Conclusion

Summary

Summary of MATLAB Onramp

Basic Syntax

Example	Description
`x = pi`	Create variables and assign values with the equal sign (`=`). The left side (`x`) is the variable name, and the right side (`pi`) is its value.
`y = sin(-5)`	Provide inputs to a function using parentheses.

Desktop Management

Function	Example	Description
`save`	`save data.mat`	Save your current workspace to a MAT-file.
`load`	`load data.mat`	Load the variables in a MAT-file to the workspace.
`clear`	`clear`	Clear all variables from the workspace.
`clc`	`clc`	Clear all text from the Command Window.
`format`	`format long`	Change how numeric output appears in the Command Window.

Array Types

Example	Description
`4`	scalar
`[3 5]`	row vector
`[1;3]`	column vector
`[3 4 5; 6 7 8]`	matrix

Evenly Spaced Vectors

Example	Description
`1:4`	Create a vector from `1` to `4`, spaced by `1`, using the colon operator (`:`).
`1:0.5:4`	Create a vector from `1` to `4`, spaced by `0.5`.
`linspace(1,10,5)`	Create a vector with `5` elements. The values are evenly spaced from `1` to `10`.

Matrix Creation

Example	Description
`rand(2)`	Create a square matrix with `2` rows and `2` columns.
`zeros(2,3)`	Create a rectangular matrix with `2` rows and `3` columns of `0`s.
`ones(2,3)`	Create a rectangular matrix with `2` rows and `3` columns of `1`s.

Array Indexing

Example	Description

Example	Description
`A(end,2)`	Access the element in the second column of the last row.
`A(2,:)`	Access the entire second row.
`A(1:3,:)`	Access all columns of the first three rows.
`A(2) = 11`	Change the value of the second element of an array to `11`.

Array Operations

Example	Description

`[1 2; 3 4] + 1`
`ans =`
`2 3`
`4 5`
	Perform array addition.

`[1 1; 1 1]*[2 2; 2 2]`
`ans =`
`4 4`
`4 4`
	Perform matrix multiplication.

`[1 1; 1 1].*[2 2; 2 2]`
`ans =`
`2 2`
`2 2`
	Perform element-wise multiplication.

Multiple Outputs

Example	Description
`[xrow,xcol] = size(x)`	Save the number of rows and columns in `x` to two different variables.
`[xMax,idx] = max(x)`	Calculate the maximum value of `x` and its corresponding index value.

Documentation

Example	Description
`doc randi`	Open the documentation page for the `randi` function.

Plots

Example	Description
`plot(x,y,"ro--",LineWidth=5)`	Plot a red (`r`) dashed (`--`) line with a circle (`o`) marker, with a heavy line width.
`hold on`	Add the next line to the existing plot.
`hold off`	Create new axes for the next plotted line.
`title("My Title")`	Add a title to a plot.

`xlabel("x")` `ylabel("y")`	Add labels to axes.
`legend("a","b","c")`	Add a legend to a plot.

Tables

Example	Description
`data.HeightYards`	Extract the variable `HeightYards` from the table `data`.
`data.HeightMeters = data.HeightYards*0.9144`	Derive a table variable from existing data.

Logical Indexing

Example	Description
`[5 10 15] > 12`	Compare the elements of a vector to the value `12`.
`v1(v1 > 6)`	Extract all elements of `v1` that are greater than `6`.
`x(x==999) = 1`	Replace all values in `x` that are equal to `999` with the value `1`.

Programming

Example	Description

`if x > 0.5`
`  y = 3`
`else`
`  y = 4`
`end`
	If `x` is greater than `0.5`, set `y` to `3`.
	
	Otherwise, set `y` to `4`.

`for c = 1:3`
`  disp(c)`
`end`
	The loop counter (`c`) progresses through the values `1:3` (`1`, `2`, and `3`).
	
	The loop body displays each value of `c`.