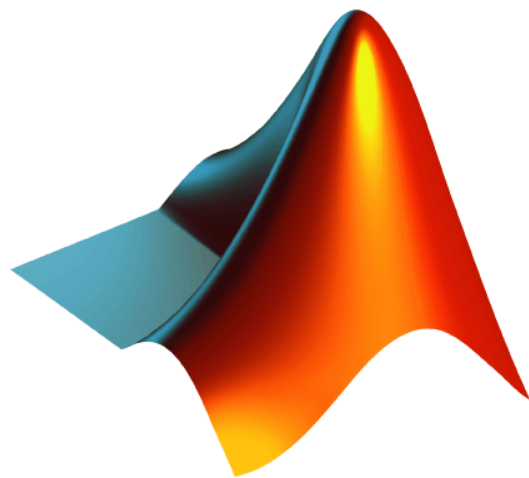


MATLAB

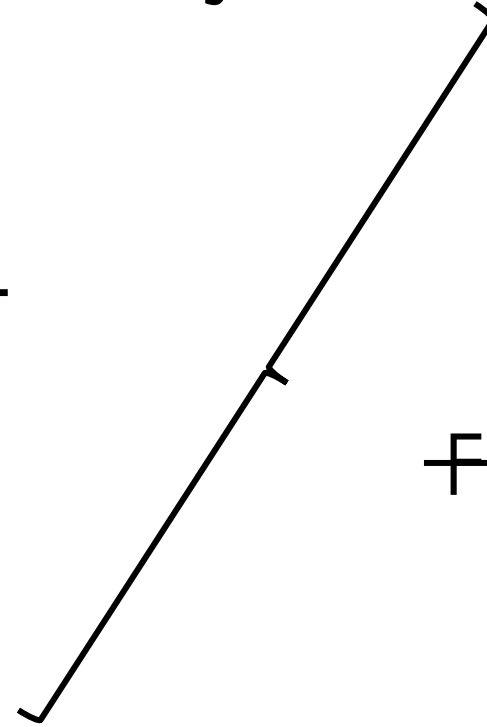


PROJECTS

Engineering Team Project

Integrated Project

Final Year Project

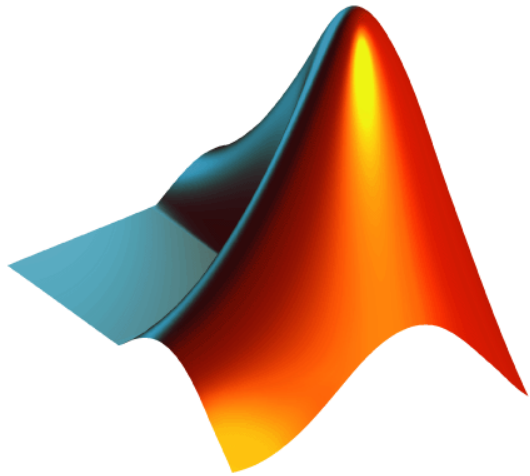


~~Fast Prototyping~~



Simulations

HOW?



MATLAB

GOAL FOR TODAY

Learn the basics of MATLAB
(You might use this in your assignments)

OVERVIEW

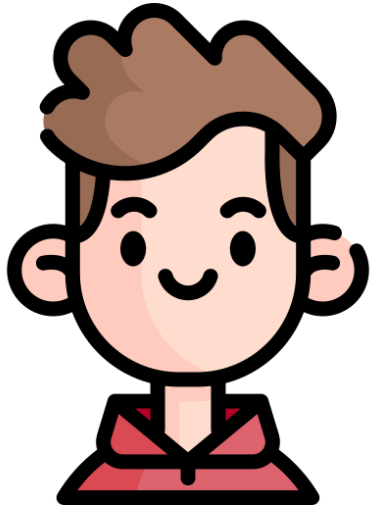
- Introduction to MATLAB
- Applications of MATLAB
- Hands-on MATLAB Activities
- Questions and Answer

WHAT IS MATLAB?

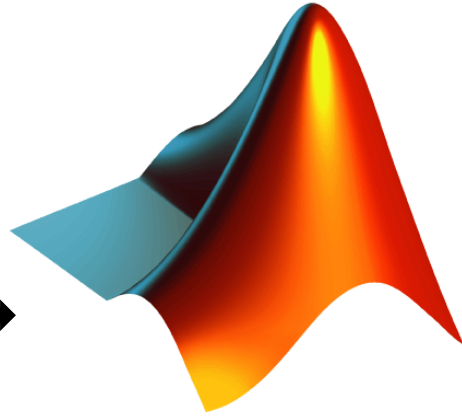
- MATLAB® (**Matrix Laboratory**) is a **programming platform** designed specifically for engineers and scientists.
- Written in Fortran in the late 1970s by **Cleve Moler**, it was a simple interactive matrix calculator.
- In 1983, Jack Little, who was an engineer, visited Moler for MATLAB. He sees the commercial potential in MATLAB, so Jack Little joined hands with Steve Bangert and Moler to **rewrite MATLAB in C Language**.
- This was also accompanied with the **formation of MathWorks** in the year 1984 to further enhance the development of MATLAB.

HISTORY OF MATLAB

1970s

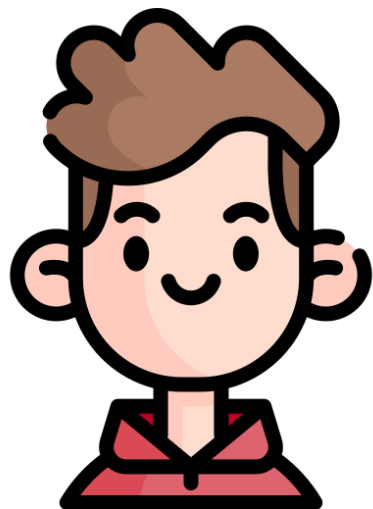


Cleve Moler



HISTORY OF MATLAB

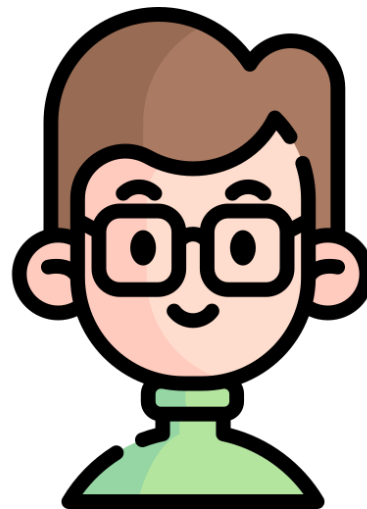
1983 - 1984



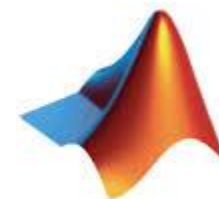
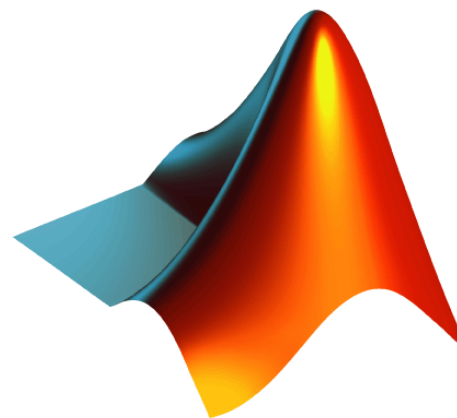
Cleve Moler



Jack Little



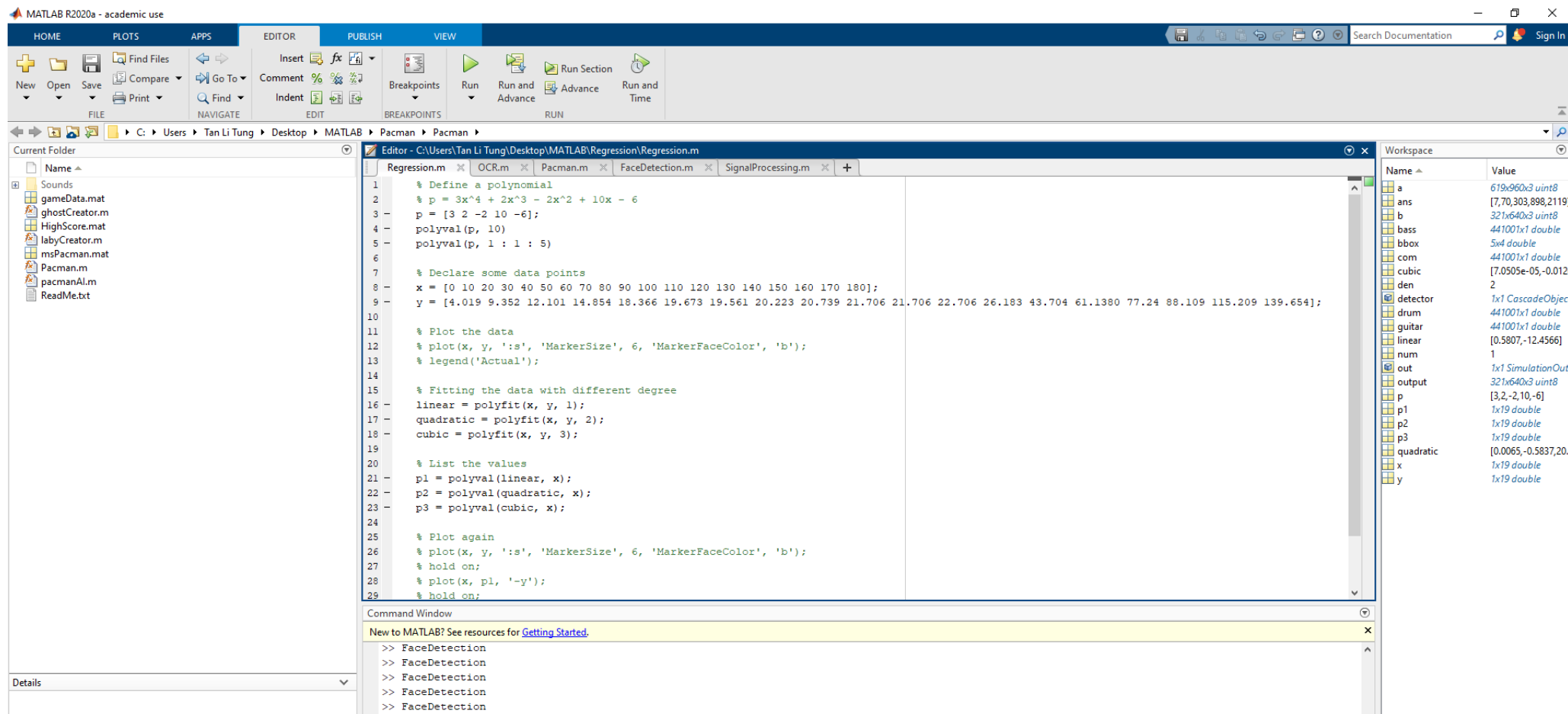
Steve Bangert



MathWorks®

WHY USE MATLAB?

- Easy and intuitive.



WHY USE MATLAB?

- Designed for Engineers and Scientist.



———— *ENGINEER* ————

WHY USE MATLAB?

- Lots of Toolboxes available (Lots of resources).

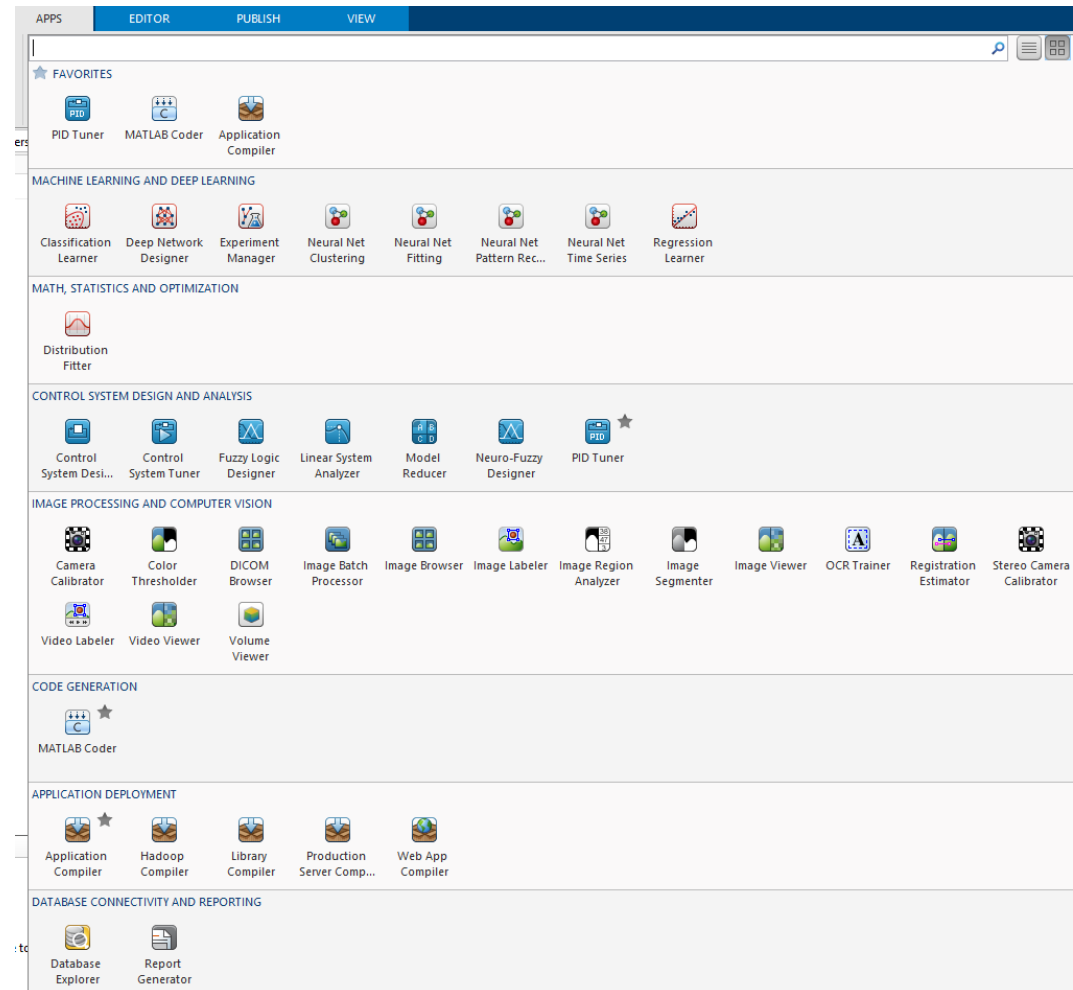
APPLICATIONS		
Signal Processing <hr/>	RF and Mixed Signal <hr/>	Automotive <hr/>
Signal Processing Toolbox	Antenna Toolbox	Model-Based Calibration Toolbox
Phased Array System Toolbox	RF Toolbox	Powertrain Blockset
DSP System Toolbox	RF Blockset	Vehicle Dynamics Blockset
Audio Toolbox	Mixed-Signal Blockset	Automated Driving Toolbox
Wavelet Toolbox	SerDes Toolbox	IEC Certification Kit (for ISO 26262 and IEC 61508)
Image Processing and Computer Vision <hr/>	Wireless Communications <hr/>	Vehicle Network Toolbox
Image Processing Toolbox	Communications Toolbox	AUTOSAR Blockset
Computer Vision Toolbox	WLAN Toolbox	RoadRunner
Lidar Toolbox	LTE Toolbox	RoadRunner Asset Library
Vision HDL Toolbox	5G Toolbox	RoadRunner Scene Builder
Control Systems <hr/>	Autonomous Systems <hr/>	Aerospace <hr/>
Control System Toolbox	Automated Driving Toolbox	Aerospace Blockset
System Identification Toolbox	Robotics System Toolbox	Aerospace Toolbox
Predictive Maintenance Toolbox	UAV Toolbox	UAV Toolbox
Robust Control Toolbox	Navigation Toolbox	DO Qualification Kit (for DO-178)
Model Predictive Control Toolbox	ROS Toolbox	
Fuzzy Logic Toolbox	Sensor Fusion and Tracking Toolbox	Computational Finance <hr/>
Simulink Control Design	RoadRunner	Econometrics Toolbox
Simulink Design Optimization	RoadRunner Asset Library	Financial Toolbox
Reinforcement Learning Toolbox	RoadRunner Scene Builder	Datafeed Toolbox
Motor Control Blockset		Database Toolbox
Test and Measurement <hr/>	FPGA, ASIC, and SoC Development <hr/>	Spreadsheet Link (for Microsoft Excel)
Data Acquisition Toolbox	HDL Coder	Financial Instruments Toolbox
Instrument Control Toolbox	HDL Verifier	Trading Toolbox
Image Acquisition Toolbox	Deep Learning HDL Toolbox	Risk Management Toolbox
OPC Toolbox	Wireless HDL Toolbox	
Vehicle Network Toolbox	Vision HDL Toolbox	Computational Biology <hr/>
		Bioinformatics Toolbox
		SimBiology

More at:

https://www.mathworks.com/products/index.html?s_tid=hp_fp_viewall

WHY USE MATLAB?

- MATLAB App (No need to code!).



WHY USE MATLAB?

- MATLAB integrates workflow. Fast. Reliable.

[Documentation](#) [Examples](#) [Functions](#) [Videos](#) [Answers](#)

[Trial Software](#) [Product Updates](#)

Calling MATLAB from Python

R2020b

Write Python® programs that work with MATLAB®

The MATLAB Engine API for Python provides a package for Python to call MATLAB as a computational engine. The engine supports the reference implementation (CPython). MATLAB supports versions 2.7, 3.6, 3.7, and 3.8. For more information, see [Versions of Python Supported by MATLAB Products by Release](#).

- To install and start the engine, see [Get Started with MATLAB Engine API for Python](#).
- To call Python functions from MATLAB, see [Python Libraries in MATLAB](#).

Engine applications require an installed version of MATLAB; you cannot run the MATLAB engine on a machine that only has the MATLAB Runtime.

Call MATLAB Functions from C++

R2020b

Call MATLAB® functions from C++ using the `feval` and `fevalAsync` member functions of the `matlab::engine::MATLABEngine` class. Use these functions when you want to pass function arguments from C++ to MATLAB and to return the result of the function execution to C++. These member functions work like the MATLAB `feval` function.

To call a MATLAB function:

- Pass the function name as a `matlab::engine::String`.
- Define the input arguments required by the MATLAB function. You can use either native C++ data types or the MATLAB Data API. For more information, see [MATLAB Data API](#).
- Specify the number of outputs expected from the MATLAB function. One output is the default. For more information, see [Call Function with Multiple Returned Arguments and Control Number of Outputs](#).
- Define the appropriate returned type for the results of the MATLAB function.
- Use stream buffers to redirect standard output and standard error from the MATLAB command window to C++. For more information, see [Redirect MATLAB Command Window Output to C++](#).

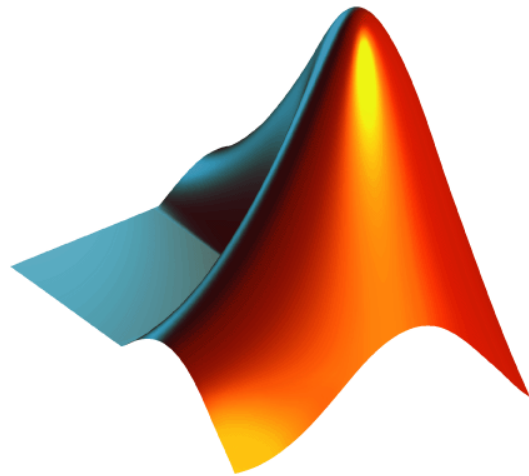
APPLICATIONS

- Matrix calculations.
- Plots of graphs.
- Machine learning.
- Signal processing.
- Face recognition.
- Optical Character Recognition (OCR).
- And many more.....

APPLICATIONS

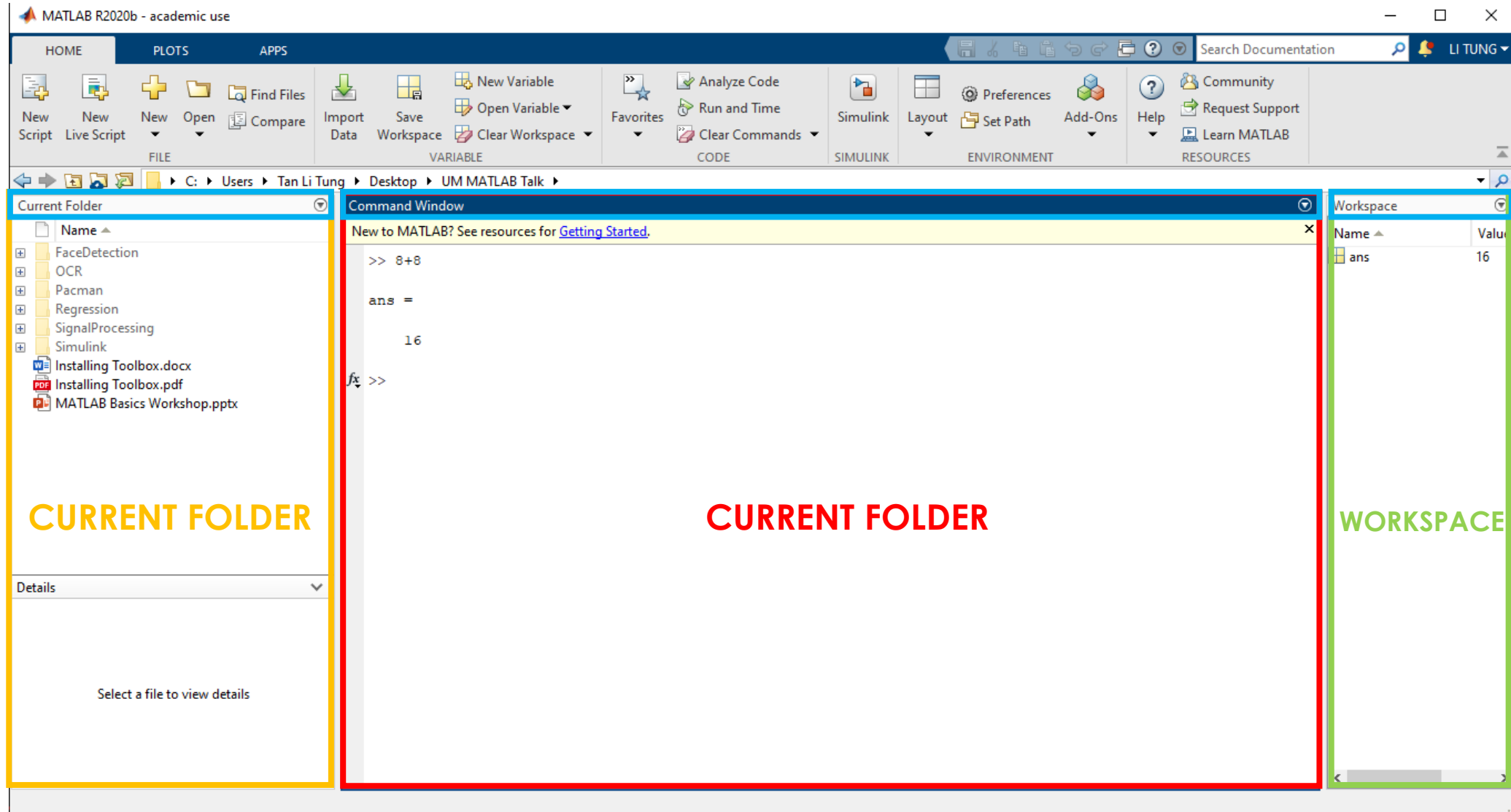
- Matrix calculations.
- Plots of graphs.
- Machine learning.
- Signal processing.
- Face recognition.
- Optical Character Recognition (OCR).
- And many more.....

HANDS-ON



MATLAB INTERFACE

MATLAB INTERFACE



****Change the layout by dragging the light blue boxes.**

BASIC OPERATIONS

BASIC OPERATIONS

- Assigning Variables (variableName = assignedValue)

```
> a = 3;
```

- Clear the command window

```
> clc
```

- Clear all variables

```
> clear all
```

- Clear all variables (clear variableName)

```
> clear a
```

- See the details of the variables

```
> whos
```

- **Variables are CASE SENSITIVE**
- **All variables are matrix**

VARIABLES

DATA TYPES

- Scalar

```
> variableName = assignedValue;
```

```
> a = 3;
```

- Matrix

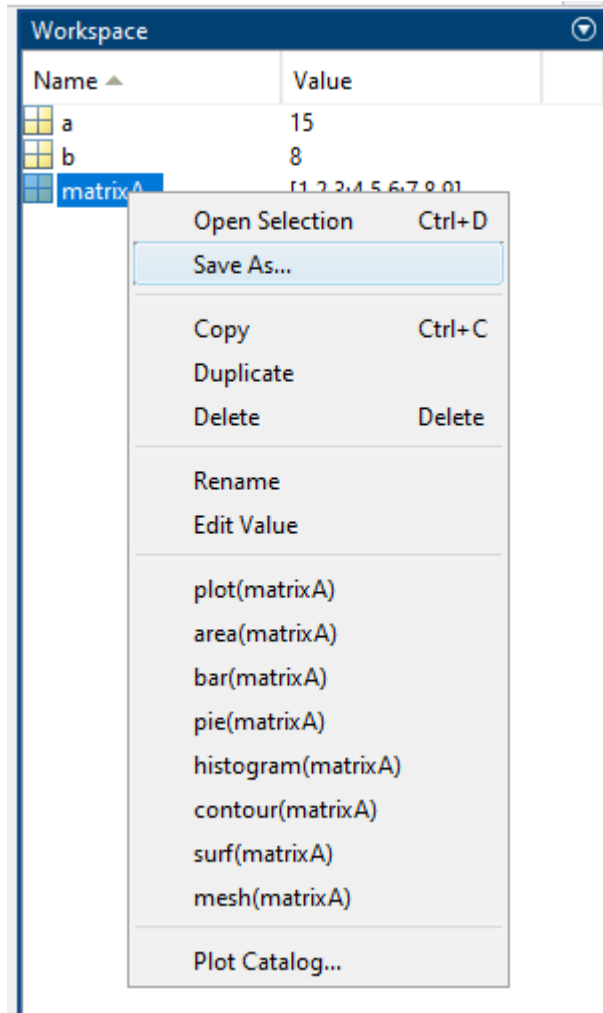
```
> matrixName = [a11 a12 ... a1n; a21 a22 ... a2n];
```

```
> matrixA = [1 2 3; 4 5 6; 7 8 9];
```

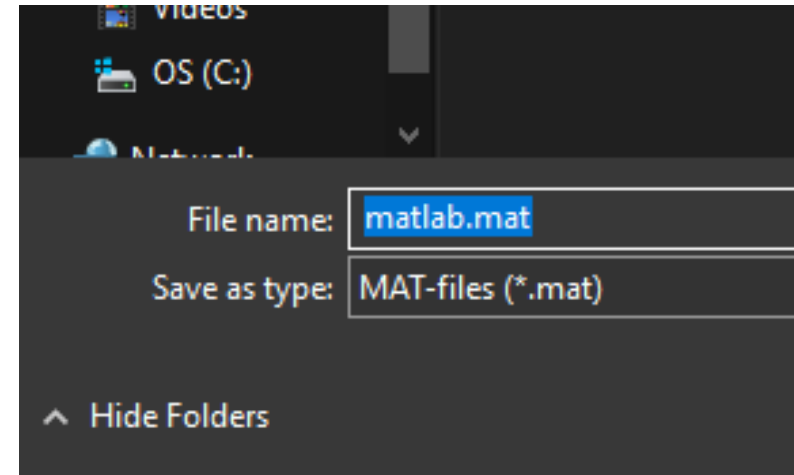
$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

*** Variables are CASE SENSITIVE**

SAVE THE VARIABLES



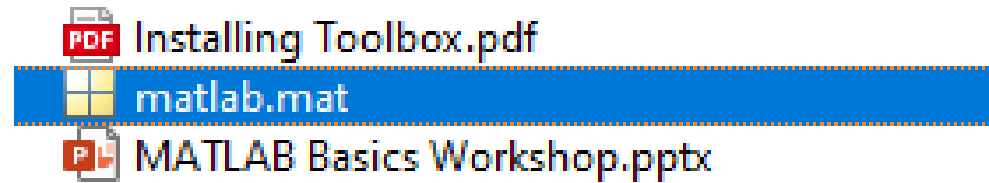
Select the variable(s) that you want to save, right click "Save As..."



Name the file and make sure it have extension of .mat

LOAD THE VARIABLES

Navigate to the folder of the .mat file
and select the file



The variable(s) will be loaded into
workspace

 matrixA

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

The image features a white background with decorative geometric elements in the corners. In the top right, there are overlapping yellow and light yellow triangles. In the bottom left, there are overlapping blue and light blue triangles. The word "SCALAR" is centered in the middle of the page.

SCALAR

SCALAR OPERATION

- + - * / ^
- pow, sqrt
- log, exp
- sin, cos, tan
- asin, acos, atan
- round, ceil, floor

Special Variable

Variable Name	Meaning
pi	$\pi = 3.1415926\dots$
eps	Machine precision
i	Imaginary unit ($\sqrt{-1}$)
inf	Infinity
NaN	Not a number (eg, $\frac{0}{0}$)
ans	Last displayed result
end	Last array element
realmax	Largest real number
intmax	Largest integer

The image features a white background with decorative geometric elements in the corners. In the top right, there are overlapping yellow and light yellow triangles. In the bottom left, there are overlapping light blue and medium blue triangles. The word "MATRIX" is centered in the middle of the page.

MATRIX

MATRIX OPERATION

- $+$ $-$ $*$ $/$ \wedge $'$
- `mpow`, `sqrtm`
- `logm`, `expm`
- `sum`, `prod`
- `cumsum`, `cumprod`
- `max`, `min`, `std`, `mean`
- `length`, `size`, `eig`

Command	Meaning
<code>ones(m, n)</code>	Create m x n matrix of 1's
<code>zeros(m, n)</code>	Create m x n matrix of 0's
<code>eye(n)</code>	Create n x n identity matrix
<code>diag(vec)</code>	Create diagonal matrix
<code>diag(A)</code>	Diagonal elements of A
<code>rand(m, n)</code>	Uniform random number matrix
<code>randn(m, n)</code>	Gaussian random number matrix
<code>magix(m)</code>	Magic square matrix
<code>hilb</code>	Hilbert matrix

ELEMENT WISE OPERATION

$+$ $-$ \cdot^* $\cdot/$ \cdot^{\wedge}

Consider $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ $B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$

$$A^*B = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

Matrix multiplication

$$A.\^*B = \begin{bmatrix} 5 & 12 \\ 21 & 32 \end{bmatrix}$$

Element Wise multiplication

SUBSET A MATRIX

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

- Subset an element

```
Recall matrixA = [1 2 3; 4 5 6; 7 8 9];
```

```
> variableName(row, column);
```

```
> matrixA(1, 2); % Ans = 2
```

- Subset a matrix

```
> variableName(row1:row2, column1:column2);
```

```
> matrixA(1:2, 2:3); % Ans = [2 3; 5 6]
```

GRAPH PLOTTING

GRAPH PLOTTING

- Using the PLOT function

```
> plot(x, y, 'lineStyle lineColor markerStyle');
```

```
> plot(x, y, '--ks');
```

- Example

```
> x = 0:0.5:10;
```

```
> y = x .^ 2;
```

```
> z = x .^ 3;
```

```
> plot(x, y, ':ks');
```

```
> hold on
```

```
> plot(x, z, '-.r^');
```

```
> grid on
```


LINE STYLE

Line Style	Description
-	Solid line
--	Dashed line
:	Dotted line
-.	Dash-dot line

Color	Description
y	yellow
m	magenta
c	cyan
r	red
g	green
b	blue
w	white
k	black

Marker	Description
'o'	Circle
'+'	Plus sign
'*'	Asterisk
'.'	Point
'x'	Cross
'_'	Horizontal line
' '	Vertical line
's'	Square
'd'	Diamond
'^'	Upward-pointing triangle
'v'	Downward-pointing triangle
'>'	Right-pointing triangle
'<'	Left-pointing triangle
'p'	Pentagram
'h'	Hexagram

GRAPH PLOTTING

- 3D Plot

```
> z = peaks(25);
```

```
> mesh(z)
```

```
> surf(z)
```

```
> surf1(z)
```

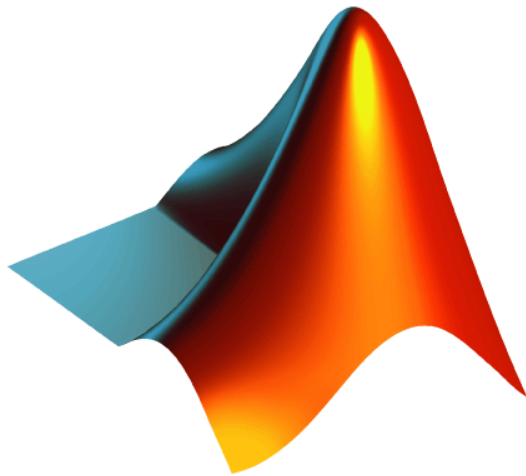
```
> colormap(jet) % change color map
```

GRAPH PLOTTING

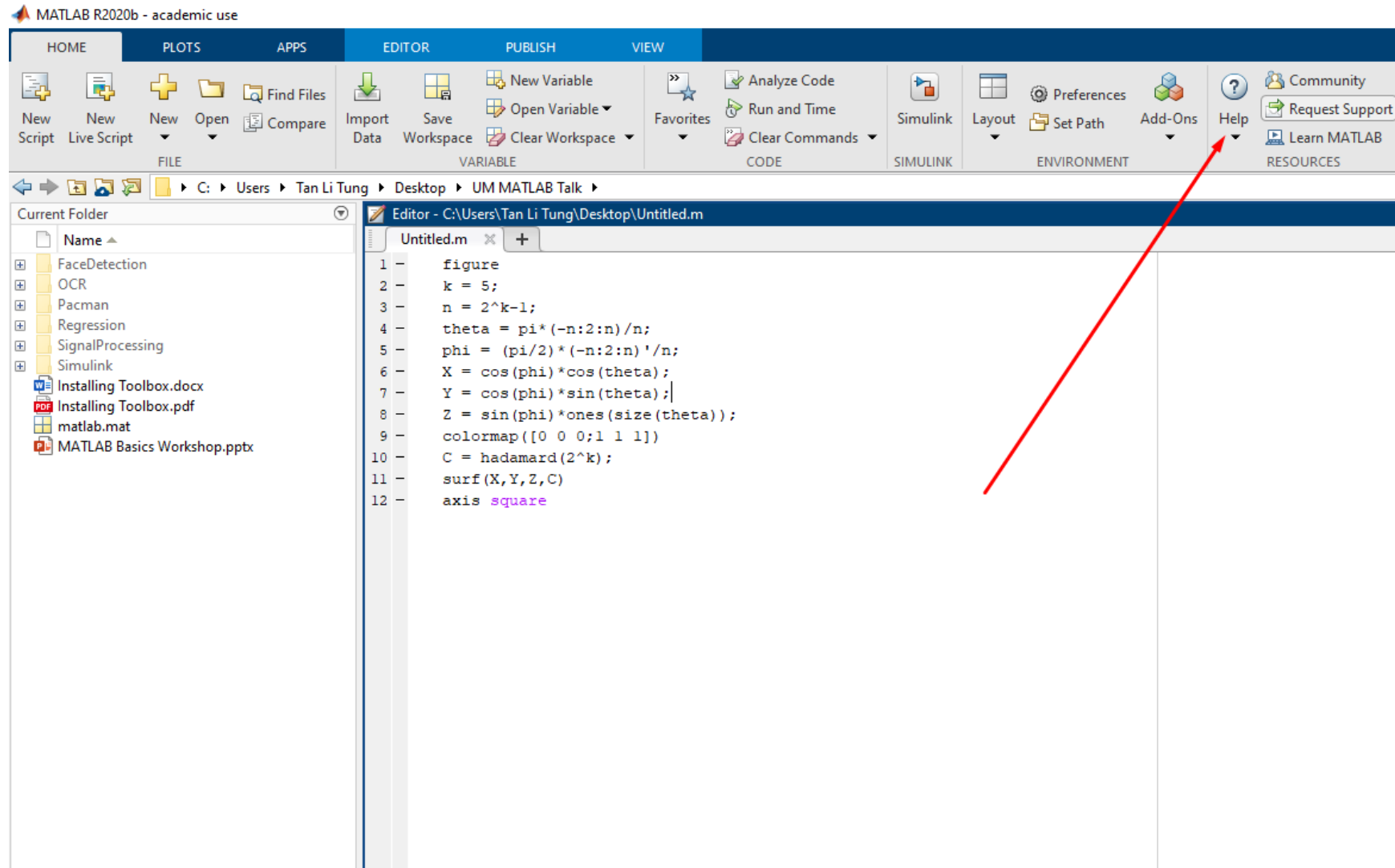
- Another Example

```
figure
k = 5;
n = 2^k-1;
theta = pi*(-n:2:n)/n;
phi = (pi/2)*(-n:2:n)'/n;
X = cos(phi)*cos(theta);
Y = cos(phi)*sin(theta);
Z = sin(phi)*ones(size(theta));
colormap([0 0 0;1 1 1])
C = hadamard(2^k);
surf(X,Y,Z,C)
axis square
```

RESOURCES



MATLAB DOCUMENTATION



MATLAB DOCUMENTATION

The screenshot shows the MATLAB documentation search interface. The search bar at the top contains the word "plot". The left sidebar shows filter options under "FILTER", including "All Products" and "Refine by Type" (Functions: 1950, Blocks: 136, Apps: 15, System Objects: 74, Examples: 1449, Help Topics: 2361, Release Notes: 15). Below this is "Refine by Product" (MATLAB: 1209, Simulink: 418, Computer Vision Toolbox: 149, Control System Toolbox: 321, Deep Learning Toolbox: 337, DSP System Toolbox: 271, Image Processing Toolbox: 143, Signal Processing Toolbox: 201). The main content area shows search results for "plot", with "Results 1 to 10 of 4,728". The first result is "plot - 2-D line plot", which is highlighted. It includes a brief description: "This MATLAB function creates a 2-D line plot of the data in Y versus the corresponding values in X." and a link to the documentation: "Documentation > MATLAB > Graphics > 2-D and 3-D Plots > Line Plots". The second result is "plot - Plot timeseries", the third is "plot - Plot polyshape", and the fourth is "plot - Plot alpha shape". Each result also includes a brief description and a link to the documentation. The bottom of the window shows the file path: "file:///C:/Program%20Files/MATLAB/R2020b/help/matlab/ref/plot.html".

Help

Search Results - plot

Search

plot

Filter

« All Products

Refine by Type

- Functions 1950
- Blocks 136
- Apps 15
- System Objects 74
- Examples 1449
- Help Topics 2361
- Release Notes 15

Refine by Product

- MATLAB 1209
- Simulink 418
- Computer Vision Toolbox 149
- Control System Toolbox 321
- Deep Learning Toolbox 337
- DSP System Toolbox 271
- Image Processing Toolbox 143
- Signal Processing Toolbox 201

Searched for **plot**

Results 1 to 10 of 4,728

plot - 2-D line plot *fx*

This MATLAB function creates a 2-D line **plot** of the data in Y versus the corresponding values in X.

[Documentation > MATLAB > Graphics > 2-D and 3-D Plots > Line Plots](#)

plot - Plot timeseries *fx*

This MATLAB function plots the timeseries data in ts against time, interpolating values between samples.

[Documentation > MATLAB > Language Fundamentals > Data Types > Time Series > Time Series Objects](#)

plot - Plot polyshape *fx*

This MATLAB function plots a polyshape object.

[Documentation > MATLAB > Mathematics > Computational Geometry > Elementary Polygons](#)

plot - Plot alpha shape *fx*

This MATLAB function plots alpha shape shp in a figure window.

[Documentation > MATLAB > Mathematics > Computational Geometry > Bounding Regions](#)


plot - Plot graph nodes and edges *fx*

This MATLAB function plots the nodes and edges in graph G.

[Documentation > MATLAB > Mathematics > Graph and Network Algorithms](#)

file:///C:/Program%20Files/MATLAB/R2020b/help/matlab/ref/plot.html


MATLAB DOCUMENTATION

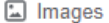



matlab plot 2d graph


×


🔍


 All

 Images

 Videos

 News

 Maps

 More

Settings

Tools

About 1,800,000 results (0.47 seconds)

www.mathworks.com › answers › 421897-plot-2d-graph ▾

Plot 2D graph - MATLAB Answers - MATLAB Central

Plot 2D graphs for each of the following functions on an interval $[-2\pi, 2\pi]$. Then **plot** all the **graphs** in the same chart and add a title to the **graph**. `sin(x)`; ...

1 answer

www.mathworks.com › MATLAB › Graphics ▾

2-D and 3-D Plots - MATLAB & Simulink - MathWorks

Line **Plots** Line **plots**, log **plots**, and function **plots** · Data Distribution **Plots** Histograms, pie charts, word clouds, and more · Discrete Data **Plots** Bar **graphs**, scatter ...

[Create Common 2-D Plots](#) · [Types of MATLAB Plots](#) · [Combine Contour Plot and ...](#)

www.mathworks.com › ... › Line Plots ▾


2-D line plot - MATLAB plot - MathWorks

plot(Y) creates a **2-D** line **plot** of the data in Y versus the index of each value. If Y is a vector, then the x-axis scale ranges from 1 to length(Y) . If Y is a matrix, then the **plot** function **plots** the columns of Y versus their row number.

[MATLAB Plot Gallery](#) · [Legend](#) · [Line Properties](#) · [Xlim](#)

You visited this page on 1/6/21.

Videos




2D Plotting in Matlab

25:08

2D Plotting in Matlab

YouTube · Christopher Lum

Oct 6, 2018



6

2D Graph Plotting

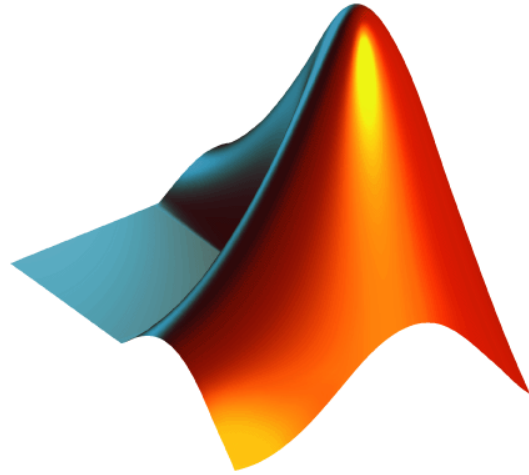
8:11

MATLAB Tutorial for Beginners 6 - 2D Graph Plotting in MATLAB

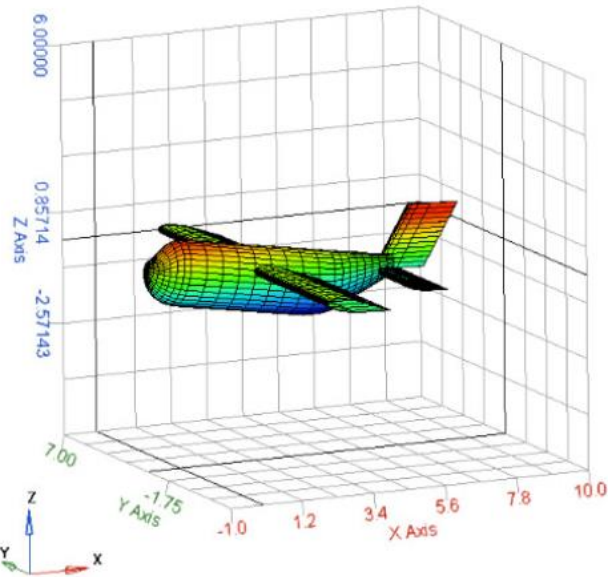
YouTube · KGP Talkie

May 17, 2018

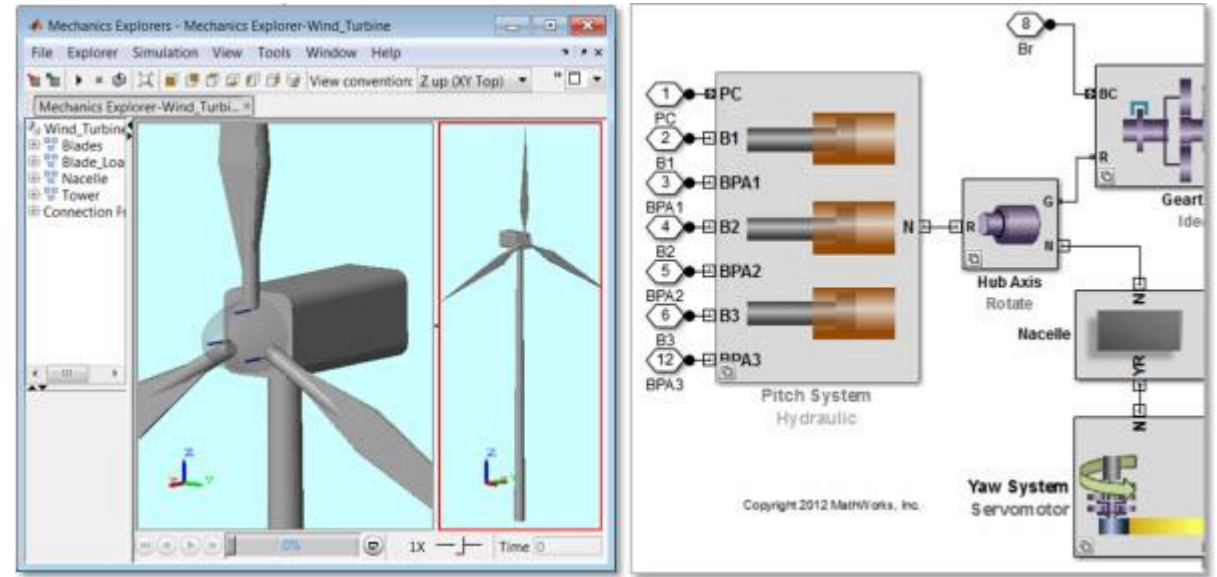
INDUSTRY USE



MATLAB USAGE IN THE INDUSTRIES

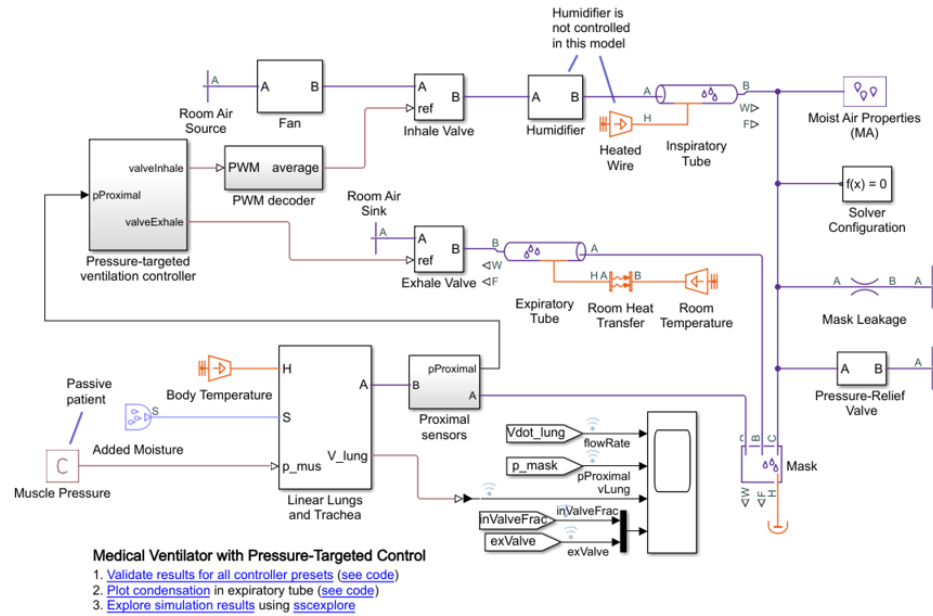


Aircraft System Modelling

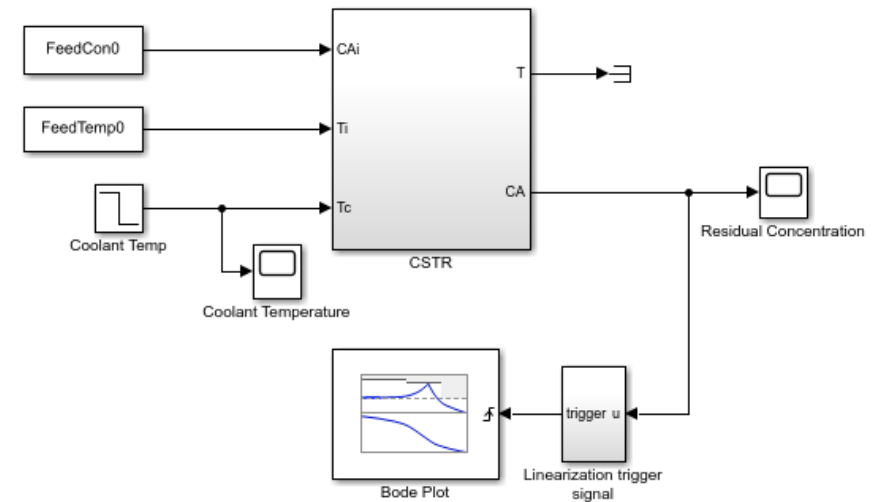


Wind Turbine Modelling

MATLAB USAGE IN THE INDUSTRIES



Medical Ventilation Modelling



Copyright 2010 The MathWorks, Inc.

Chemical Reactor Modelling

USEFUL LINKS

MATLAB Academy (Certification)

<https://matlabacademy.mathworks.com>

MATLAB for Beginners

https://www.youtube.com/watch?v=T_ekAD7U-wU

Solving ODEs in MATLAB

<https://www.mathworks.com/help/symbolic/solve-a-single-differential-equation.html>

THANK YOU

