**MATLAB Introduction - Summary Page**

**Starting MATLAB** (if icon is not immediately obvious)Start Menu: search for MATLAB (or All Programs 🡺 MATLAB 🡺 R2019A 🡺 MATLAB)  
 Windows Explorer: C:\Program Files\MATLAB\R2019A\bin Choose: MATLAB

**Operators**

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | ***Scalar*** | ***Element-by-Element\****  ***(the dot-operators)*** | **Order of precedence:** |
| Subtraction | *-* | *-* | parentheses => inner to outer |
| Addition | *+* | *+* | *exponentiation => L to R* |
| Multiplication | *\** | *.\** | Multiplication/Division => L to R |
| Division | / | ./ | Addition/Subtraction => L to R |
| Exponentiation | ^ | .^ |  |

\* Element-by-element operators are one way to carry out operations involving two vectors (lists of numbers)

**Command Window Items**

Command Prompt >> Last answer is stored in “ans”

You can suppress showing result with ; To go to previous commands up arrow

Assignment Operator = Comment line – begin with %

The constant π piFormat command, compact switch format compact  
Help commandHelp *function name*

**Some Built in Functions:**

sum(x) - sum a set of values

mean(x) - find the mean of a list of values

std(x) - find the sample standard deviation for a list of values

sin(x), cos(x), tan(x) - calculate the trig. Function for an angle or list of angles in radians

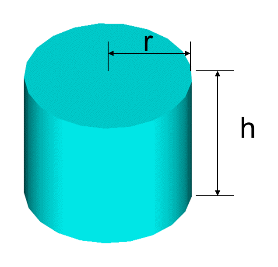
disp('text') - display the text that is in the single quotes in the command window

why - try it out

**Variables**

* Are simply names of storage locations, not true algebraic variables
* Values can be assigned to a variable or read from a variable
* = is the “assignment operator” it stores a value from its right into the variable on its left
* The left-hand side (LHS) of the equal sign must be a variable able to hold the value from the right-hand side (RHS)
* No operators are allowed on the LHS
* Variables can be a range of types (Scalar, Vector, Matrix, Text, Complex Structures …)
* Variable names start with a letter and can be made up of letters, numbers and underscores. They can be up to 63 characters long. Starting with a capital letter will avoid internal functions.

**Scripts**

* Any sequence of commands from the command window can be put in a script file (and run)
* Select **New** 🡺 **Script** from the home tab to open an editing window (or simply open Template.m).
* A **%** at the beginning of a line indicates this is a comment line and is not to be executed
* Begin with a comment header containing the file name, your name, date, & purpose of the script
* You can copy commands from the command history to a script. To select click at beginning and then shift-click at the end of the list of commands you wish to copy. Then copy and paste into script
* Add comments that define the variables used, their units and the logic of the program steps
* Save file to a program name that begins with a letter, is unique, and **has no spaces** in the name. MATLAB will automatically add a .m to the end of your file name.
* ****To “call” (run) a script type the file name (without the .m) at the command line and press enter.

**Example of a MATLAB Script**

**Problem:** Create a script that will calculate the area and volume of a cylinder. In an input section define r = 1.3 m and h = 2 m then use those variables in the calculations.

**Given:** r = 1.3 m, h = 2 m **Find:** the volume and surface area of the cylinder

**Known:** formulas for a cylinder 🡺 Volume:   
 Surface Area:

**Script**

Sections & notes

* Initial Comments
  + *Program filename.m*
  + Name, Date
  + ID & Purpose
* Variable List
  + Includes all variables
  + Defines meaning and units
* Input Section

Hard coded in this case

* Volume Calc. & display
  + uses disp() function to label
  + semicolon left off to display
* Surface Area Calc. & display

% Program CylScript.m

% S. Scott Moor Sept. 2012

% This program will calculate the Volume and

% Surface Area of a right circular cylinder   
% based on radius and height.

%

% Variables used:

% Inputs:

% r = radius (length units)

% h = height (same length units)

%

% Outputs

% V = volume of cylinder (cubic length units)

% SA = surface Area of cylinder (squared length units)

% input section - hard coded in Meters

r = 1.3;

h = 2;

% Calculation & display of volume

disp(‘Volume of Cylinder (cubic meters):’)

V = pi\*r.^2.\*h

% Calculation & display of surface area

disp(‘Surface Area of Cylinder (square meters):’)

SA = 2\*pi\*r.\*(r + h)

**Execution:**

>> format compact

>> CylScript

Volume of Cylinder (cubic meters):

V =

10.6186

Surface Area of Cylinder (square meters):

SA =

26.9549