

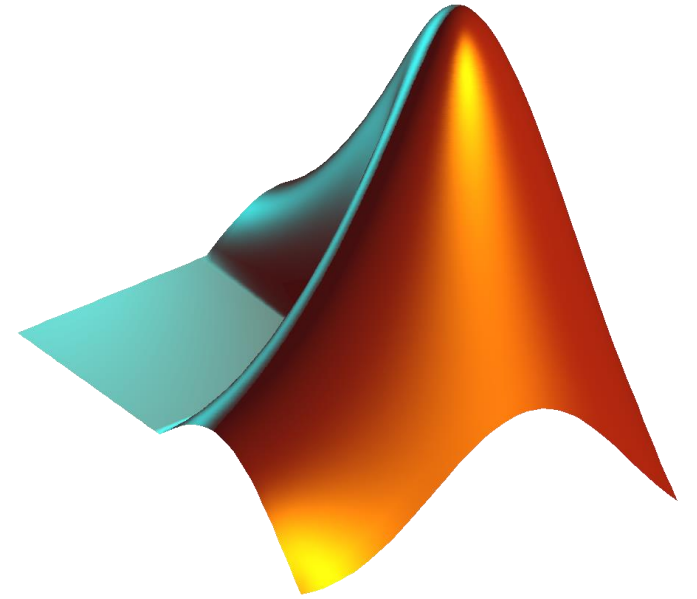
Introduction to MATLAB

Panchatcharam Mariappan

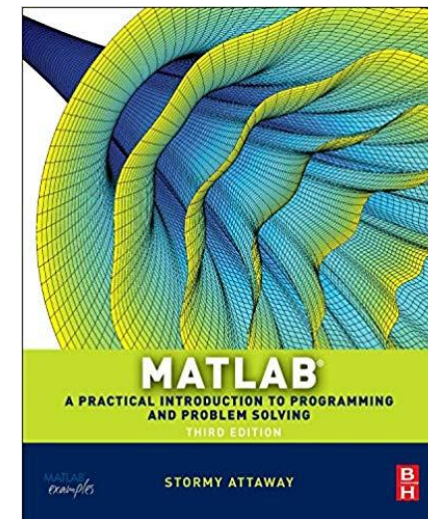
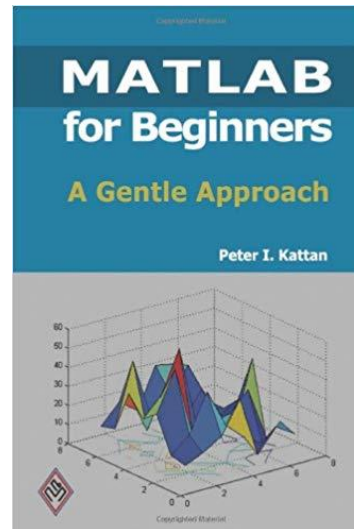
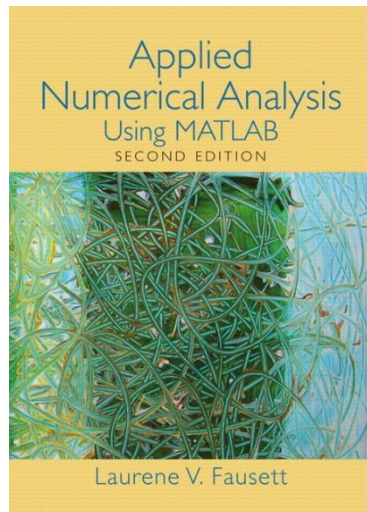
Associate Professor

**Department of Mathematics,
IIT Tirupati**

MATLAB



- Laurene v. Fausett, Applied Numerical Analysis Using MATLAB, 2nd Edition, 2008
- Matlab Primer, 2014b, The Mathworks Inc.
- Stormy Attarway, Matlab: A Practical Introduction to Programming and Problem Solving 3rd Edition, 2013
- Peter I. Kattan, MATLAB For Beginners: A Gentle Approach, 2018
- S. J. Chapman, Matlab Programming for Engineers, 2004



- <https://in.mathworks.com/help/index.html>

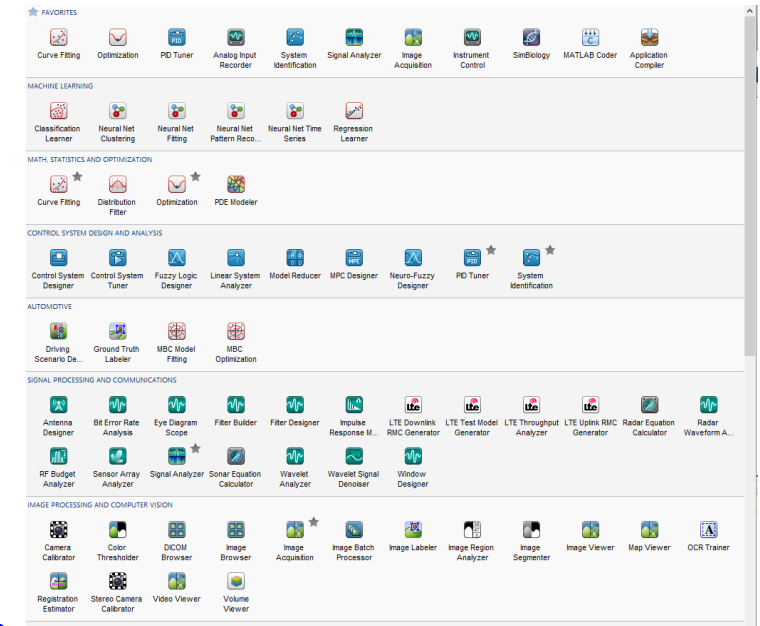
- MATLAB stands for **MATrix LABoratory**
- Originally written to provide easy access to matrix software
 - Developed by LINPACK and EISPACK projects
- High-performance language for technical Computing

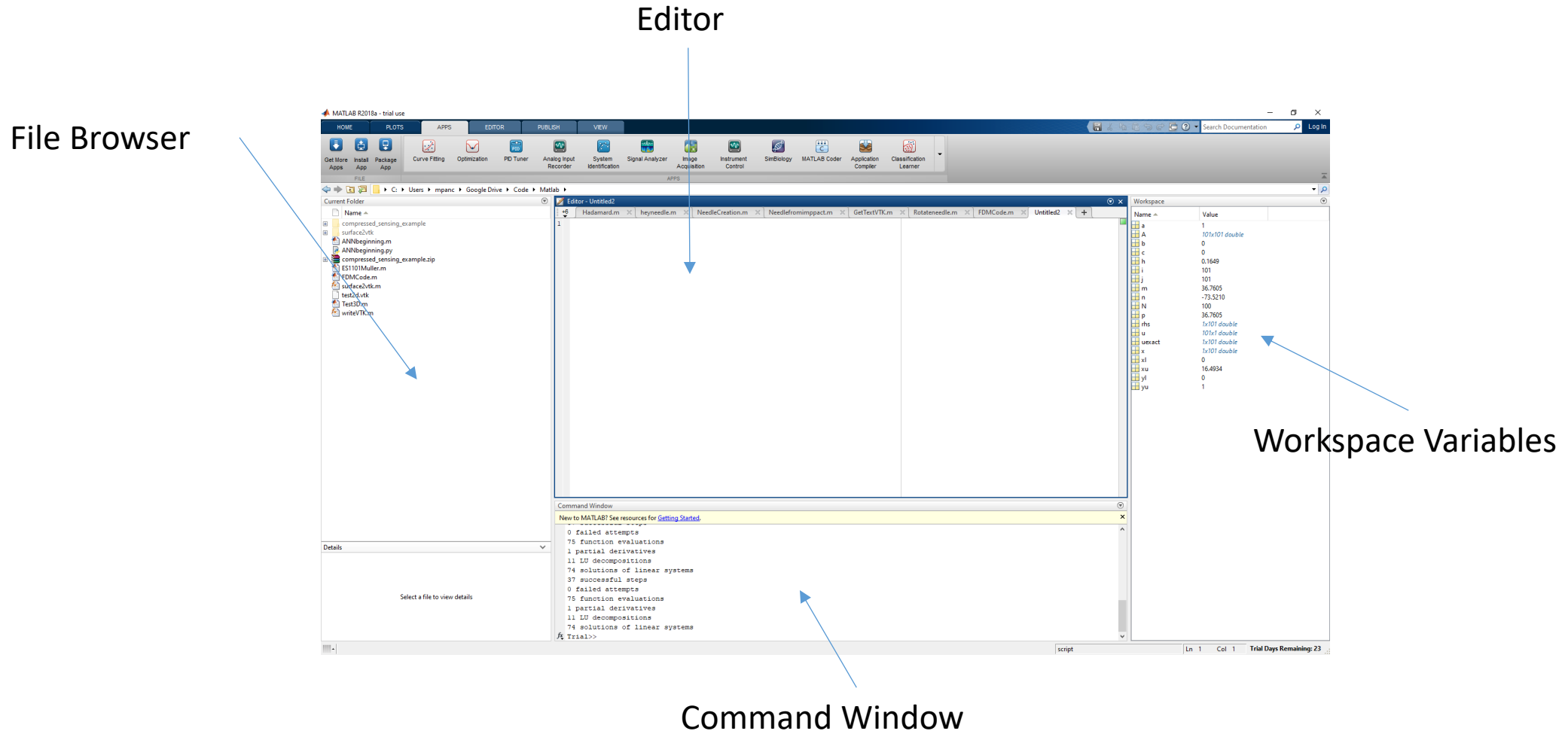
- Computation
- Visualization
- Programming
- Data Structures
- Debugging tool
- Object-Oriented Programming

- Interactive System
- Basic data is an array without any dimension
- Powerful built-in routines
- Graphics commands for visualization

- Signal Processing
- Symbolic Computation
- Control theory
- Simulation
- Optimization
- Curve Fitting

- PID Tuner
- Analog Input Recorder
- Image Acquisition
- Simbiology
- Machine Learning
- Distribution





1. Open a new file.
2. Save the file name. FirstProgram.m (.m denotes matlab file)
3. Type the following code

```
x = 1 + 2*3
```

```
Y=[2,3,4,5]
```

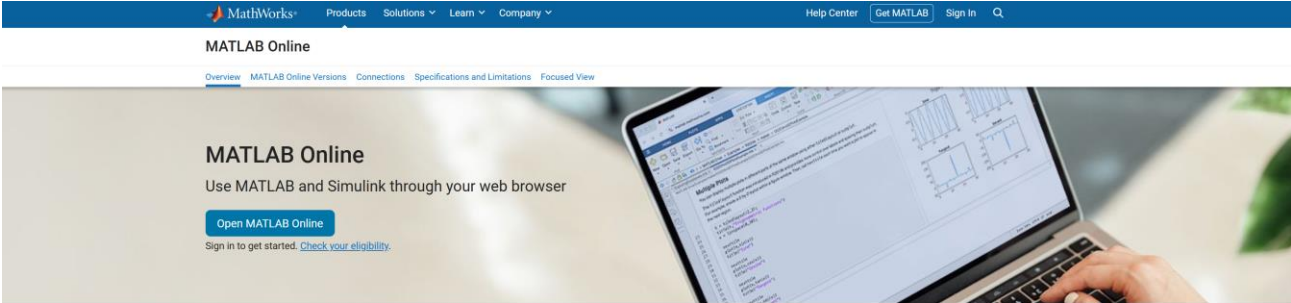
```
A=[1,2,3;4,5,6;7,8,9]
```

4. Save the file
5. Press F5 or go to Editor menu and then press RUN button
6. View the output

Visit the Following Link

<https://in.mathworks.com/products/matlab-online.html>

Click **Open MATLAB Online**



The screenshot shows the MATLAB Online website. The header includes the MathWorks logo and navigation links for Products, Solutions, Learn, and Company. A search bar is on the right. The main content area features the title "MATLAB Online" and the subtitle "Use MATLAB and Simulink through your web browser". A prominent blue button labeled "Open MATLAB Online" is present, along with a link to "Check your eligibility". Below this, a paragraph explains that MATLAB Online extends capabilities to the cloud, allowing connection to cloud storage and collaboration without software installation. At the bottom, three sections are highlighted: "Use MATLAB Through Your Browser" (describing access to the latest versions), "Connect Your Files and Projects" (showing integration with cloud storage like OneDrive and Amazon S3), and "Connect to Hardware" (showing integration with mobile devices via ThingSpeak and various IoT hardware like Arduino and Raspberry Pi).

MATLAB Online
Use MATLAB and Simulink through your web browser
[Open MATLAB Online](#)
Sign in to get started. [Check your eligibility.](#)

MATLAB Online extends the capabilities of MATLAB and Simulink to the cloud. You can connect to cloud storage solutions and collaborate on projects through a web browser without installing software. With support for hardware integration and scalable cloud computing, MATLAB Online enhances your workflow for flexible, on-the-go productivity.

Use MATLAB Through Your Browser
Use the latest versions of MATLAB, Simulink, and add-on products* directly through your web browser —no downloads, installations, or setup necessary. Get several vCPUs with 16GB of memory (provided compute) that can burst depending on load. Choose between 5GB (free) or 20GB (licensed) storage through MATLAB Drive.

Connect Your Files and Projects
Connect to your workflow essentials with MATLAB Online using MATLAB Drive and Microsoft OneDrive* for cloud-based file storage. Sync files with your local desktop via MATLAB Connector. Leverage Amazon S3* for cloud data management. Explore and share projects on File Exchange and GitHub*.

Connect to Hardware
Use your Android® or Apple iOS mobile phone sensors via MATLAB Mobile or access IoT devices with ThingSpeak. Connect and manipulate data from Arduino® and Raspberry Pi®. Use webcams for real-time image and video analysis.
[Using USB Webcams with MATLAB Online \(2:22\)](#)

Create Account



Sign In

By signing in you agree to our [privacy policy](#).

[Create Account](#) [Next](#)



Create Account

i To access your organization's MATLAB license, use your work or university email.

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

[Next](#)

Open MATLAB Online

[Open MATLAB Online](#)
[Install MATLAB](#)

Recent

[Files](#)
[Folders](#)

Name	Last Accessed
Chapter4_TaylorSeriesDemo.mlx MATLAB Drive/MathExamples/mathcourseware/Calculus	7 months ago
ML02_Heatflow2D_simple.mlx ...rive/MathExamples/mathcourseware/SolvingODEsPDEs/Applications of PDEs/thermal-tile-modelling-master	7 months ago
Double_TripleIntegral.mlx MATLAB Drive/MathExamples/mathcourseware/Calculus	7 months ago
Chapter3_Continuity.mlx MATLAB Drive/MathExamples/mathcourseware/Calculus	7 months ago
Chapter2_Limits.mlx MATLAB Drive/MathExamples/mathcourseware/Calculus	7 months ago

[View more](#)
[View all files](#)

Online Training

MATLAB Onramp
Unlimited Access

[Start](#)

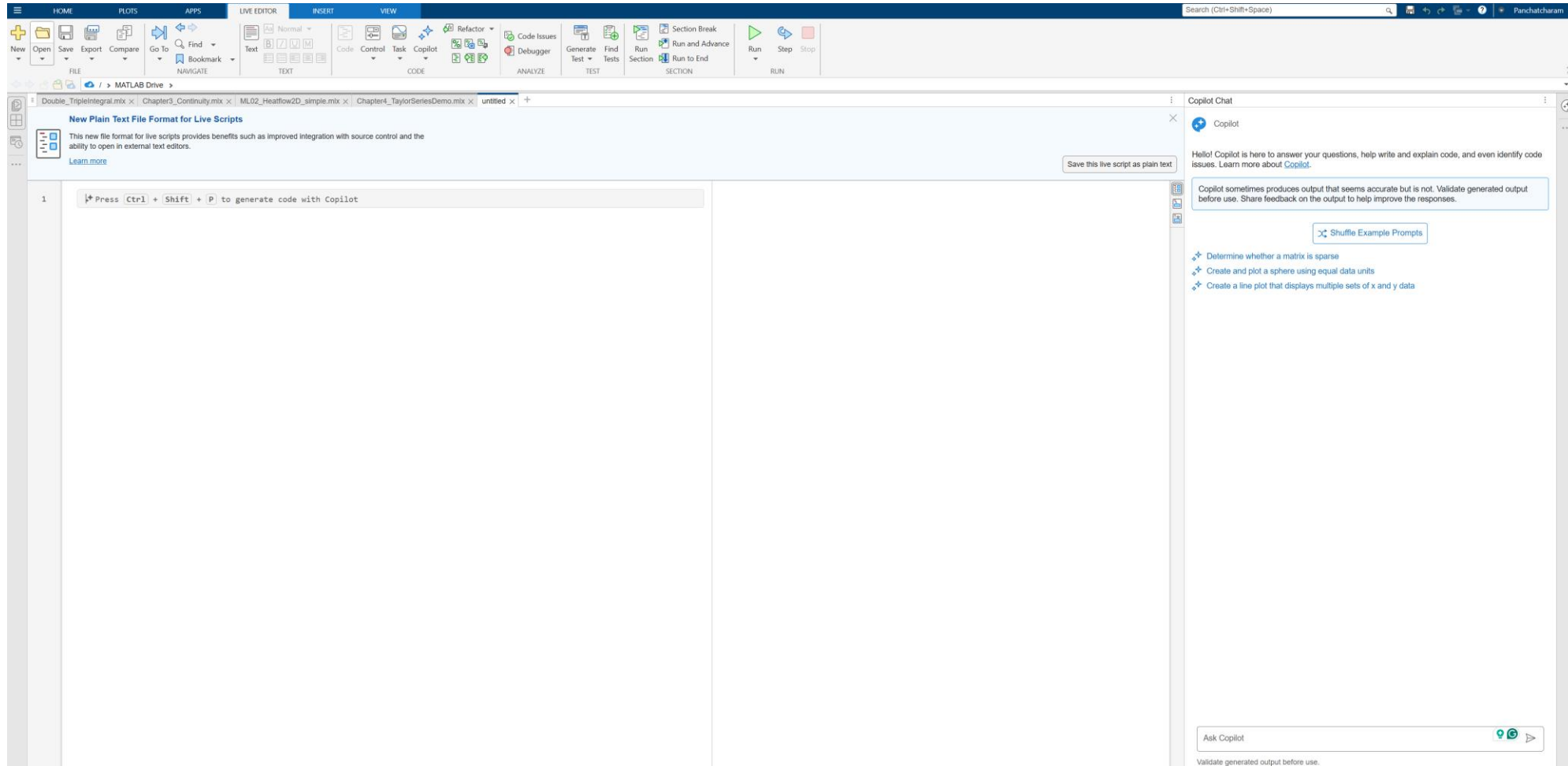
Simulink Onramp
Unlimited Access

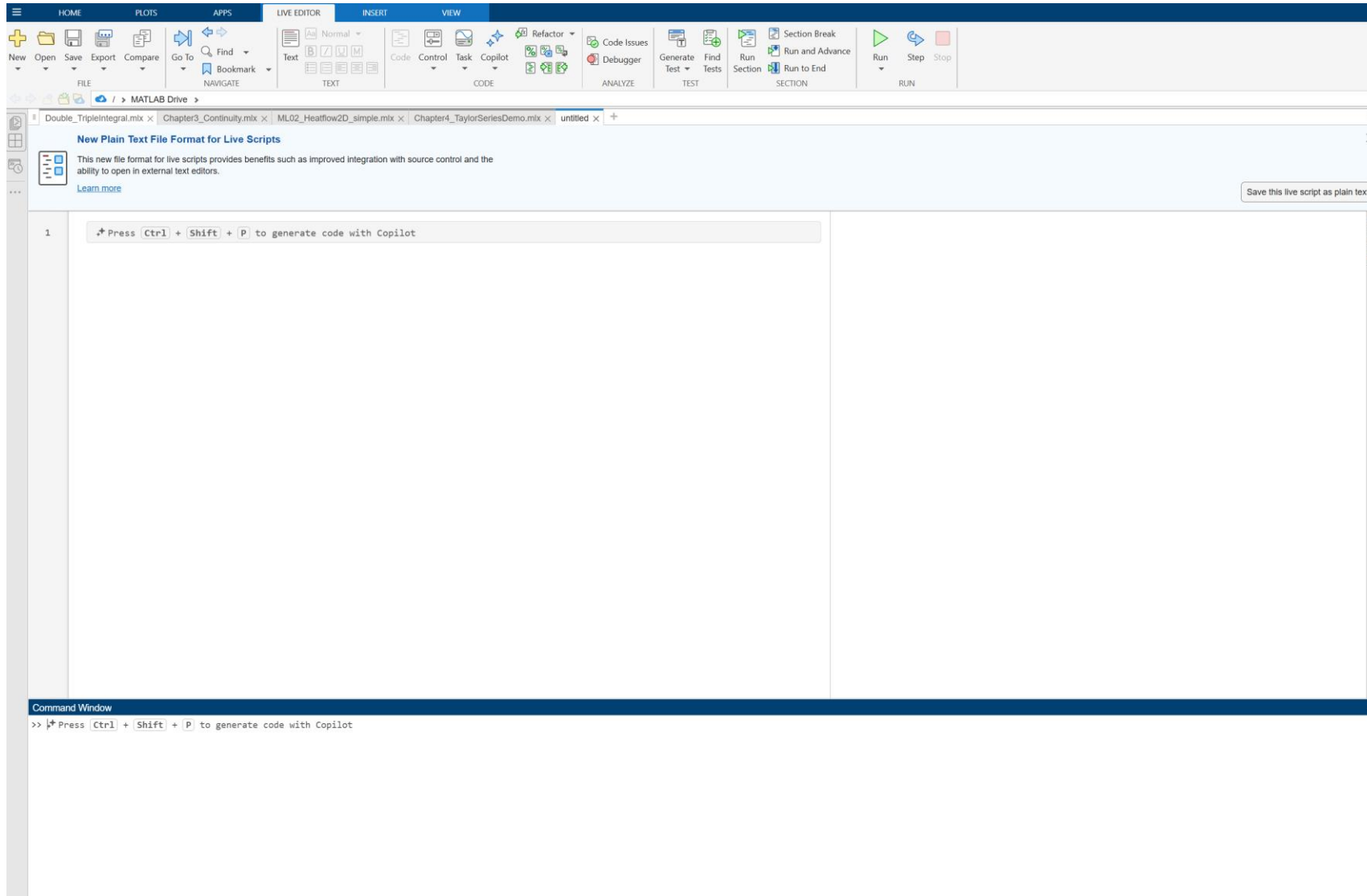
[Start](#)

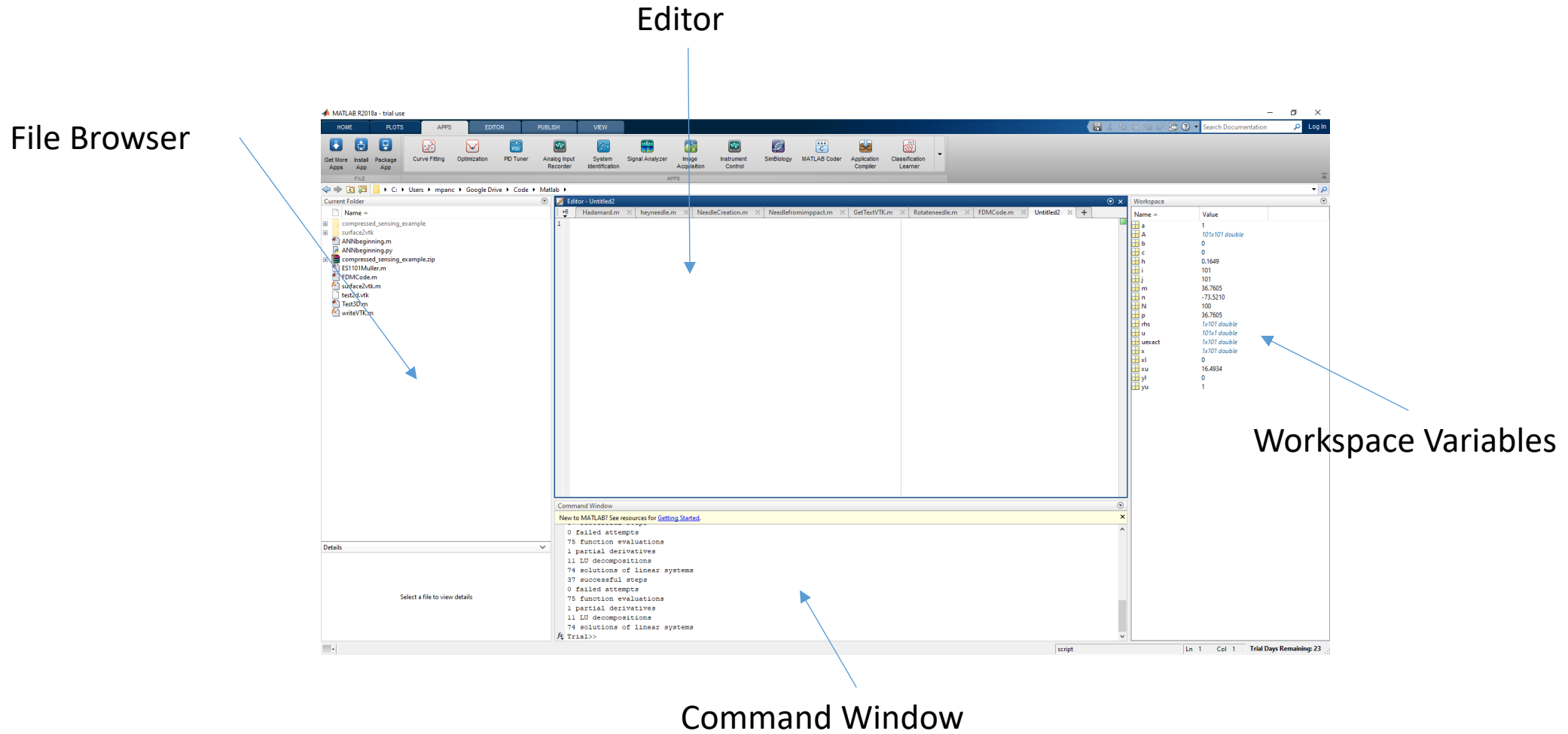
Machine Learning Onramp
Unlimited Access

[Start](#)

[View more](#)
[View all courses](#)



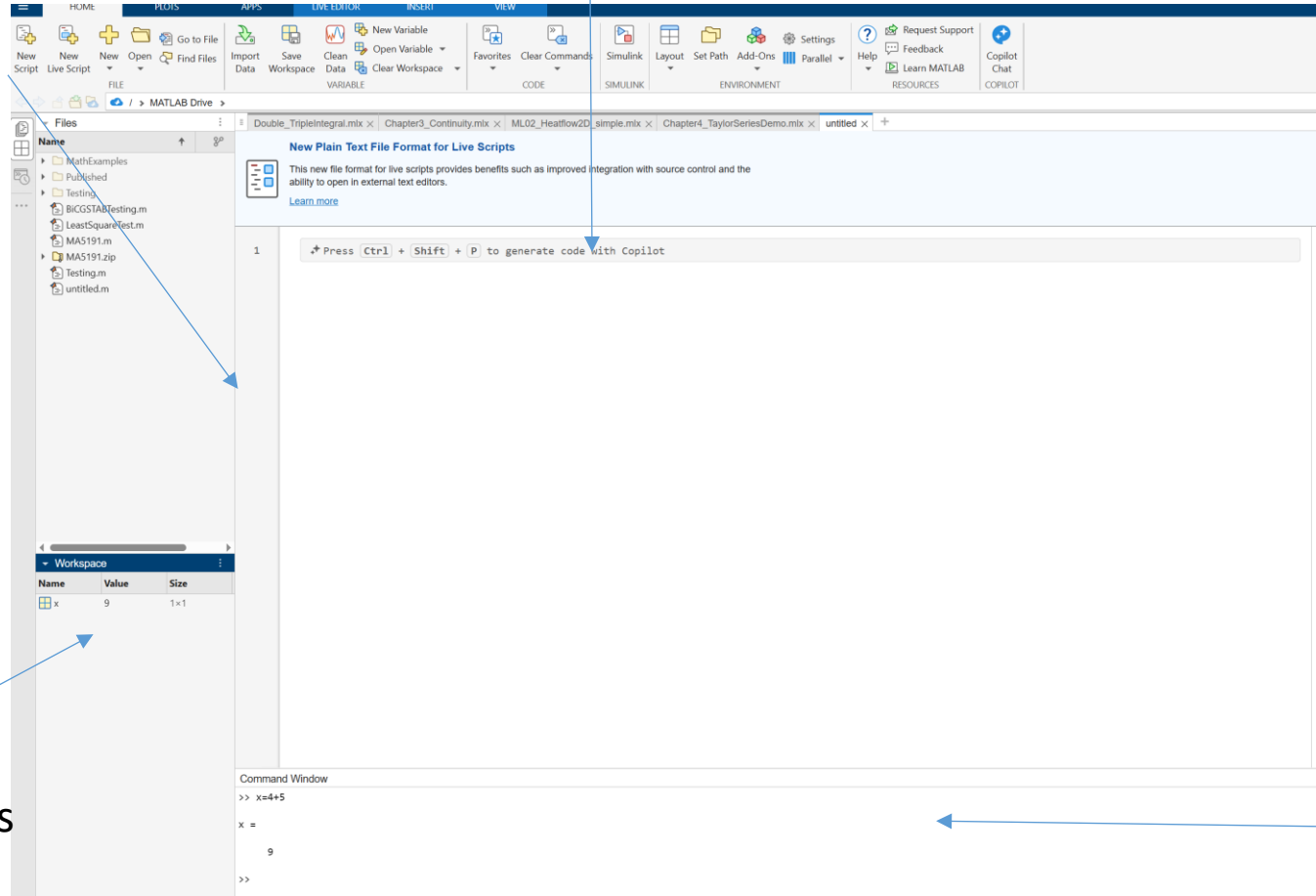




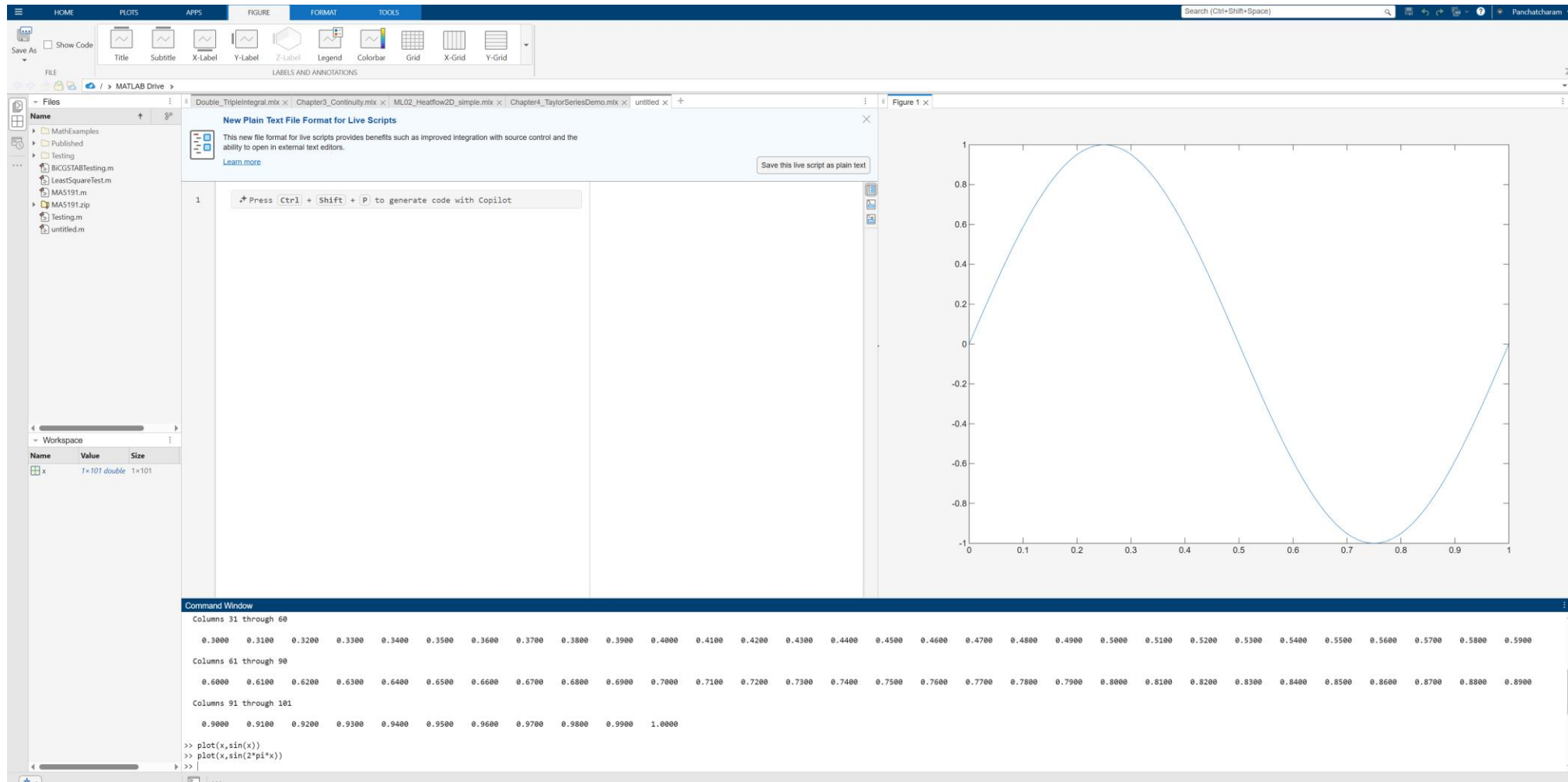
Editor

File Browser

Workspace Variables



Command Window



```
>> variable name = a value (or an expression)
```

Example:

```
X = expression
```

Symbol	Operation	Example
+	Addition	2+3
-	Subtraction	2-3
*	Multiplication	2*3
/	Division	2/3
^	Exponent	2^3

- If you enter incorrect expression and run the file, you will get an error message saying type of error and the line number.
- Correct the error to proceed further

Example

$x=3+5$

$2x$

```
New to MATLAB? See resources for Getting Started.  
Trial>> TestingCodeCourse  
Error: File: TestingCodeCourse.m Line: 2 Column: 2  
Invalid expression. Check for missing multiplication operator, missing or unbalanced delimiters, or other syntax error. To construct  
matrices, use brackets instead of parentheses.  
fx Trial>>
```

```
clear all  
clc  
close all  
who  
diary  
diary filename  
help sqrt  
doc plot  
lookfor inverse
```

Mathematical Functions

Function	Description	Function	Description
<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
<code>asin (x)</code>	Arcsin	<code>min (x)</code>	Minimum
<code>acos (x)</code>	Arc cosine	<code>ceil (x)</code>	Ceiling
<code>atan (x)</code>	Arc tangent	<code>floor (x)</code>	Floor
<code>exp (x)</code>	Exponential	<code>round (x)</code>	Round to nearest integer
<code>sqrt (x)</code>	Square root	<code>rem (x)</code>	Remainder
<code>log (x)</code>	Natural Log	<code>conj (x)</code>	Complex conjugate
<code>log10 (x)</code>	Common log		

Function	Description
<code>pi</code>	
<code>i, j</code>	Imaginary unit
<code>Inf</code>	infinity
<code>Nan</code>	Not a Number

```
V=[1 4 7 10 13] %Row Vector  
U=[1;2;3;4;5]%column vector  
W=U' % Transpose  
V(1:3) %slicing ans = 1 4 7  
V(3:end) %slicing ans =7 10 13  
V(:) %Complete vector (column)  
V(1:end) %Complete Vector (row)
```

Indexing
Starts at 1

End indices
are inclusive

```
A=[1 4 7; 10 13 14; 12 1 2] %3x3 matrix
```

```
A(2,1) %ans = 10
```

Indexing: $A(i,j)$, $i \geq 1$, $j \geq 1$

No negative or zero scripts

Remember the brackets not square brackets []

Instead ()

```
X=0:0.1:1
```

```
Ans
```

```
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
```

```
A=[1 4 7; 10 13 14; 12 1 2] %3x3 matrix
```

```
A(2,:)
```

```
Ans = 10 13 14
```

```
A(:,2:3)
```

```
4 7
```

```
13 14
```

```
1 2
```

```
A=[1 4 7; 10 13 14; 12 1 2] %3x3 matrix
```

```
A(:,2)
```

```
Ans =
```

```
4
```

```
13
```

```
1
```

```
A=[1 2 3; 4 5 6; 7 8 9] %3x3 matrix
```

```
B=A([2,3],[1,2])
```

```
B=
```

```
4 5
```

```
7 8
```

```
C=A([2,1,3],:)
```

```
C=
```

```
4 5 6
```

```
1 2 3
```

```
7 8 9
```

```
A=[1 2 3; 4 5 6; 7 8 9] %3x3 matrix
```

```
A(:)
```

1

4

7

2

5

3

3

6

9

```
A=[1 2 3; 4 5 6; 7 8 9] %3x3 matrix
```

```
A(3,:)=[]
```

```
A=
```

```
1 2 3
```

```
4 5 6
```

```
A=[A(1,:);A(2,:);[7 8 9]]
```

```
A=
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
A=[1 2 3; 4 5 6; 7 8 9] %3x3 matrix
```

```
size(A)
```

```
ans=
```

```
3 3
```

```
[m,n]=size(A)
```

```
A' % A transpose
```

Concatenating matrices

$A = [1 \ 2 \ 3; \ 4 \ 5 \ 6; \ 7 \ 8 \ 9]$ %3x3 matrix

$B = [A \ 10*A; \ -A \ [1 \ 0 \ 0; \ 0 \ 1 \ 0; \ 0 \ 0 \ 1]]$

B =

1	2	3	10	20	30
4	5	6	40	50	60
7	8	9	70	80	90
-1	-2	-3	1	0	0
-4	-5	-6	0	1	0
-7	-8	-9	0	0	1

`A= eye(m,n) % Diagonal 1, remaining zeros`

`A= eye(n) % Identity matrix`

`A=zeros(m,n) %Zero matrix`

`A=ones(m,n) % All entries 1`

`diag(A) % diagonal elements of A`

`A=rand(m,n) %Random matrix`

`A=hilb(n) % Hilbert matrix`

`A=invhilb(n) %Inverse Hilbert matrix`

```
A= magic(n) %magic square
```

```
A=pascal(n) %Pascal matrix
```

```
A=hadamard(2^n) %Hadamard matrix
```

```
Y=linspace(a,b)
```

Generates 100 points with equal intervals including a and b

```
Y=linspace(a,b,n)
```

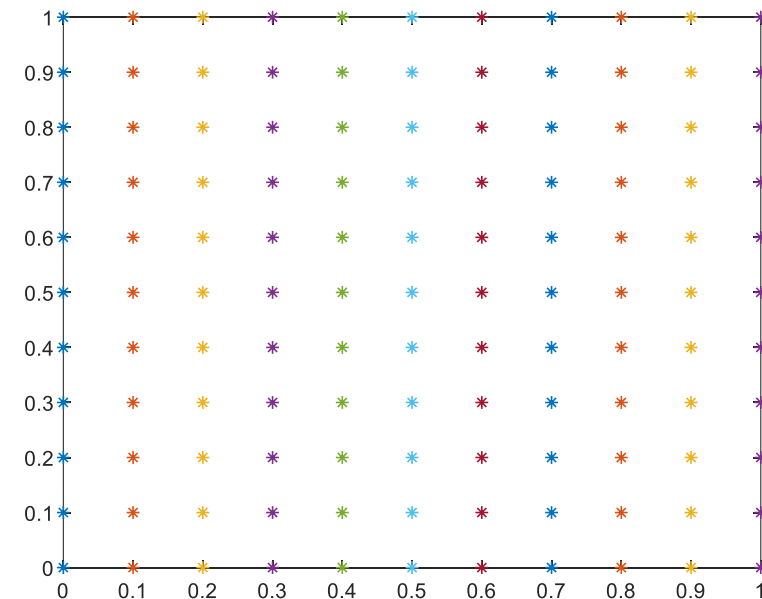
Generates n points

```
[X,Y] = meshgrid(x,y);
```

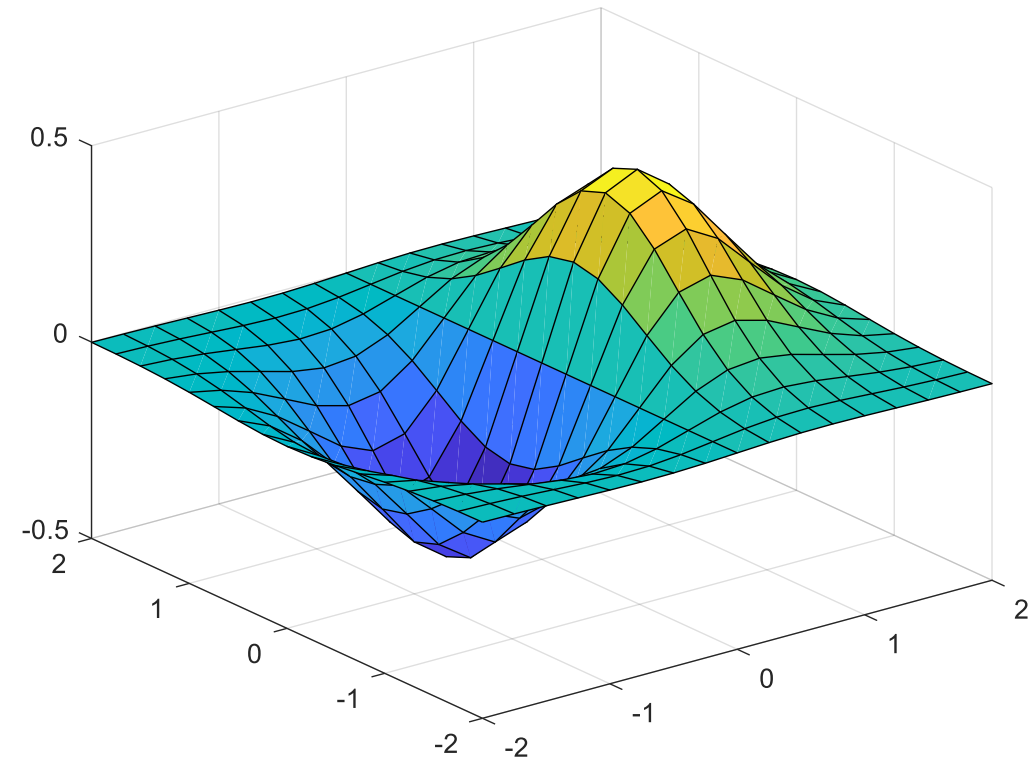
`meshgrid(x,y)` returns 2-D grid coordinates based on the coordinates contained in vectors `x` and `y`. `X` is a matrix where each row is a copy of `x`, and `Y` is a matrix where each column is a copy of `y`. The grid represented by the coordinates `X` and `Y` has `length(y)` rows and `length(x)` columns.

```
x = -1:0.1:1;
y = x;
[X,Y] = meshgrid(x);
plot(X,Y,'*')
```

https://in.mathworks.com/help/matlab/ref/meshgrid.html?searchHighlight=meshgrid&siteid=doc_srchtile



```
x = -2:0.25:2;  
y = x;  
[X,Y] = meshgrid(x);  
F = X.*exp(-X.^2-Y.^2);  
surf(X,Y,F)
```



https://in.mathworks.com/help/matlab/ref/meshgrid.html?searchHighlight=meshgrid&siteid=doc_srchtile

Array operations

Symbol	Operation	Example
.*	Element-by-element multiplication	C=A.*B
./	Element-by-element division	C=A./B
.\	Left Division	C=A.\B
.^	Element-by-element exponentiation	C=A.^B
+	Addition	C=A+B
-	Subtraction	C=A-B

A = [1 2 3]

B = [1 2 -1]

C = A .* B

A .^ 2

C = A .^ 2

C = A ./ B

Matrix operations

Symbol	Operation	Example
*	Matrix Matrix Multiplication	$C=A*B$
/	Division (mrdivide)	$C=A/B$
\	Solution	$C=A\backslash B$
+	Addition	$C=A+B$
-	Subtraction	$C=A-B$
^	Exponentiation	$C=A^2$

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

$$B = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}$$

$$C = A * B$$

$$A^2$$

$$C = A / B$$

Division (mrdivide): A/B

- If A and B are vectors, then A/B is the same as A./B
- If A is a square matrix and B is a matrix with n columns, then A/B is roughly $B * \text{inv}(A)$
- $A/B = (B' \backslash A')'$

Matrix Functions

Function	Operation	Example
det	Determinant	det(A)
diag	Diagonal matrix	diag(A)
eig	Eigenvalues and eigenvectors	eig(A)
inv	Inverse of a matrix	inv(A)
norm	Norm of a matrix	norm(A)
rank	Rank of a matrix	rank(A)
trace	Trace of a matrix	trace(A)
lu	LU factorization	
cond	Condition number of a matrix	

Function	Operation
cross	Vector cross product
dot	Vector dot product
sum	Sum of array elements
reshape	Reshape array
prod	Product of array elements
min	Minimum value of array
kron	Kronecker tensor product
sort	Sort elements

```
if expression
    statements
elseif expression
    statements
else
    statements
end
```

```
Disc=b*b-4*a*c
if discr<0
    disp('imaginary')
end
```

Relational and Logical Operators

Operator	Description	Example
>	Greater than	A>B
<	Less than	A=	Greater than or equal to	A>=B
<=	Less than or equal to	A<=B
==	Equal to	A==B
~=	Not equal to	A~=B
&	AND operator	A&B
	OR Operator	A B
~	NOT operator	~A

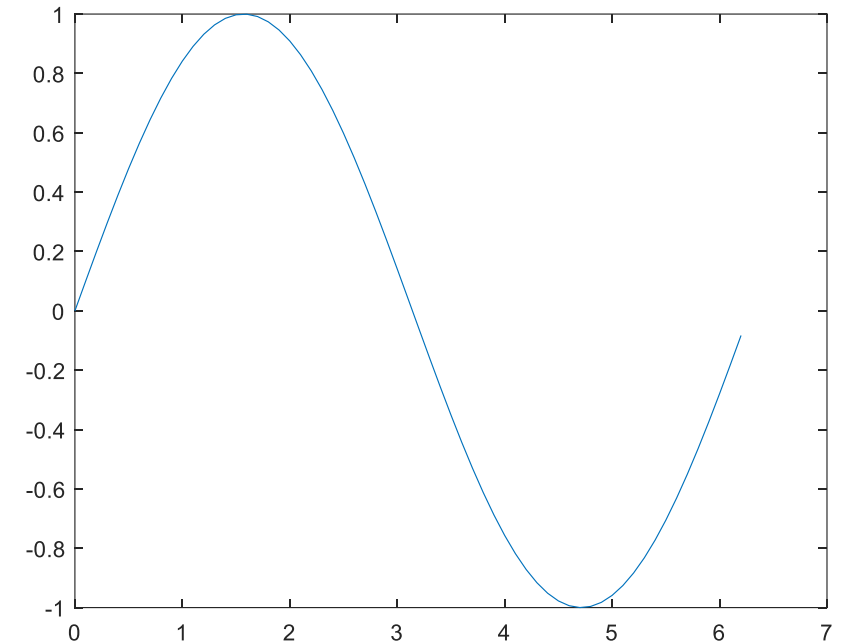
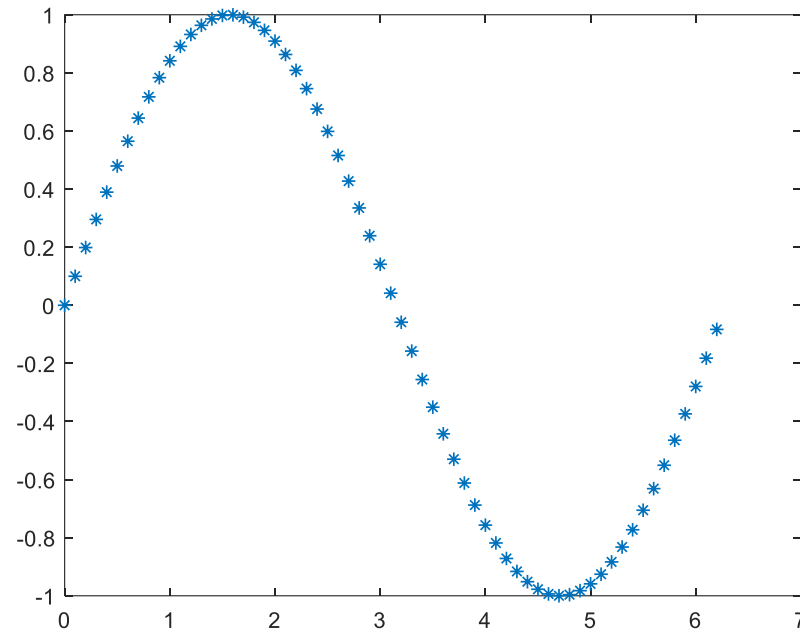
```
for variable = expression
    statements
end
```

```
for i = 1:10
    x=i*i
end
```

```
for i = 1:5
    for j=1:5
        if i==j
            A(i,j)=3
        elseif i==j+1 | i==j-1
            A(i,j)=1
        end
    end
end
end
```

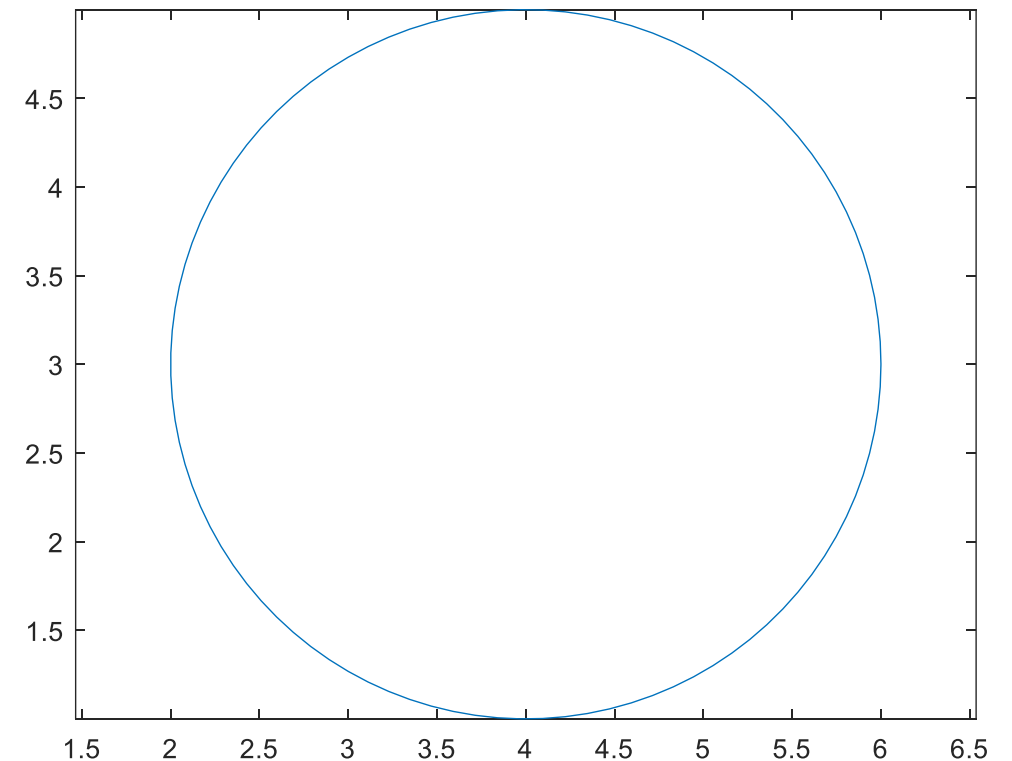
<https://in.mathworks.com/help/matlab/ref/plot.html>

```
X=0:0.1:2*pi  
Y1=sin(X)  
plot(X,Y1)  
plot(X,Y1,'*')
```



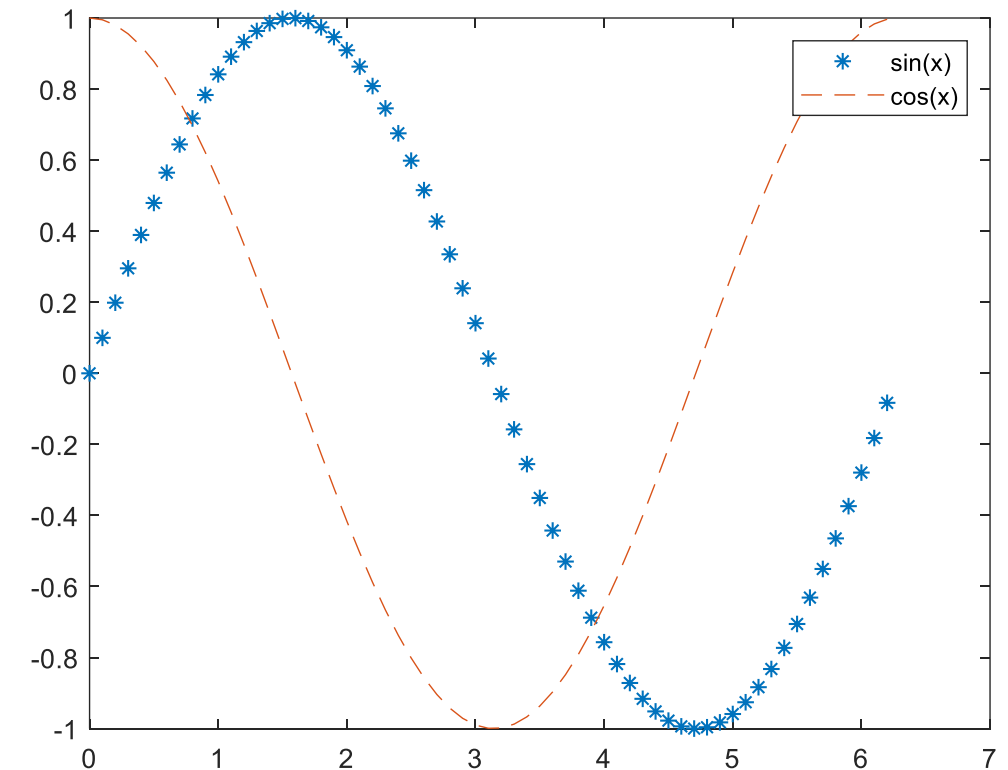
<https://in.mathworks.com/help/matlab/ref/plot.html>

```
r = 2;  
xc = 4;  
yc = 3;  
  
theta = linspace(0, 2*pi);  
x = r*cos(theta) + xc;  
y = r*sin(theta) + yc;  
plot(x,y)  
axis equal
```



<https://in.mathworks.com/help/matlab/ref/plot.html>

```
X=0:0.1:2*pi  
Y1=sin(X)  
Y2=cos(X)  
plot(X,Y1,'* ',X,Y2,'-- ')  
legend('sin(x)', 'cos(x)')
```



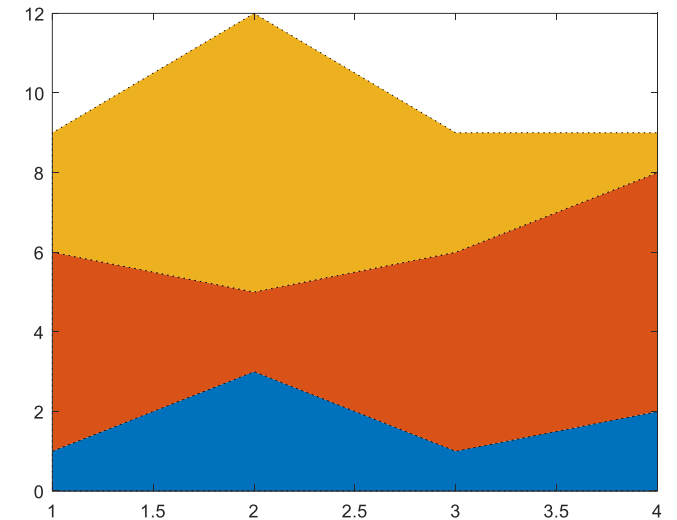
Symbol	Color
k	Black
r	Red
b	Blue
g	Green
c	Cyan
m	Magenta
y	Yellow

Symbol	Line Style
-	Solid
--	Dashed
:	Dotted
-.	Dash-dot

Symbol	Marker
+	Plus Sign
o	Circle
*	Asterisk
.	Point
x	Cross
S	Square
d	Diamond

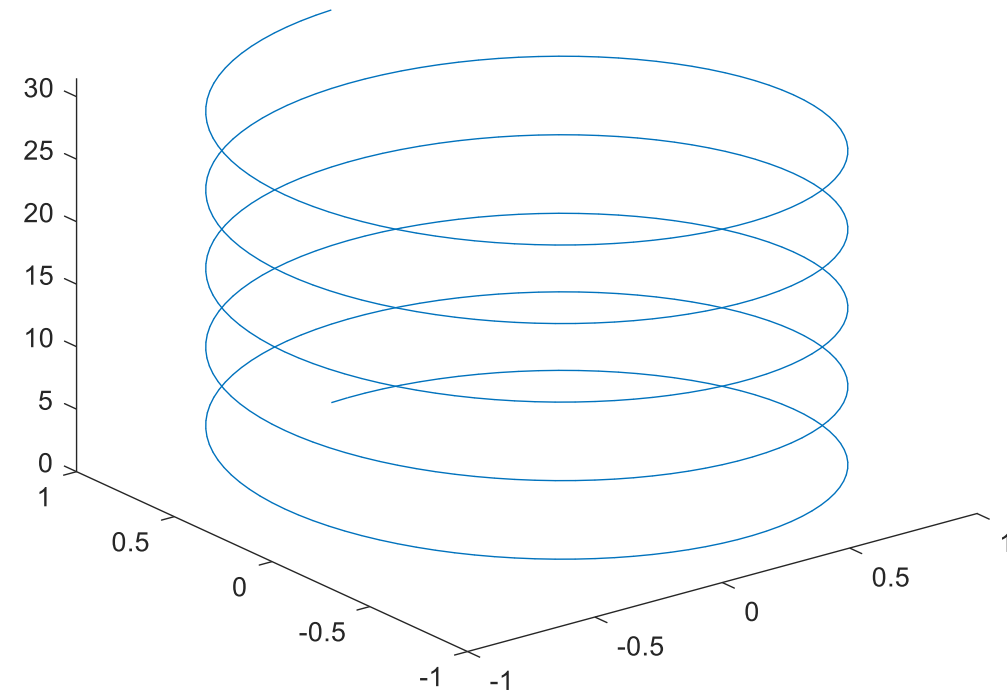
<https://in.mathworks.com/help/matlab/ref/area.html>

```
Y = [1, 5, 3;  
     3, 2, 7;  
     1, 5, 3;  
     2, 6, 1];  
h = area(Y, 'LineStyle', ':');
```

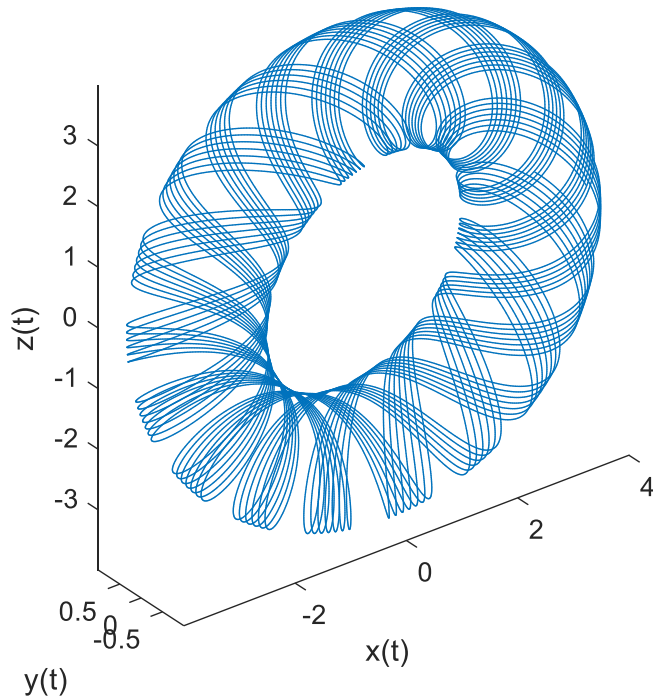


```
t=0:pi/50:10*pi  
st=sin(t)  
ct=cos(t)  
plot3(st,ct,t)
```

<https://in.mathworks.com/help/matlab/ref/plot3.html>

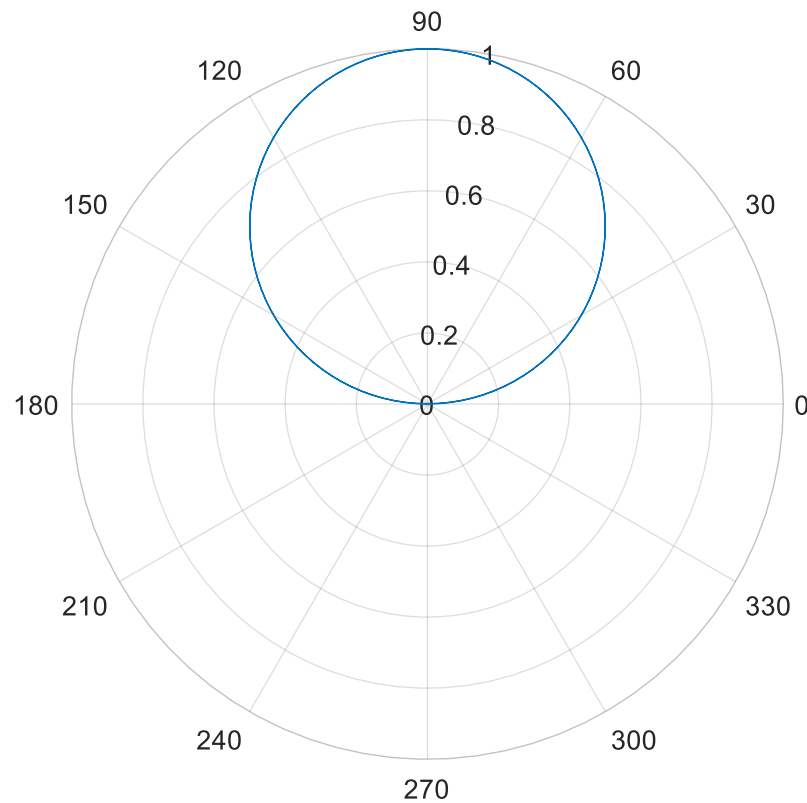


<https://in.mathworks.com/help/matlab/ref/plot3.html>



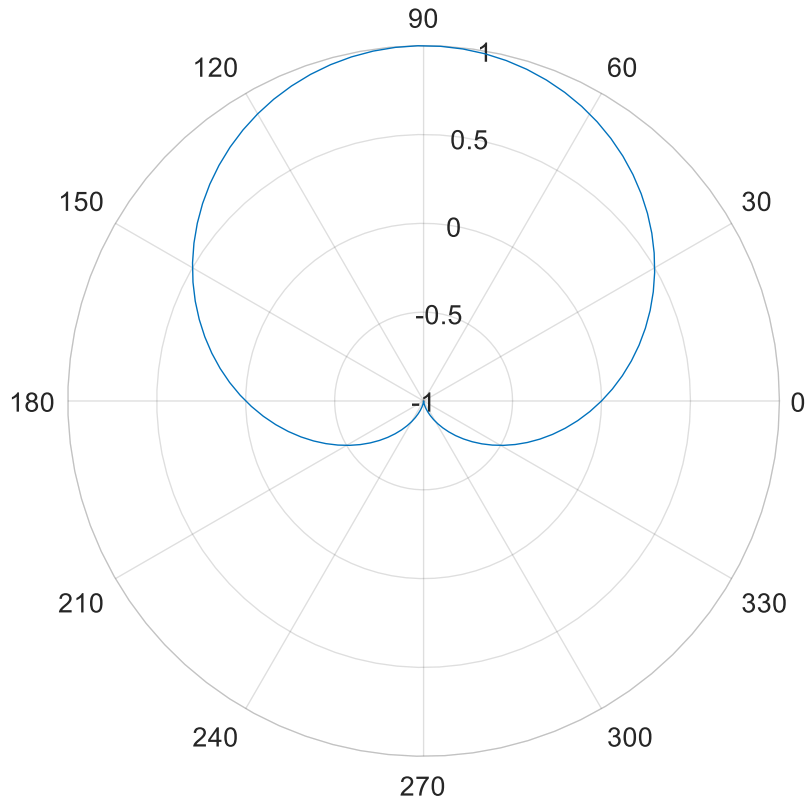
```
t = 0:pi/500:40*pi;  
xt = (3 + cos(sqrt(32)*t)).*cos(t);  
yt = sin(sqrt(32) * t);  
zt = (3 + cos(sqrt(32)*t)).*sin(t);  
plot3(xt,yt,zt)  
axis equal  
xlabel('x(t)')  
ylabel('y(t)')  
zlabel('z(t)')
```

<https://in.mathworks.com/help/matlab/ref/plot3.html>



```
theta = linspace(0,2*pi);  
rho = sin(theta);  
polarplot(theta,rho)
```

<https://in.mathworks.com/help/matlab/ref/polarplot.html>



```
theta = linspace(0,2*pi);  
rho = sin(theta);  
polarplot(theta,rho)  
rlim([-1 1])
```

<https://in.mathworks.com/help/matlab/ref/quiver.html>

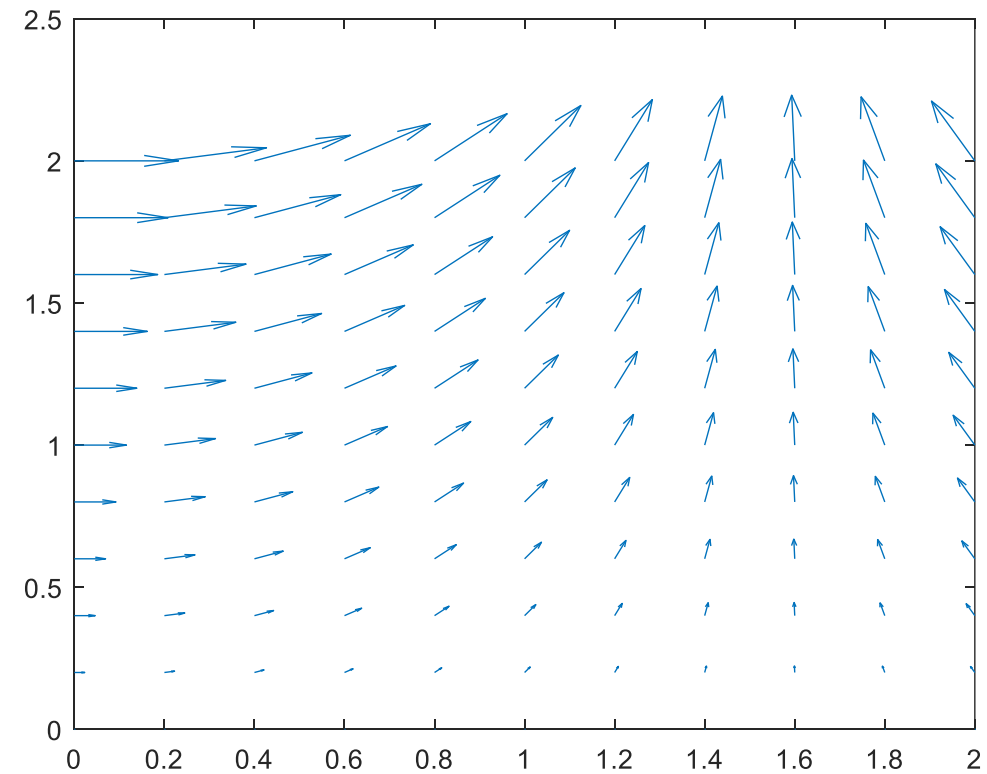
```
[x,y] = meshgrid(0:0.2:2,0:0.2:2);
```

```
u = cos(x).*y;
```

```
v = sin(x).*y;
```

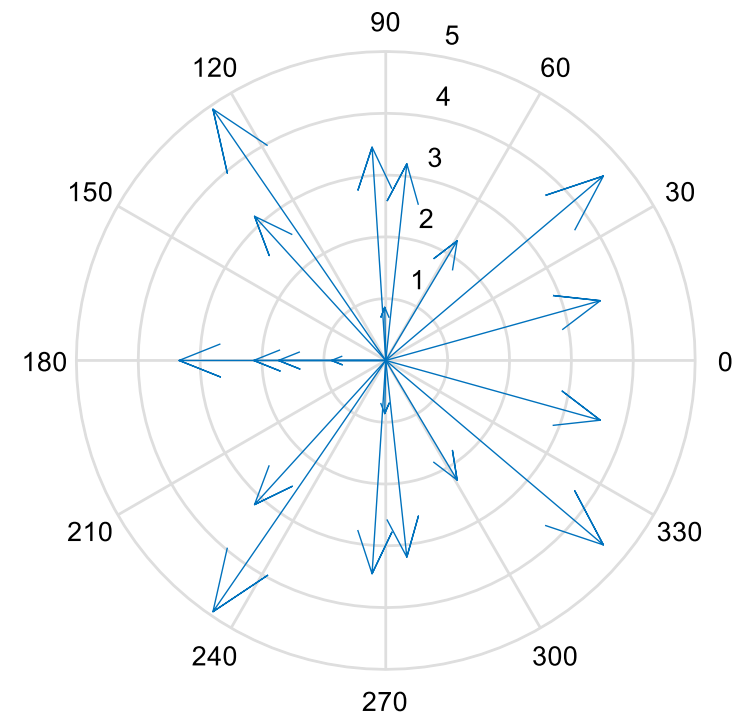
```
figure
```

```
quiver(x,y,u,v)
```



<https://in.mathworks.com/help/matlab/ref/quiver.html>

```
rng(0,'twister') % initialize  
random number generator  
M = randn(20,20);  
Z = eig(M);  
  
figure  
compass(Z)
```

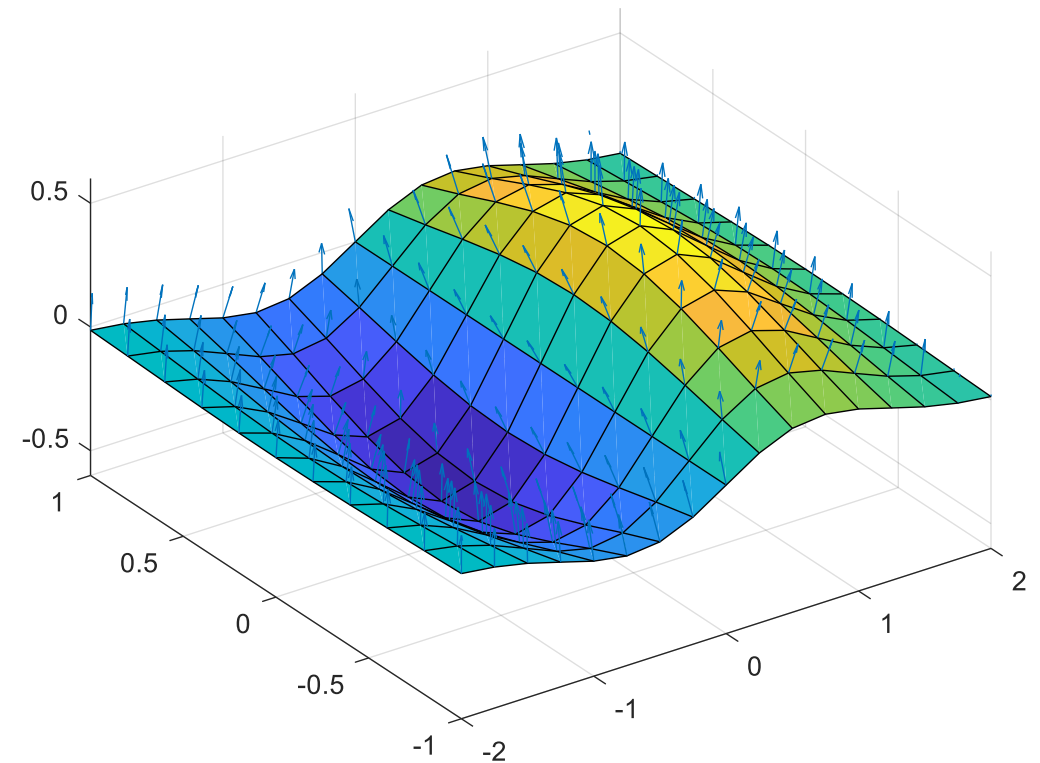


<https://in.mathworks.com/help/matlab/ref/quiver3.html>

```
[X,Y] = meshgrid(-2:0.25:2,-1:0.2:1);  
Z = X.* exp(-X.^2 - Y.^2);  
[U,V,W] = surfnorm(X,Y,Z);
```

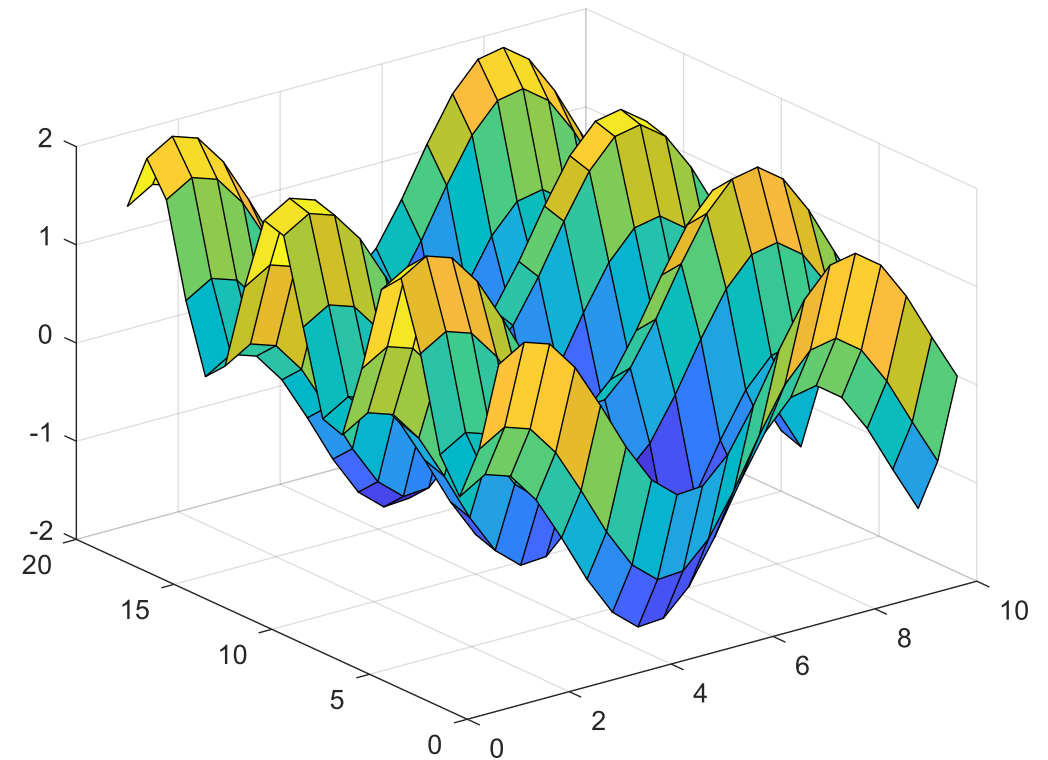
```
figure  
quiver3(X,Y,Z,U,V,W,0.5)
```

```
hold on  
surf(X,Y,Z)  
view(-35,45)  
axis([-2 2 -1 1 -.6 .6])  
hold off
```



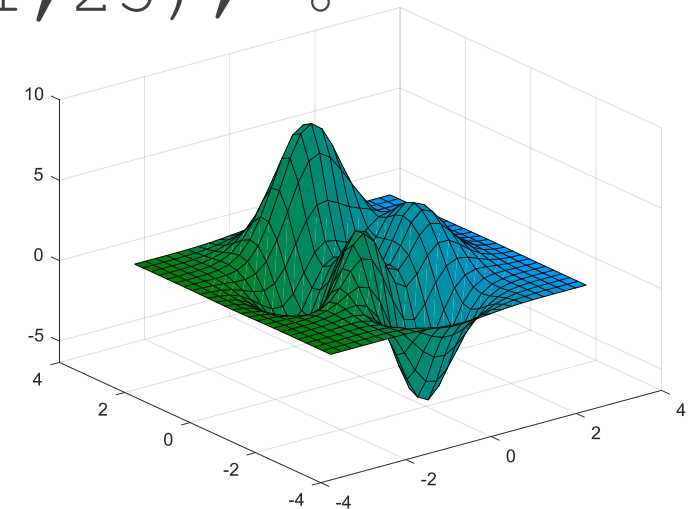
<https://in.mathworks.com/help/matlab/ref/surf.html>

```
[X,Y] = meshgrid(1:0.5:10,1:20);  
Z = sin(X) + cos(Y);  
surf(X,Y,Z)
```



<https://in.mathworks.com/help/matlab/ref/surf.html>

```
[X,Y,Z] = peaks(25);  
CO(:, :, 1) = zeros(25); % red  
CO(:, :, 2) =  
ones(25).*linspace(0.5,0.6,25); % green  
CO(:, :, 3) = ones(25).*linspace(0,1,25); %  
blue  
surf(X,Y,Z,CO)
```

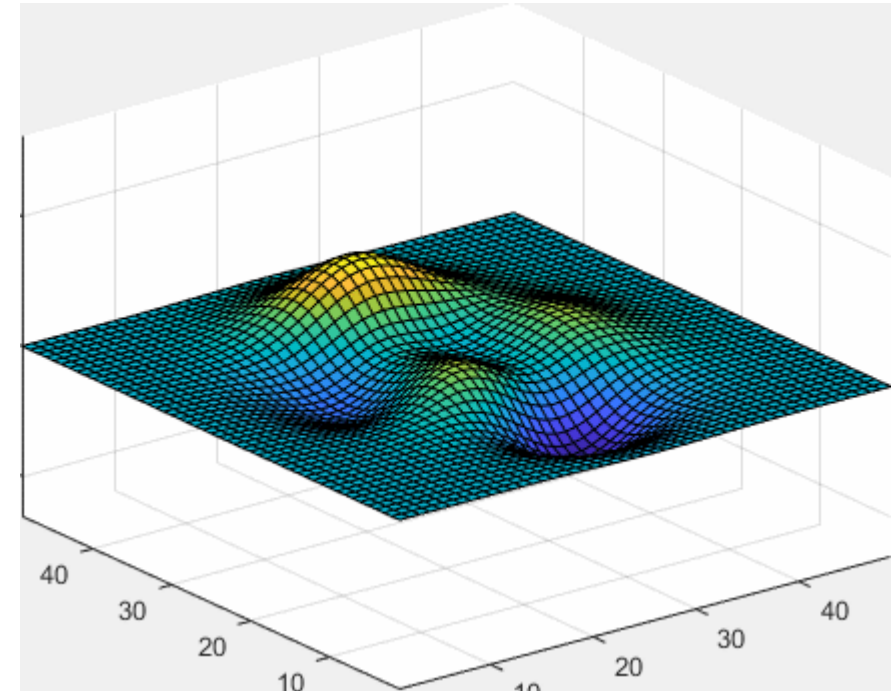


- stairs
- errorbar
- stackedplot
- loglog
- semilogx
- semilogy
- fplot
- fimplicit
- fplot3
- histogram
- histogram2
- morebins
- fewerbins
- histcounts
- histcounts2
- scatter
- scatter3
- binscatter
- scatterhistogram
- spy
- plotmatrix
- pie
- pie3
- heatmap
- bar
- bar3
- stem
- stem3
- geoplot
- geoscatter
- geobubble
- polarscatter
- compass
- ezpolar
- contourc
- contourf
- contour3
- contourslice
- feather
- surfc
- surfl
- mesh
- surfnorm
- meshc
- meshz
- hidden
- fsurf
- waterfall
- ribbon
- peaks
- cylinder
- ellipsoid
- sphere
- flow
- isocaps
- isonormals
- isosurface
- reducepath
- reducevolume
- subvolume
- fill
- fill3

```
figure
Z = peaks;
surf(Z)
axis tight manual
ax = gca;
ax.NextPlot = 'replaceChildren';

loops = 40;
F(loops) = struct('cdata', [], 'colormap', []);
for j = 1:loops
    X = sin(j*pi/10)*Z;
    surf(X,Z)
    drawnow
    F(j) = getframe;
end
```

https://in.mathworks.com/help/matlab/animation-1.html?s_tid=CRUX_lftnav



- <https://in.mathworks.com/help/index.html>

Diwali Exercises

Self-Paced Online Courses

[Online Courses Home](#) | [My Courses](#) | [Online Training Suite](#)

Statistics Onramp
[Start course](#)
[Share Course](#) | [Share Certificate & Progress](#) | [Quick Reference](#) | [Settings](#)

0%

This course covers the basics of using statistical methods for analysis. Through the use of visualizations, descriptive statistics, normal probability distributions, and hypothesis testing, you'll analyze the evolution of a finch species on the Galapagos Archipelago.

Course modules

- > [Introduction](#) 5 min
- > [Visualizing Data](#) 20 min
- > [Descriptive Statistics](#) 20 min
- > [Normal Distributions](#) 15 min
- > [Hypothesis Testing](#) 15 min
- > [Conclusion](#) 5 min

About this course

Format: [Self-paced](#)
Length: About 1 hour
Language: [English \(change\)](#)

Recommended prerequisites

MATLAB Onramp

Features

- Hands-on exercises with automated feedback
- Access to MATLAB through your web browser
- Shareable progress report and course certificate

Authored By:

Self-Paced Online Courses

[Online Courses Home](#) | [My Courses](#) | [Online Training Suite](#)

Introduction to Linear Algebra with MATLAB
[Resume course](#)
[Share Course](#) | [Share Certificate & Progress](#) | [Quick Reference](#) | [Settings](#)

6% Access expires 30-Jan-2026

Learn the basics of Linear Algebra in MATLAB®. Use matrix methods to analyze and solve linear systems.

Course modules

- > [Introduction](#) 5 min
- > [Solving Systems of Linear Equations](#) 1 hour
- > [Eigenvalue Decomposition](#) 16% | 0.5 hours
- > [Singular Value Decomposition](#) 11% | 0.5 hours
- > [Next Steps](#) 5 min

About this course

Format: [Self-paced](#)
Length: About 3 hours
Language: [English \(change\)](#)

Recommended prerequisites

MATLAB Onramp

Features

- Hands-on exercises with automated feedback
- Access to MATLAB through your web browser
- Shareable progress report and course certificate

- Complete the following Courses from MATLAB ACADEMY
 1. [MATLAB onramp](#)
 - <https://matlabacademy.mathworks.com/details/matlab-onramp/gettingstarted>
 2. [Statistics Onramp:](#)
 - <https://matlabacademy.mathworks.com/details/statistics-onramp/orst>
 3. [Introduction to Linear Algebra with MATLAB](#)
 - <https://matlabacademy.mathworks.com/details/introduction-to-linear-algebra-with-matlab/linalg>
 4. Machine Learning onramp
 - <https://matlabacademy.mathworks.com/details/machine-learning-onramp/machinelearning>
 5. How MATLAB Graphics Work
 - <https://matlabacademy.mathworks.com/details/how-matlab-graphics-work/otmlhgw>

- After completion of the course Share Both Certificates as a PDF file by clicking the following link

The screenshot shows the MATLAB Academy interface for the 'Statistics Onramp' course. A blue circle highlights the 'Share Certificate & Progress' dropdown menu, which is open. The menu contains two tabs: 'Course Certificate' and 'Progress Report'. Under the 'Course Certificate' tab, there is a 'Share:' section with a text input field containing the URL 'https://matlabacademy.mathworks.com' and a 'Copy' button. Below this are social media sharing icons for LinkedIn and Facebook. There is also a 'Download or print:' section with a 'View PDF' button and a note stating 'Note: PDF link is not shareable.' At the bottom of the menu, there is a link to 'Share progress with another user (e.g., your instructor)'.

- Visit this Course and understand how to read an image (imread command) and show it as a matrix
 - <https://matlabacademy.mathworks.com/details/image-processing-onramp/imageprocessing>
- Extract a grayscale image or RGB image and convert it to a matrix format
- Compute the SVD of this matrix (SVD is learnt from [Introduction to Linear Algebra with MATLAB](#))

Thank you