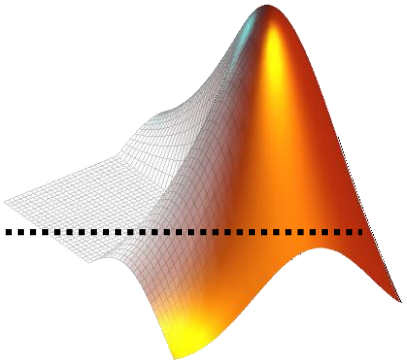
Matlab Help



يمكن أيضا
استخدام أيقونة
help لمعرفة
المزيد عن أوامر
Matlab

نافذة مستقلة

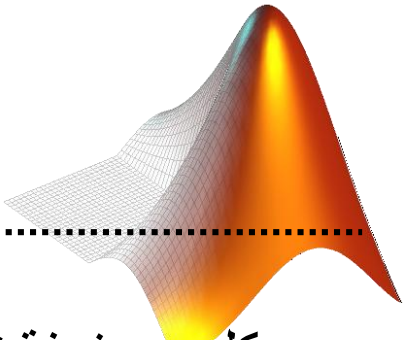
المصفوفات



```
>> A=[6 5; 3 4]
A=
     6     5
     3     4
>> B=[1 2 3 4]
B=
     1  2  3  4
>> C=[4 8
        7 5];
```

- تبرز قوة Matlab عند استخدام المصفوفات
- لادخال مصفوفة، استخدم الأقواس المربعة **[ ]**
- لادخال صف آخر، اضغط **;** أو **Enter**

المصفوفات (تابع)



```
>> A=[6 5; 3 4]
A=
   6          5
   3          4
>> A(1,1)=10
A=
   10         5
   3          4
>> 1:2:7
ans=
1  3  5  7
>> 1:4
ans=
1    2    3     4
```

- كل مصفوفة تتكون من عناصر
- عناصر المصفوفة تكون الصفوف والأعمدة
- يمكن التعامل مع عناصر المصفوفة باستخدام رقم الصف والعمود
- يمكن توليد متجه (مصفوفة ذات بعد واحد) باستخدام الأمر التالي:

initial:step:final

النهاية:الخطوة:البداية

(اذا لم يتم تحديد الخطوة، سيفرض أنها ١)

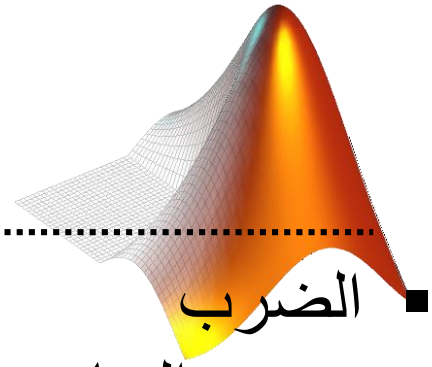
Scalars, Vectors, Matrices

MATLAB treat variables as "matrices"

- **Matrix** (m × n) - a set of numbers arranged in rows (m) and columns (n)
- **Scalar**: 1 × 1 matrix
- **Row Vector**: 1 × n matrix
- **Column Vector**: m × 1 matrix

$$A = \begin{bmatrix} 5.27 \end{bmatrix} \qquad B = \begin{bmatrix} 5.02 & -2.3 & 7.21 \end{bmatrix}$$

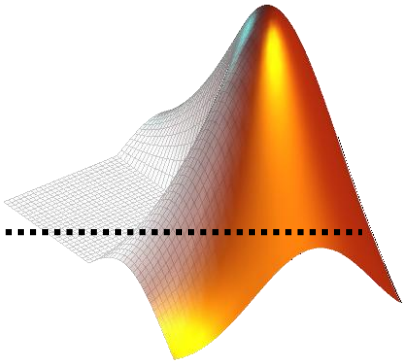
$$D = \begin{bmatrix} 1 & 3 & -2 & 5 \\ -2 & 4 & 3.2 & 9.5 \\ -0.5 & -1 & 7.2 & -2 \end{bmatrix}$$



عمليات المصفوفات

الضرب	*
ضرب العناصر	.*
القسمة	/
قسمة العناصر	./
الأس	^
الأس للعناصر	.^

أمثلة لعمليات المصفوفات



$$\mathbf{A}=\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix}\quad \mathbf{B}=\begin{bmatrix}2 & 3 & 4\\ 5 & 6 & 7\end{bmatrix}\quad \mathbf{C}=\begin{bmatrix}4 & 5\\ 6 & 7\\ 8 & 9\end{bmatrix}$$

- لجمع وطرح المصفوفات ، يجب أن تكون المصفوفات من نفس الحجم
- لضرب المصفوفات لا بد من تساوي الأبعاد الداخلية

صحيحة

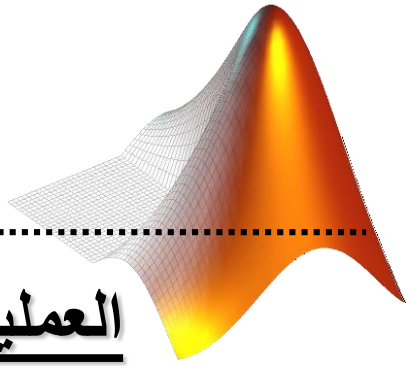
```
>>D=A+B;
>>D=A-B;
>>D=A*C;
>>D=C*A;
>>D=A.*B;
```

ضرب المصفوفات غير تبديلي

خطأ

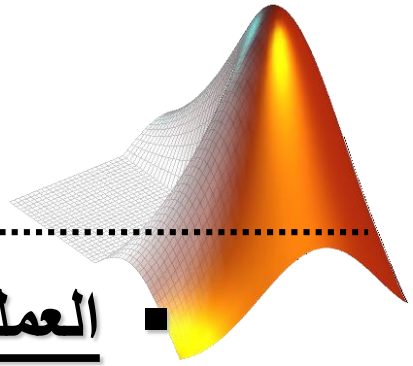
```
>>D=A+C;
>>D=A*B;
>>D=B*A;
>>D=A.*C;
```

عمليات المصفوفات (تابع)



العملية الرياضية	أمر Matlab
$\mathbf{C} = \mathbf{A}^{-1}\mathbf{B}$	`>> C=A\B;`
$\mathbf{C} = \mathbf{B}\mathbf{A}^{-1}$	`>> C=B/A;`

منقول مصفوفة



- **العملية الرياضية** **أمر Matlab**

$C=A^T$

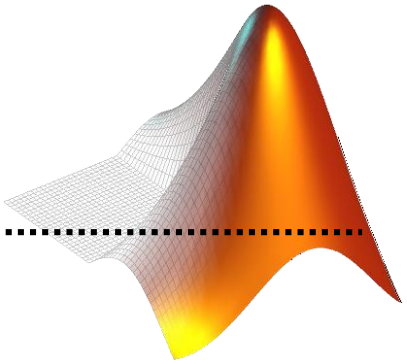
```
>>C=A';
```

$$\mathbf{A}=\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix}$$

```
>>B=A';
```

$$\mathbf{B}=\begin{bmatrix}1 & 4\\ 2 & 5\\ 3 & 6\end{bmatrix}$$

تضمين المصفوفات

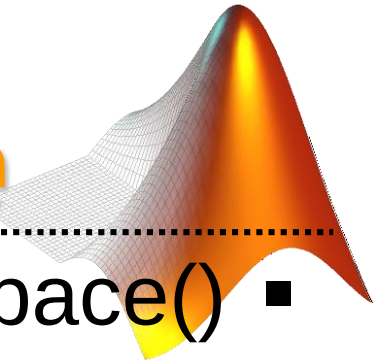


$$\mathbf{A}=\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix} \qquad \mathbf{B}=\begin{bmatrix}2 & 3 & 4\\ 5 & 6 & 7\end{bmatrix}$$

```
>> D=[A B];
```

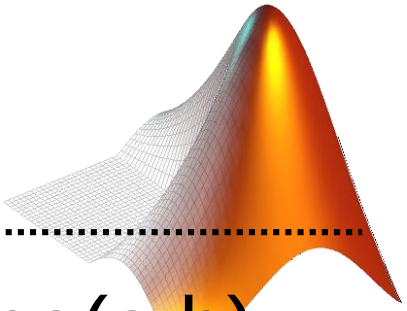
$$D=\begin{bmatrix}\overbrace{1 & 2 & 3}^{A} & \overbrace{2 & 3 & 4}^{B}\\ 4 & 5 & 6 & 5 & 6 & 7\end{bmatrix}$$

بعض دوال الماتلاب الأساسية



- Linspace()
- Size()
- Length()
- Magic()
- Zeros()
- Ones()
- Inv()&det()
- Eig()

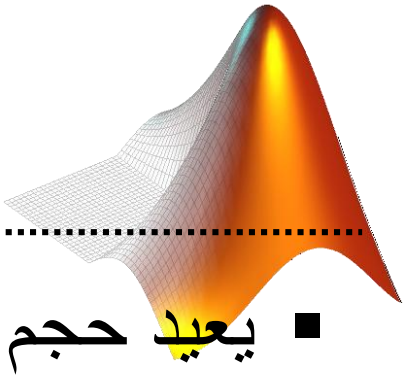
Linspace()



- Linspace(a,b) يولد نسق سطري من 100 قيمة بدءا من a وانتهاء بـ b.
- Linspace(a,b,n) يولد نسق سطري من n قيمة بدءا من a وانتهاء بـ b.
- القيم المولدة تشكل متوالية حسابية اساسها r=(b-a)/(n-1)
- مثال:

```
>>x=linspace(1,10,5)
    x=
       1  3.25  5.5  7.57  10
```

Size()



- يعيد حجم المصفوفة أو النسق الممر اليه وله الأشكال التالية:
- d=size(a) يعيد في d عدد اعمدة وعدد اسطر a
- [m n]=size(a) يعيد في m عدد اسطر a وفي n عدد أعمدة a
- مثال:

- >>a=[1 2 3];
- >>d=size(a)

d=

1 3

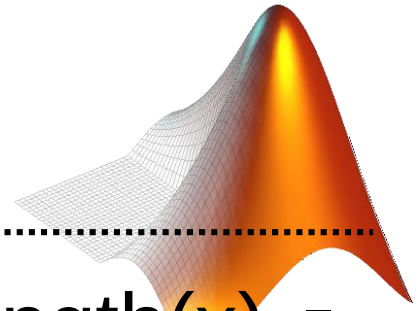
>>a=[2 3;4 5];

>> [m n]=size(a)

m=2

n=2

Length()



- Length(x) يعيد طول الشعاع x ، وهو يعيد القيمة العظمى من القيمتين التي يعيدها التابع size من أجل المصفوفات غير الفارغة و صفر من أجل المصفوفات الفارغة.
- مثال :

- >>a=[1;2;3;4];
- >>b=length(a)

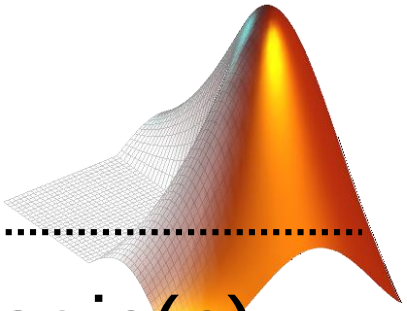
  b=4

>>a=[];

>>b=length(a)

b=0

Magic()



- Magic(n) يولد مصفوفة n*n تتألف من أعداد صحيحة تنتمي قيمها إلى المجال [1..n^2] وتكون اسطرها و أعمدتها و اقطارها متساوية المجموع.
- ستكون المصفوفة السحرية الناتجة صحيحة في حال n>2.
- مثال :

```
>> m=magic(3)

m =

     8     1     6
     3     5     7
     4     9     2
```

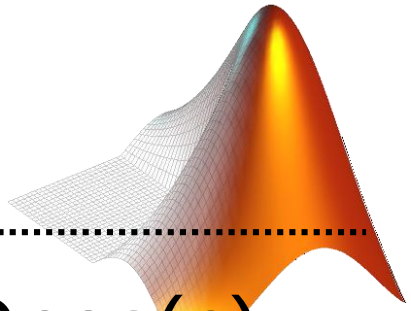
Zeros()

- Zeros(n) : يولد مصفوفة n*n من الأصفار
- Zeros(m,n) أو zeros([m n]) يولد مصفوفة m*n من الأصفار.

```
>> zeros(3)

ans =

     0     0     0
     0     0     0
     0     0     0
```

```
>> zeros(2,5)

ans =

     0     0     0     0     0
     0     0     0     0     0
```

Ones()

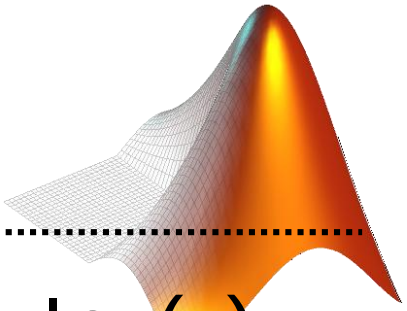


- Ones(n) يولد مصفوفة n*n من الواحدات.
- Ones(m,n) أو ones([m n])يولد مصفوفة m*n من الواحدات
- أمثلة:

```
>> ones(4,2)

ans =

     1     1
     1     1
     1     1
     1     1
```

```
>> ones(3)

ans =

     1     1     1
     1     1     1
     1     1     1
```

Inv() & det()

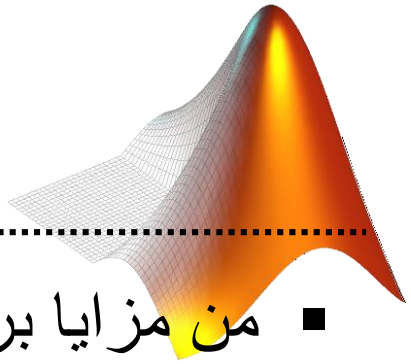


- Inv(x) يعيد مقلوب المصفوفة المربعة x.
- Det(x) يعيد محدد المصفوفة المربعة x.
- أمثلة :

```
>> a=[1 3;4 5];
>> inv(a)

ans =

   -0.7143    0.4286
    0.5714   -0.1429
```

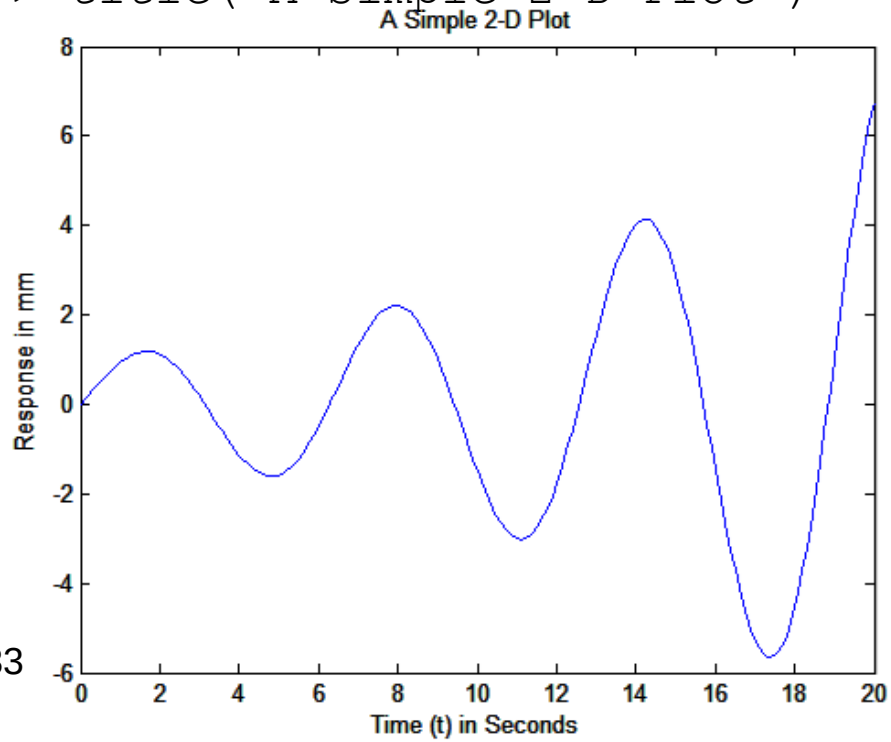
```
>> a=[1 3;4 5];
>> det(a)

ans =

    -7
```

رسومات Matlab

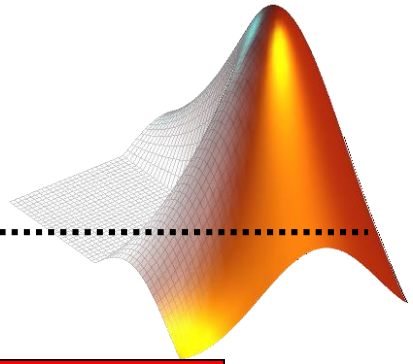


- من مزايا برنامج Matlab الرسومات
- توجد العديد من الخيارات، حيث يمكن التحكم بـ:
  - نوع الرسم
  - نوع النقاط والخطوط
  - العناوين
  - التحكم بالمحاور
  - عدد الرسومات

```
>> x=0:.1:20;
>> y=exp(0.1*x).*sin(x);
>> plot(x,y)
>> xlabel('Time (t) in Seconds')
>> ylabel('Response in mm')
>> title('A Simple 2-D Plot')
```



أمر plot

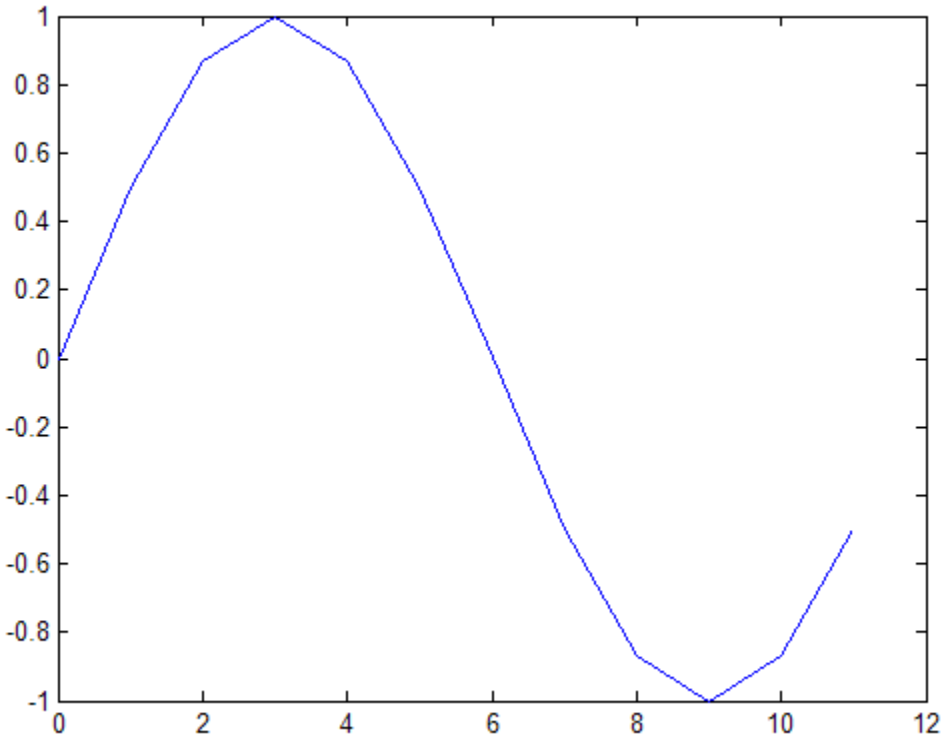


```
>> n=0:11;
>> y=sin((pi/6)*n);
```

لا بد من تساوي أبعاد x1, y1 , ...!

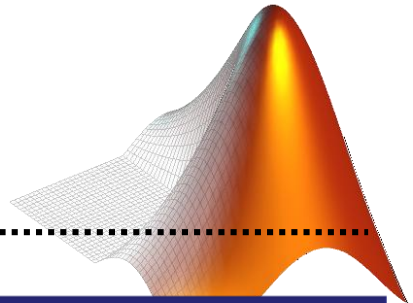
```
>> plot(n,y)
```

رسم بدون أية خيارات



يمكن معرفة المزيد
عن الأمر بكتابة:

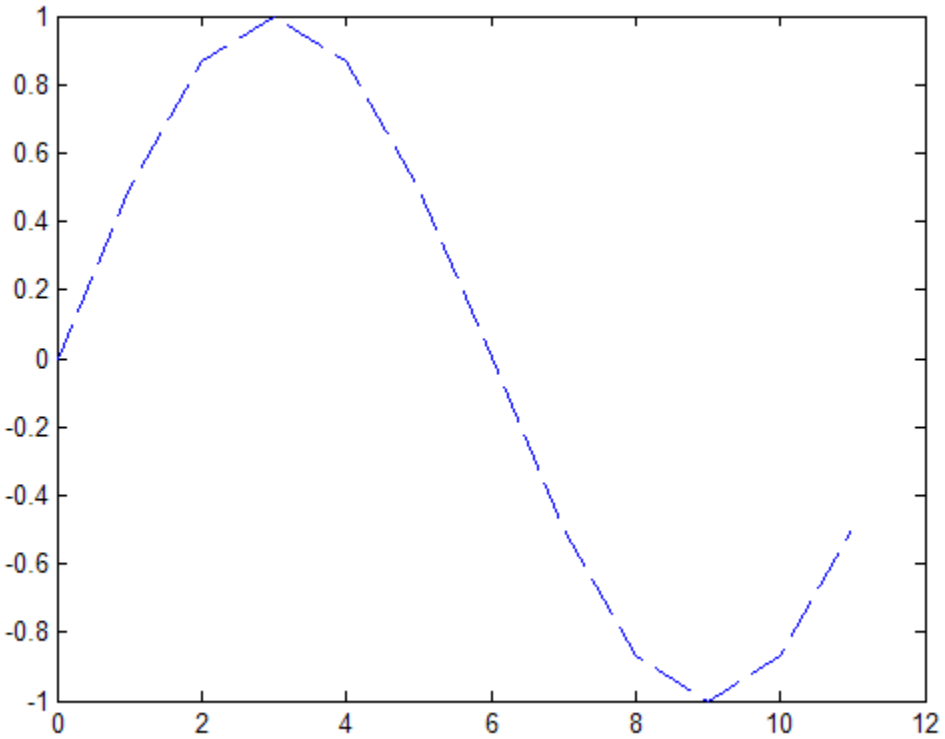
```
>> help plot
```



```
plot(x1,y1[,x2,y2,x3,y3.....])
```

```
>> plot(n,y,'--')
```

رسم باستخدام
خط متقطع

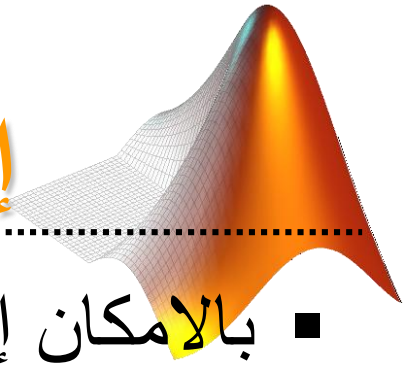


Color, Symbols, and Line Types

Use "help plot" to find available Specifiers

Colors		Symbols		Line Types	
b	blue	.	point	-	solid
g	green	o	circle	:	dotted
r	red	x	x-mark	-.	dashdot
c	cyan	+	plus	--	dashed
m	magenta	*	star		
y	yellow	s	square		
k	black	d	diamond		
		v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		

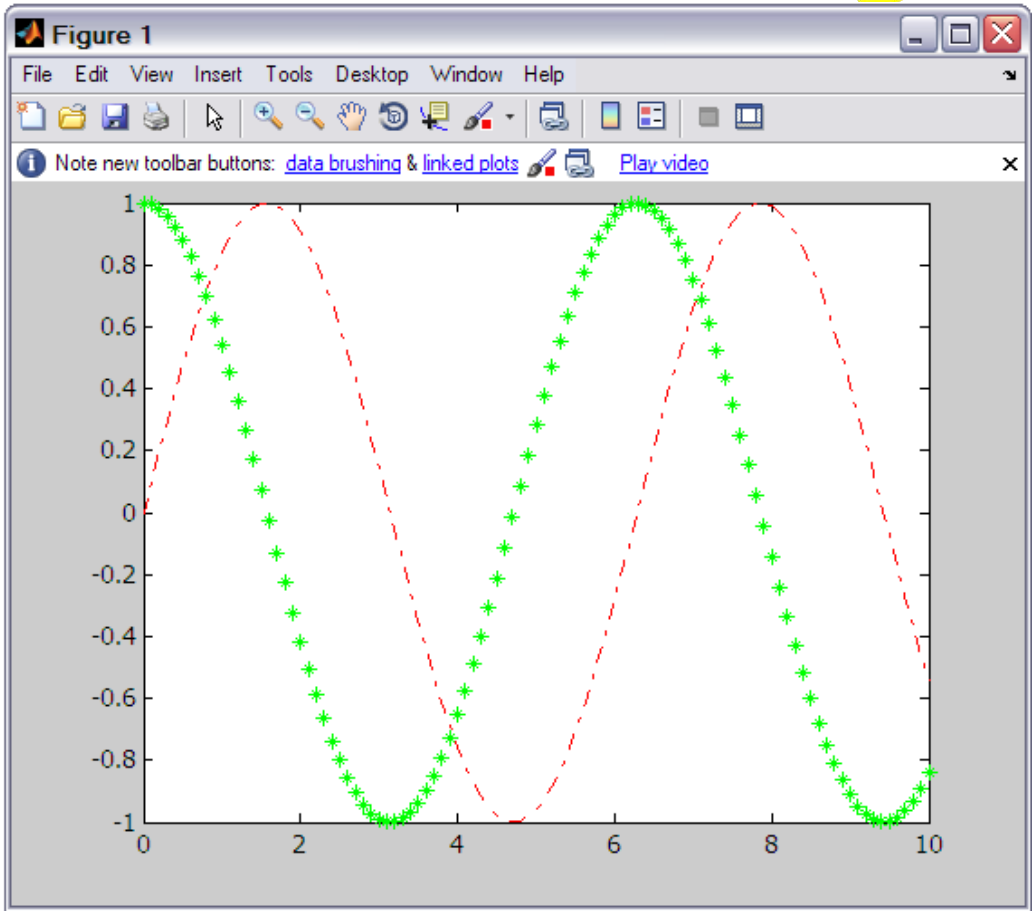
إضافة شكل إلى شكل موجود



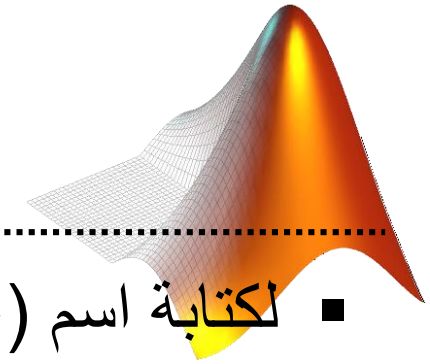
- بالامكان إضافة رسم إلى رسم آخر موجود من خلال استخدام الأمر hold فعند استخدام الأمر hold on فإن ماتلاب لا يزيل الرسم الموجود و إنما يضيف الرسم الجديد إلى الرسم الحالي .
- يستخدم hold off لتحرير الرسم السابق .

مثال

```
>> x=0:0.1:10;
>> y=sin(x);
>> plot(x,y,'r-.')
>> hold on
>> z=cos(x);
>> plot(x,z,'g*')
>> hold off
```



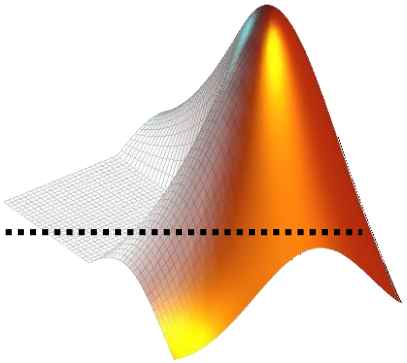
دوال الرسومات



- لكتابة اسم (عنوان) الرسمة
- لكتابة اسم محور **x**
- لكتابة اسم محور **y**
- لعرض شبكة بالرسم
- لتحديد القيم الصغرى والكبرى للمحاور
- لتوضيح أسماء الخطوط المرسومة
- للانتقال الى رسمة أخرى

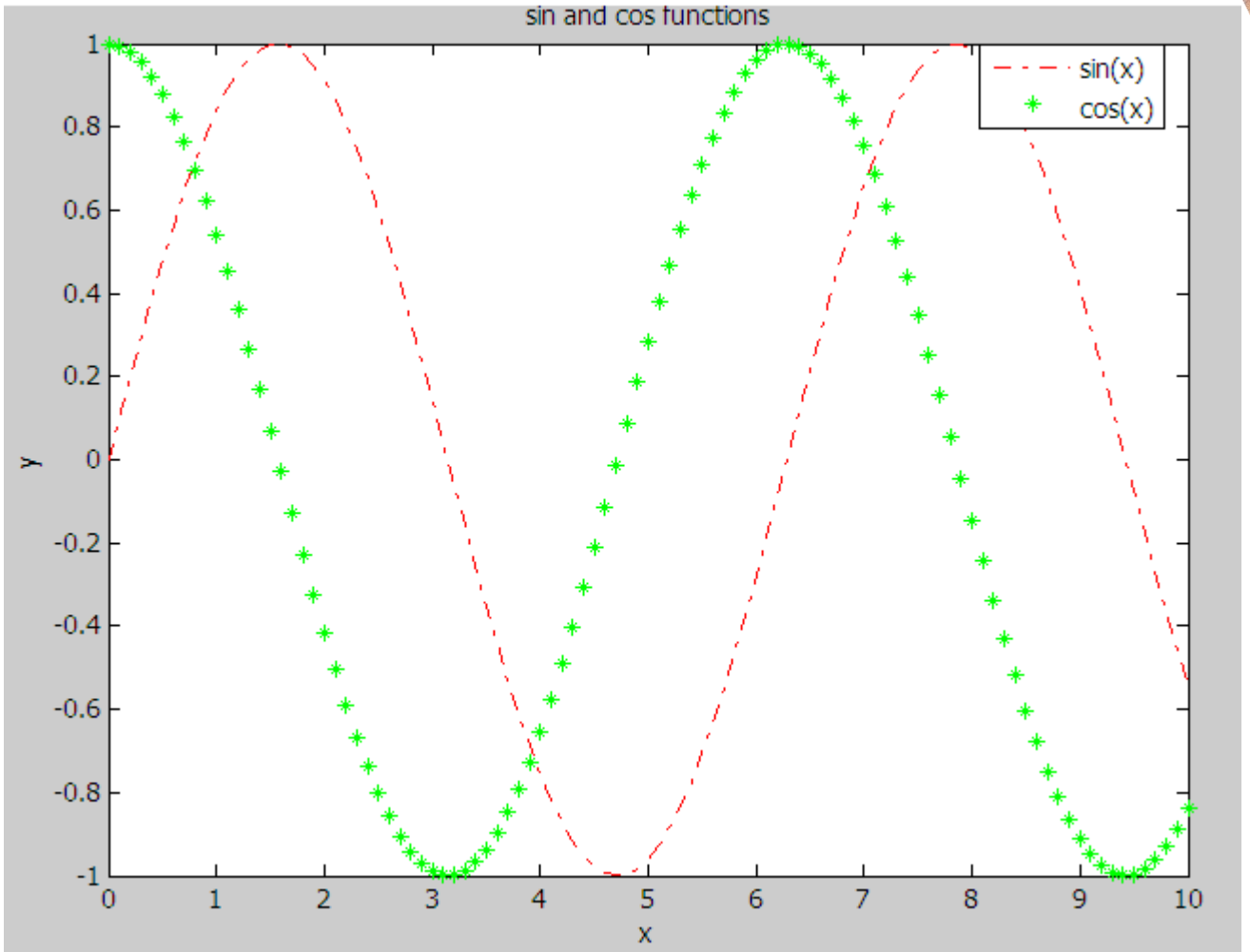
- `title`
- `xlabel`
- `ylabel`
- `grid`
- `axis`
- `legend`
- `figure`

مثال

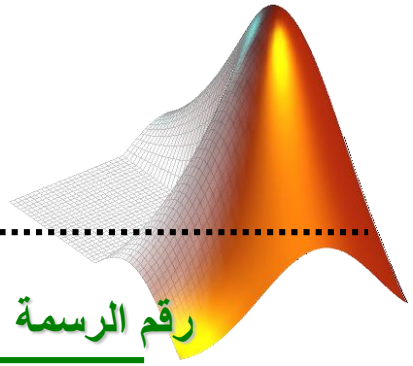


```
>> xlabel('x')
>> ylabel('y')
>> title('sin and cos functions')
>> x=0:0.1:10;
>> y=sin(x);
>> plot(x,y,'r-.')
>> z=cos(x);
>> hold on
>> plot(x,z,'g*')
>> xlabel('x')
>> ylabel('y')
>> title('sin and cos functions')
>> hold off
>> legend('sin(x)','cos(x)')
```

مثال


sin and cos functions
sin(x)
cos(x)
y
x
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
1
2
3
4
5
6
7
8
9
10

أمر subplot

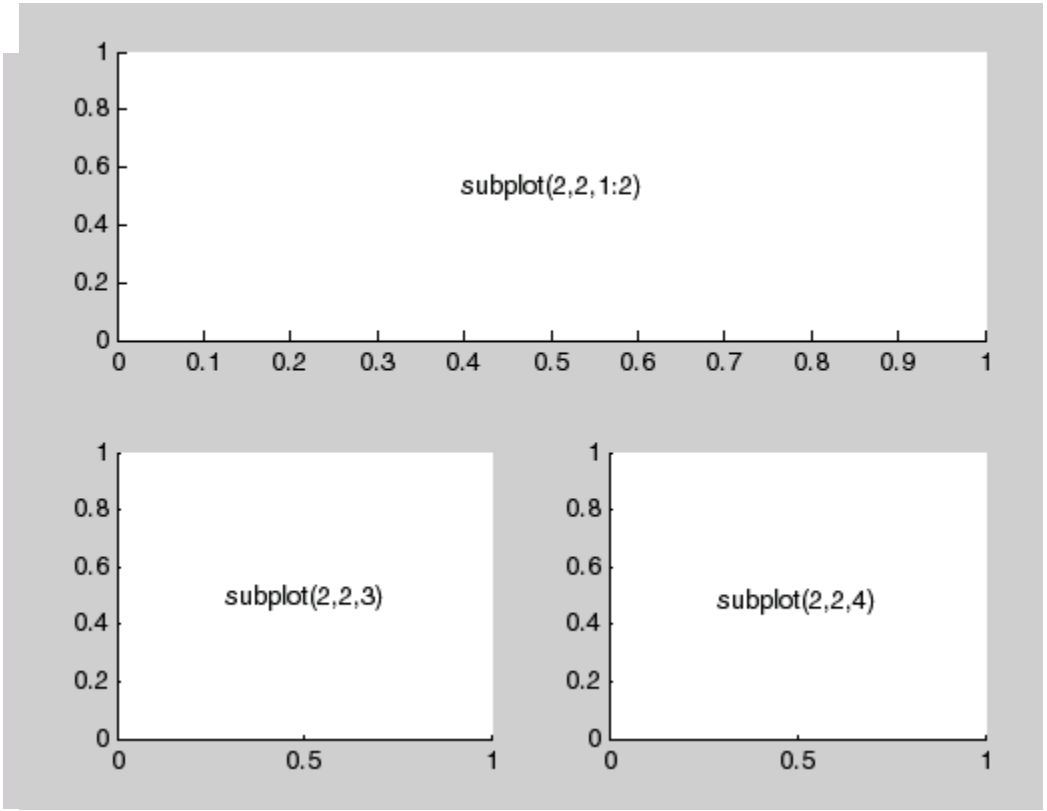


```
subplot(m,n,p)
```

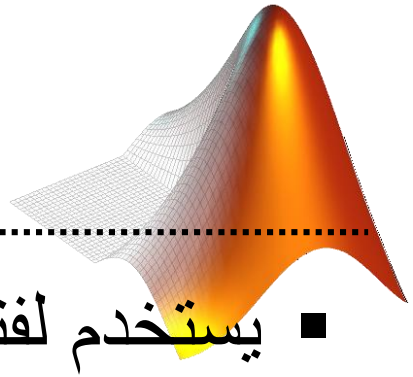
رقم الرسمة

عدد الأعمدة

عدد الصفوف



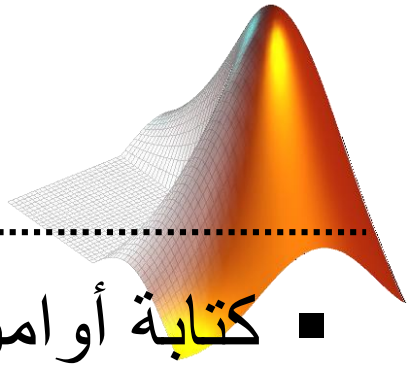
الأمر figure



- يستخدم لفتح نافذه لصورة جديدة
- مثال:

```
>> x=0:100;
>> y=2*x;
>> z=x.^3;
>> w=x.^3+2*x;
>> figure
>> plot(x,y)
>> figure
>> plot(x,z)
>> figure
>> plot(x,w)
```

ملفات Matlab



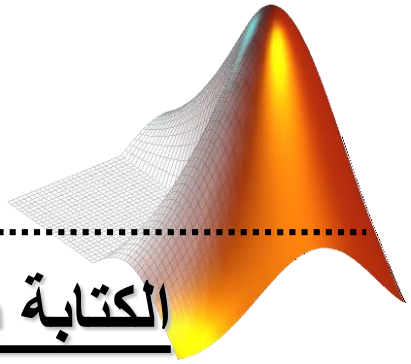
- كتابة أوامر Matlab كثيرة عدة مرات غير عملية
- يتيح Matlab إمكانية تجميع الأوامر بشكل ملف
- يكفي كتابة إسم الملف لتنفيذ جميع الأوامر المضمنة
- يوجد نوعان من الملفات:
  - Script files (ملفات تحتوي على مجموعة أوامر)
  - Function files (ملفات بشكل دوال)
- جميع الملفات تحتوي على الامتداد **.m**

انشاء الملفات M (M-files)

- لكتابة أي ملف M-file:
  - بكتابة `>>edit filename` في نافذة الأوامر
  - باختيار New > M-file
  - بالنقر على

- لتنفيذ الملفات، فقط اكتب اسم الملف، وستنفذ جميع الأوامر المتضمنة بالملف

Script Files



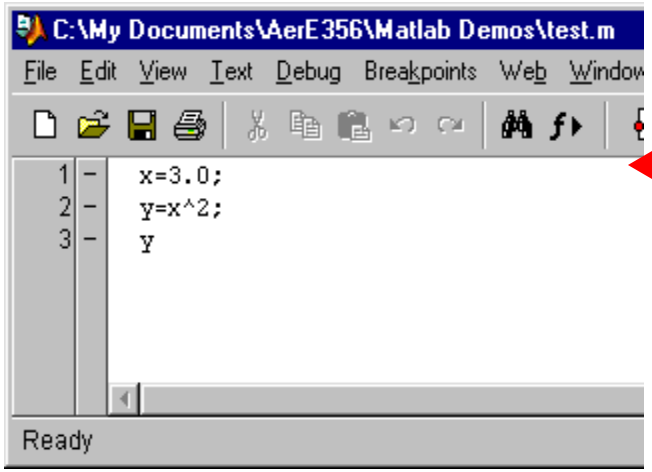
الكتابة بنافذة الأوامر

```
>> x=3.0;
>> y=x^2;
>> y
y =
   9.0
>>
```

يمكنها التعامل مع المتغيرات بذاكرة Matlab!

ملف script:

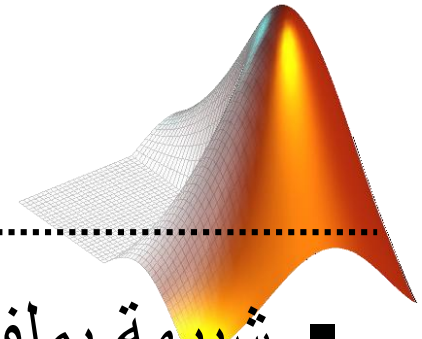
- تخزين الأوامر بملف (على سبيل المثال باسم test)



- ثم كتابة الاسم بنافذة الأوامر:

```
>> test
y =
     9.0
```

ملفات الدوال



تعريف الدالة

- شبيهة بملفات script، مع الفروقات التالية:

```
function [list of outputs] = ge204(list of inputs)
% function help
.
.
.
program
.
.
.
return
```

قائمة المخرجات

اسم الدالة

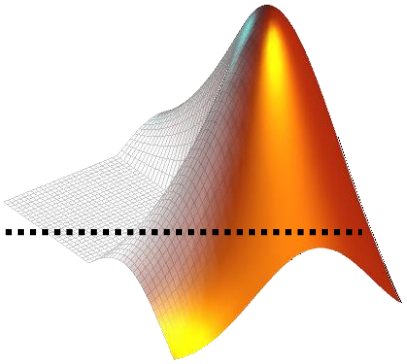
قائمة المدخلات

توضيح الدالة للمستخدم

للرجوع

لا تتعامل مع المتغيرات بذاكرة Matlab (متغيرات مستقلة)!

مثال



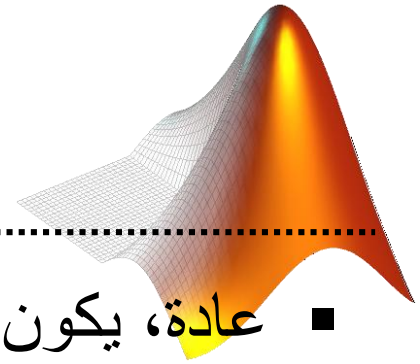
الدالة cmb

```
function [x,y]=cmb(a,b)
%Test function
x=a+b;
y=a/b;
return
```

نافذة الأوامر Command Window

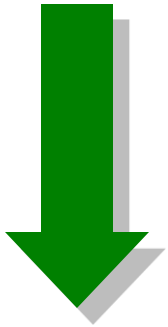
```
>> [j,k]=cmb(5,2)
j=7
k=2.5
>>help cmb
Test function
```

التحكم بالتنفيذ



- عادة، يكون تنفيذ الأوامر بالتسلسل

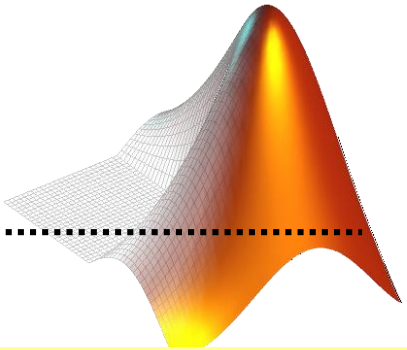
```
a=2;
b=3;
c=a+b;
d=sqrt(b);
```



- باستخدام أوامر التحكم بالتنفيذ، يمكن تغيير هذا التسلسل
- <u>الأوامر المهمة:</u>

– إذا	`if`	أوامر شرطية (تنفذ مرة واحدة)
– إذا-غير ذلك	`if-else`	
– طالما	`while`	أوامر تكرار
– لـ...	`for`	

أوامر if و while



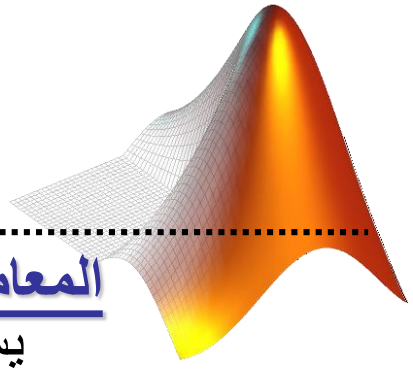
```
while expression
        statements
end
```

```
if expression
        statements
end
```

```
if expression
        statements1
else
        statements2
end
```

- يقوم Matlab بتنفيذ الأوامر (*statements*) فقط إذا تحقق الشرط (*expression*)
- عند التحقق من الشرط (*expression*)، يخرج Matlab إما بنتيجة صحيح (true) أو غير صحيح (false)
- الأوامر (*statements*) هي أي أوامر بالـ Matlab

تقييم الشرط



المعاملات الشرطية

==	يساوي
<	أقل من
>	أكبر من
<=	أقل من أو يساوي
>=	أكبر من أو يساوي
~=	لا يساوي

أية معادلة حسابية

```
a=4;
b=5;
c=5;
if a+b  "true"  ← لأن الناتج ≠ ٠
if b-c  "false" ← لأن الناتج = ٠
```

المعاملات المنطقية

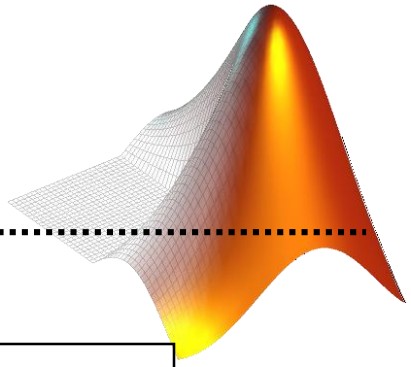
&	and	و
|	or	أو

```
while(3<=a)&(a<=5)
```

انتبه من أخطاء التقريب!

```
if sin(0) "false"
if sin(pi)"true"
sin(pi) = 1.22e-16
```

أمثلة (if/if-else)



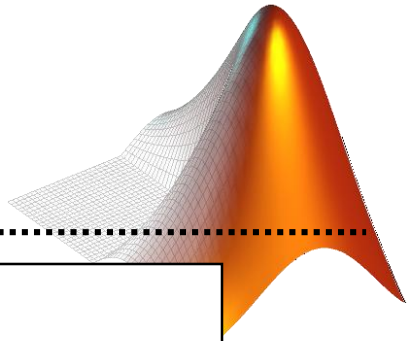
```
>> a=6;
>> if a<10
b=a/2;
end
>> b
b = 3
```

```
>> a=50;
>> if a<10
b=a/2;
end
>> b
??? Undefined function or variable 'b'.
```

```
>> a=6;
>> if a<10
b=a/2;
else
b=a/5;
end
>> b
b = 3
```

```
>> a=50;
>> if a<10
b=a/2;
else
b=a/5;
end
>> b
b = 10
```

أمثلة (while)



```
>> a=1;
>> while a<5
a=a+1
end
a = 2
a = 3
a = 4
a = 5
```

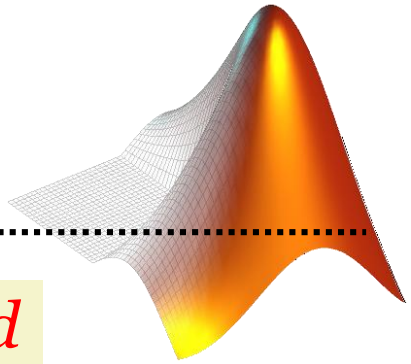
```
>> a=10;
>> while a<5
a=a+1
end
>> a
a = 10
```

أمر تكرار!

```
>> a=1;
>> while a<5
b=a+1
end
b = 2
b = 3
b = 4
b = 5
b = 6
b = 7
.
.
.
.
```

∞ loop!

أمر for

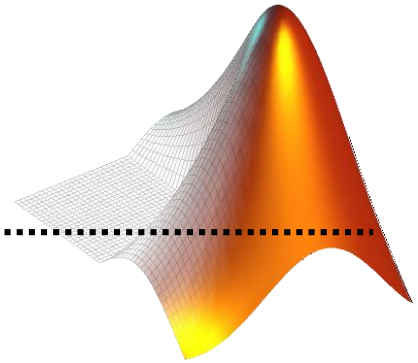


أمر تكرار!

```
for index = start : [increment :] end
        statements
end
```

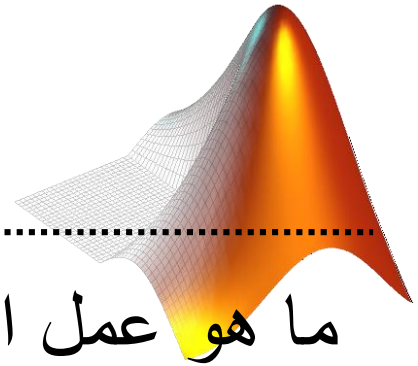
- قيمة الزيادة (*increment*) اختيارية، وستفرض قيمتها ١ إذا لم توضع
- يمكن لقيم البداية والزيادة والنهاية أن تكون كسور
- يمكن للزيادة أن تكون موجبة أو سالبة
- ينتهي التكرار إذا كان المتغير > النهاية (أو المتغير < النهاية)

أمثلة



```
>> for a=1:5
a
end
a = 1
a = 2
a = 3
a = 4
a = 5
```

وظيفة



ما هو عمل التوابع التالية ؟

- Min
- Max
- Round
- Ceil
- Sort