PHYS 361: Syntax, Built-in Variables, Keywords, and Functions

Learning Goals:

- Learn the syntax of basic arithmetic
- Learn about built-in variables, keywords, and functions

Basic Commands

The semicolon (;): Suppresses the output

```
8;
5
```

Percent (%): Indicates the line is a comment. It is used mostly in longer programs. Notice you can comment and uncomment code by highlighting it and using the toolbar.

```
%This is a comment.

% I'm a big block of code.
% Uncomment me and comment me out using the
% comment button on the toolbar.

%{
This is another way to make a big block comment. It's useful when you want to write paragraphs


This is still comment.
%}
```

Session-level commands:

clear: Clear all variables from memory. Note you can also clear variables under the Home tab by selecting "Clear Workspace."

clc: Clears the command window but not the workplace (try it).

close: Close open windows like figures.

who or whos: Currently defined variables. Whos gives additional information.

doc, help, lookfor: Pull up documentation or search for a keyword with lookfor.

ans: Most recent answer in command window

diary: Toggles logging on and off. When logging is on, commands and output from the Command Window is saved in a text file named `diary`.

```
clear    %Clear all variables
clc
```

```
x=5;
y=9;
```

```
who        %Show defined variables
whos
```

```
clear x    %Just clear x
```

```
who        %Now x should be undefined
```

System Level Commands:

cd: Changes current directory.

dir: List all files in current directory.

what: Lists all MATLAB files in current directory.

pwd: Displays current directory.

path: Displays search path.

save: Saves workspace variables.

load: Loads workspace variables from file

type: Displays content of files.

delete: Deletes a file.

```
a=3;
b=4;

%Example of using save and load
save allvariables
load allvariables.mat
```

Basic Arithmetic

Below are some basic arithmetic functions.

Operations:

Operation	Symbol	Example
Addition	+	5+3
Subtraction	-	5-3
Multiplication	*	5*3

Division	/	5/3
Exponentiation	^	5^3

Some predifined variables:

Try running the code below.

```
clear;
5
ans     %The last expression
ans * 5 %You can use ans like a variable
inf     %Infinity
3+2*i       %Complex i
i^2     %Should be negative one
j       %Complex j
NaN     %Not a number
0/0     %Like this
i=2     %note, you can still use i and j for a variable.
```

Declaring Variables

Rules about variable names:

- Must begin with a letter
- Can be up to 63 characters long (BUT SHOULD NEVER BE)
- Can contain letters, numbers, and underscores (total_resistance)
- Cannot contain punctuation (e.g., period, comma, semicolon)
- Cannot contain spaces (use an underscore)
- Cannot be a built-in keyword (if, for) and shouldn't be a built-in function (cos, sin, exp, sqrt)

```
length_pipe=10;

cos(1)
isvarname 1x   %checks if the variable name is legit, 0 is bad and 1 is good
isvarname x1   %different syntax, does the same thing
```

Keywords and Built-in Functions

Pre-defined keywords (off limits for variable names): if, while, break, case, catch, classdef, continue, else, elseif, end....

```
iskeyword   %Displays all MATLAB keywords
```

Some built-in functions: sqrt(x), nthroot(x,n), exp(x), abs(x), log(x), log10(x), sin(x), sind(x), round(x), fix(x), ceil(x), floor(x), sign(x)

A full list of built-in functions: https://www.mathworks.com/help/matlab/referencelist.html?type=function&category=index&s_tid=CRUX_lftnav_function_index

Syntax Highlighting:

- Keywords are blue.
- Built-in functions are purple
- Comments are green.

If you wonder what a function does, try typing it in the command window or looking at the documentation:

```
help sind
doc sind
```

Note that we are using **parentheses** and **lower case names**. See what happens when you make a mistake. Type "Sind(1)" and "sind[1]" in the command window.