EEE 208 – Programming for EEE

Assist. Prof. Dr. Engin Mendi

Variables and Assignments

- To assign = to give a value; to attribute
- `>>x = 2`
- `>>x = 2;`
- What is the difference between these commands?

Examples of Variables and Assignments

Legal

- `>> A = sqrt(13)`
- `>> B = exp(2);`
- `>> A = 2*B`
- `>> A = A + 1`
- `>> sum = A + B;`
- `>> theta = pi/4;`
- `>> c = tan(theta);`

Why are these commands illegal?

- `>> 3 = E`
- `>> 3*A = 14`
- `>> F = 2 3`

Note!

- MATLAB is a case-sensitive programming language
  - You can have two different variables called B and b, so pay special attention when using names for your variables
  - Always use meaningful names for your variables so you never make such mistakes
  - It will also help others who want to read your program
  - We will talk more about this in the section called Documentation

Reminder about Precedence (Priority) Rules

- Be especially careful when entering fractions, complex numbers, etc
- `>> x = 9/2*i`
- `>> y = 9/2i`

- Use parentheses whenever you are not sure
- `>> z = 9/(2*i) + (4i/3)`

- MATLAB shows unbalanced parentheses

Relational Operators

RELATIONAL OPERATOR	MEANING
<	less than
<=	less than or equal
>	greater than
>=	greater than or equal
==	equal
~=	not equal

- Used for comparisons
- If the statement is true, MATLAB returns 1; otherwise, 0.
- `>> 5<8`
- `>> sin(3*pi/2) >= sin(3*pi/4)`

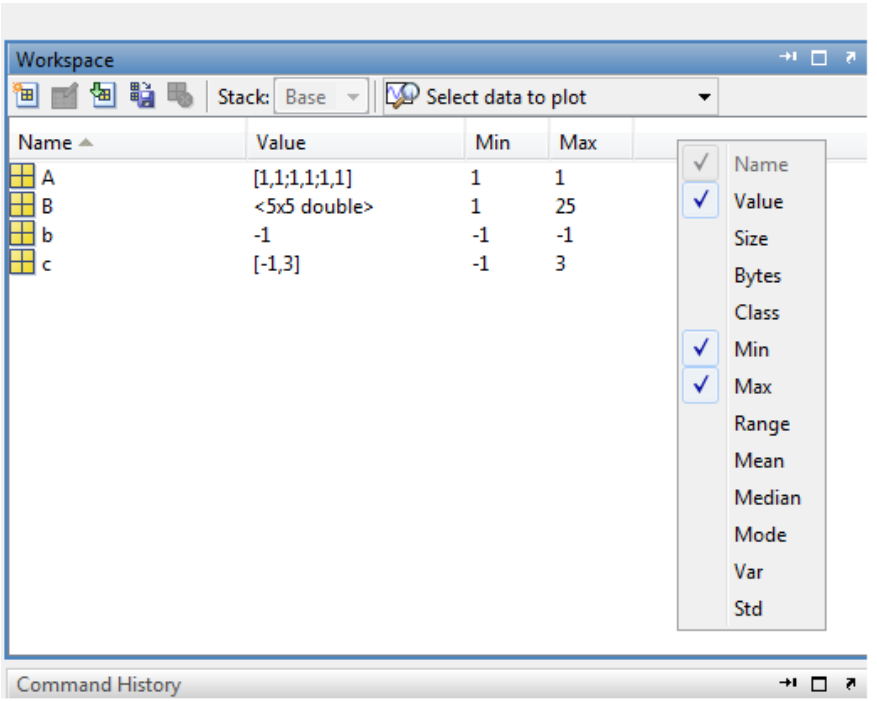
History and Command Windows

- On the right-hand, bottom side of your MATLAB desktop, there is a window called History
- In this window, you can see all commands entered during each session (since opening Matlab)
- You can use the up-arrow key to bring previous commands to the screen and then modify them

- `>>clc` clears the command window

Having access to the Workspace

- On the right-hand, top side of your MATLAB desktop, there is a window called Workspace
- You can see all variables created during the current session
- Right-click on the menu bar to add more properties for each variable



Having access to the Workspace, Cont.

- To see which variables exist in the Workspace, type >>`who` and >>`whos`
- What is their difference?

Saving the Workspace

- Variables are deleted from the memory when you close MATLAB
- Using the command `>>save`, you can save all variables in the workspace to a filed called `matlab.mat` in the current directory

Saving the Workspace, Cont.

- You can use a different name, for example, `>>save Lec1` saves all variables in a file called `Lec1.mat`

- `>> save homework1 A B c M*` saves the variables `A`, `B`, `c` and any variable beginning with `M` in the workspace into a file called `homework1.mat`

Loading a .mat file, Method 1

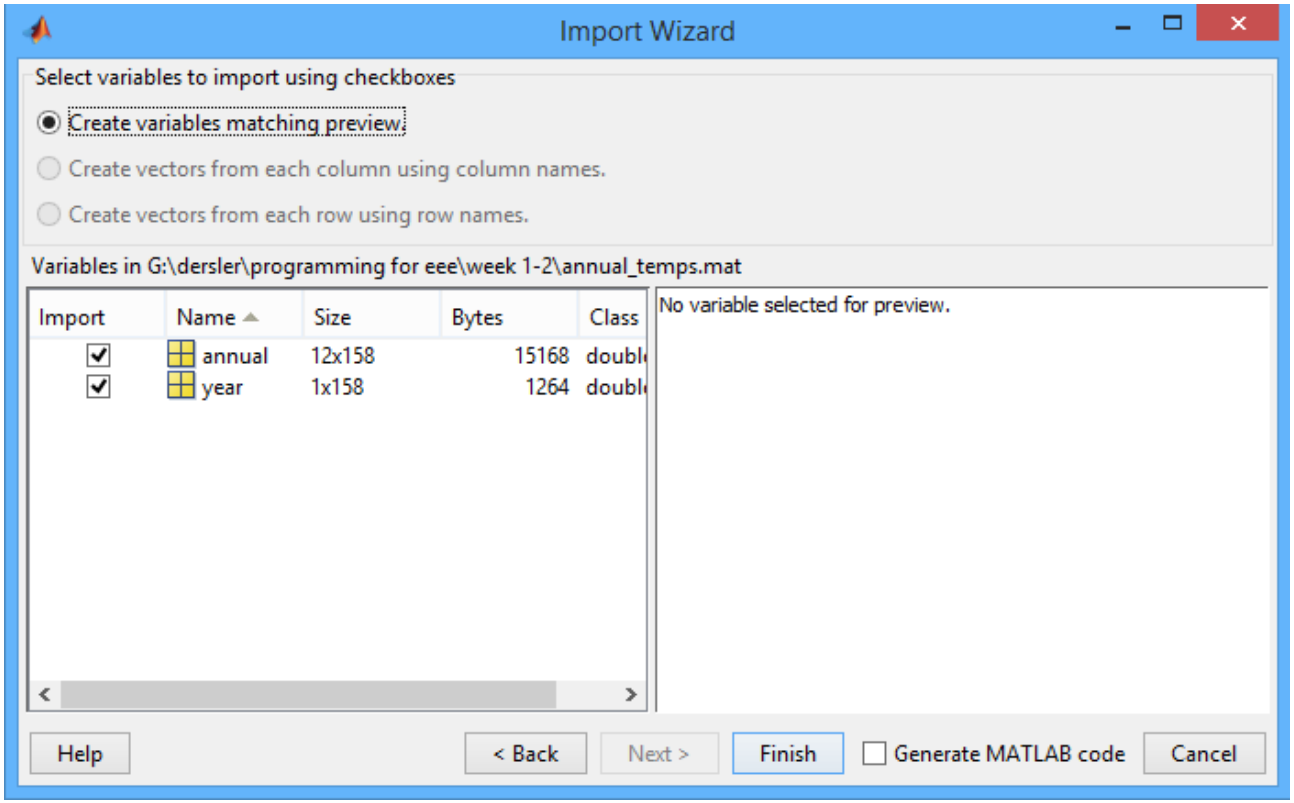
- `load` is the opposite of `save`:
- `>>load` loads all variables from the file `matlab.mat` (if it exists) into the workspace
- `>> load Lec1` loads all of the variables from the file `Lec1.mat`

Loading a .mat file, Method 2

- Another way is to go to File-> Open, and then select the `.mat` file from the right directory
- It is easier and doesn't require you to change the current directory from the toolbar

Loading a .mat file, Method 3

- File -> Import Data…
- Useful for importing large databases with hundreds or thousands of variables, or for importing data from notepad or Excel files



Clearing Variables

- `>> clear` removes all variables from memory
- `>> clear A` removes only variable `A` from memory
- `>> clear A, M*` removes variable `A` and any variables starting with `M*` from memory
- `>> clear all` removes all variables and functions from memory

- Read more `>> help clear`

Example – Temperature Analysis

- Clear the workspace
- Open the file `annual_temps.mat`
- Double click on its varibles. What kind of information do they hold?

Class Exercise 1

Problem: An object has a mass of 50 kg. Find its weight

(a) at the equator (distance from the center of earth = 6378 km), and

(b) 700 meters above the equator.

Gravitational constant G = $6.674 * 10^{-11}$ N . $(m/kg)^2$

Earth mass = $5.97*10^{24}$ kg

Class Exercise 1, Instructions

1. Clear the workspace and command windows
2. Define parameters for mass, G, earth radius, and height
3. Use the right formula to calculate the weights
4. Use two different variables to store the weight in each case

Directory and path

Command	Description
`addpath dirname`	Adds the directory `dirname` to the search path.
`cd dirname`	Changes the current directory to `dirname`.
`dir`	Lists all files in the current directory.
`dir dirname`	Lists all the files in the directory `dirname`.
`path`	Displays the MATLAB search path.
`pathtool`	Starts the Set Path tool.
`pwd`	Displays the current directory.
`rmpath dirname`	Removes the directory `dirname` from the search path.
`what`	Lists the MATLAB-specific files found in the current working directory. Most data files and other non-MATLAB files are not listed. Use `dir` to get a list of all files.
`what dirname`	Lists the MATLAB-specific files in directory `dirname`.

Numerical Display Formats

Command	Description and example
`format short`	Four decimal digits (the default); 13.6745.
`format long`	16 digits; 17.27484029463547.
`format short e`	Five digits (four decimals) plus exponent; 6.3792e+03.
`format long e`	16 digits (15 decimals) plus exponent; 6.379243784781294e−04.

COMMAND	MEANING
format short	5 significant decimal digits
format long	15 significant digits
format short e	scientific notation with 5 significant digits
format long e	scientific notation with 15 significant digits
format hex	hexadecimal
format +	+ printed if value is positive, - if negative; space is skipped if value is zero

Arrays (Vectors) and Comparisons

- Arrays or Vectors are one dimensional matrices
- We use brackets [ ] to define arrays
- `v1 = [1 2 3]` and `v2 = [1, 2, 3]` are two equal row vectors
- You can check it by writing `>> v1==v2` (meaning: compare v1 and v2 element by element; return 1 if equal, 0 if not equal)
- `>> v3=[1 2 4]`
- `>> v1==v3`

Column Vectors

- Use semicolons to separate rows
- `>> v4 = [1; 2; 3]`
- Find transpose of arrays
- `>> v5 = v2'`
- `>> v6 = v5'`
- Are they equal?
- `>> v2 == v6`
- `>> v2 == v4`

Appending Row Vectors

- Suppose you have two row vectors

`w1=[1,3,5]` and `w2=[2,4,6]`

- You can append (stick) these row vectors to each other and create a new row vector:
- `>> w_new = [w1, w2]`
- Result:

```
w_new = [1,3,5,2,4,6]
```

Creating Row Vectors with Colon Operator

- A row vector showing integers from -4 to 2:

`>>x=-4:2;` is the same as `x=[-4:2]`

- A row vector showing equal points in time

`>>t1= 0:0.5:2.5;` is the same as

`t1=[0:0.5:2.5]`

Creating Row Vectors with Colon Operator, cont.

- In general,

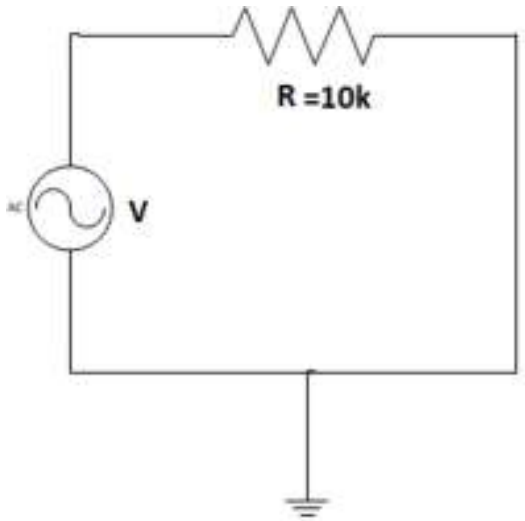
`t= init: step: final`

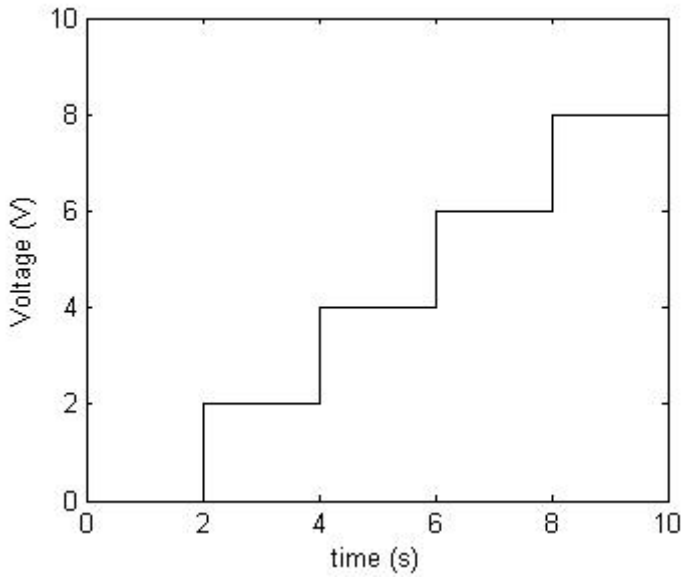
creates a row vector that starts with `init`, has spacing of `step` and ends before or at `final`.

- `t2= 1:0.3:2` creates `t2=[1, 1.3, 1.6, 1.9]` (2.2>2 so it's not included)
- `t= init: 1: final` is the same as `t=init: final`

Class Exercise 2

Problem: The voltage in a simple resistor circuit increases from 0 to 8V in steps of 2V. Find the current passing through the 10 kΩ resistor.





Class Exercise 2, Instructions

1. Clear the workspace and screen.
2. Define parameters for voltage V and resistance R. What is the size of V?
3. Calculate the current I using V and R. What is the size of I?