

Day 5

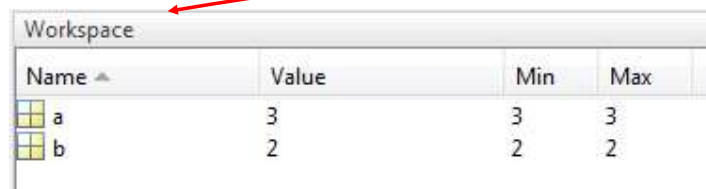
- (Concept Question)
- Clearing your variables
- Basic recursive assignment
- Initialization
- Fancy recursive assignment
- (Exercises)

ME123 Computer Programming

Clearing your variables

- Matlab remembers the values of all of the variables you defined in a given session
- You can see the variables in the Workspace in the lower left corner of your Matlab windows

```
Command Window
>> a=3;
>> b=2;
fx >>
```



Name	Value	Min	Max
a	3	3	3
b	2	2	2

ME123 Computer Programming

Clearing your variables

'clear variables' cleans out the workspace

```
Command Window
>> clear variables
fx >> |
```

Workspace			
Name ^	Value	Min	Max

Clearing your variables

We want our scripts to “start fresh” each time we run them, so from now on use ‘clear variables’ at the start of all of your scripts

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
```

Note: ***DO NOT USE just 'clear'.*** That could cause trouble later.

Basic recursive assignment

“Recursive Assignment” assigns a variable a new value that depends on its previous value

Remember: This is an “assignment”—not an equation

1) MATLAB starts here—it takes the current value of the variable “counter”

2) It then adds a “1” to that value

3) It then puts that result back into the variable “counter”

```
counter = counter + 1;
```

ME123 Computer Programming

Basic recursive assignment

Recursive assignment can be used to count the number of times we have gone through a loop

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
3 -   %
4 -   counter=0;
5 -   fprintf('Counter   Loopvariable \n');
6 -   %
7 -   for loopvariable=1:5:25
8 -       counter=counter+1;
9 -       fprintf('%5.0f   %5.0f \n',counter,loopvariable);
10 -
11 -   end
```

The variable “counter” starts with a value of 0

“counter” gets bigger by one each time we go through the loop

ME123 Computer Programming

Basic recursive assignment

The code makes a table

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
3 -   %
4 -   counter=0;
5 -   fprintf('Counter   Loopvariable \n');
6 -   %
7 -   for loopvariable=1:5:25
8 -       counter=counter+1;
9 -       fprintf('%5.0f   %5.0f \n',counter,loopvariable);
10 -
11 -   end
```

Command Window

Counter	Loopvariable
1	1
2	6
3	11
4	16
5	21

ME123 Computer Programming

Basic recursive assignment

Recursive assignments can be used with any valid variable name. Same variable on both sides!

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
3 -   %
4 -   Fred=0;
5 -   fprintf('Counter   Loopvariable \n');
6 -   %
7 -   for loopvariable=1:5:25
8 -       Fred=Fred+1;
9 -       fprintf('%5.0f   %5.0f \n',Fred,loopvariable);
10 -
11 -   end
```

ME123 Computer Programming

Initialization

Recursive assignments must be initialized.

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
3 -   %
4 -   counter=0;
5 -   fprintf('Counter   Loopvariable \n');
6 -   %
7 -   for loopvariable=1:5:25
8 -       counter=counter+1;
9 -       fprintf('%5.0f       %5.0f \n', counter, loopvariable);
10 -
11 -   end
```

Initialization
(Starting Value)

ME123 Computer Programming

Initialization

If you forget to initialize the variable in a recursive assignment you will get an error

```
Day5_class_example1.m x
1 -   clc
2 -   clear variables
3 -   %
4 -   fprintf('Counter   Loopvariable \n');
5 -   %
6 -   for loopvariable=1:5:25
7 -       counter=counter+1;
8 -       fprintf('%5.0f       %5.0f \n', counter, loopvariable);
9 -
10 -   end
```

Missing initialization

Matlab can't add 1 to the existing value of counter because it's not defined yet

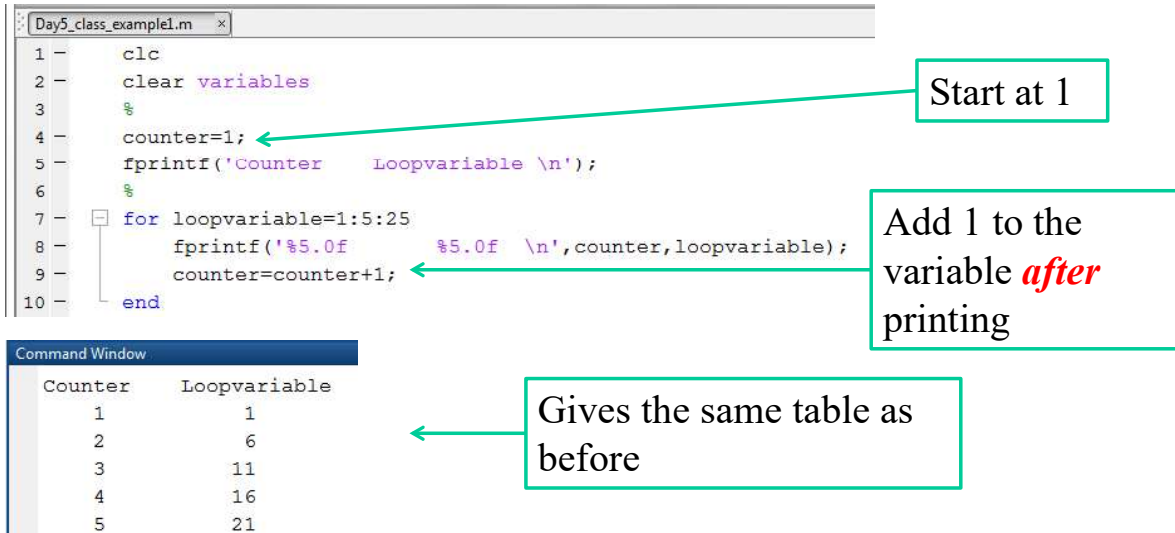
```
Command Window
Counter   Loopvariable
Undefined function or variable 'counter'.

Error in Day5_class_example1 (line 7)
    counter=counter+1;
```

ME123 Computer Programming

Initialization

We don't always want to initialize the variable to zero.



The image shows a MATLAB script editor window titled 'Day5_class_example1.m' with the following code:

```
1 -   clc
2 -   clear variables
3 -   %
4 -   counter=1;
5 -   fprintf('Counter   Loopvariable \n');
6 -   %
7 -   for loopvariable=1:5:25
8 -       fprintf('%5.0f   %5.0f \n', counter, loopvariable);
9 -       counter=counter+1;
10 -  end
```

Annotations with arrows point to specific lines:

- 'Start at 1' points to line 4: `counter=1;`
- 'Add 1 to the variable *after* printing' points to line 9: `counter=counter+1;`
- 'Gives the same table as before' points to the Command Window output table.

The Command Window shows the following output table:

Counter	Loopvariable
1	1
2	6
3	11
4	16
5	21

ME123 Computer Programming

Fancy recursive assignment

Recursive assignments can be more sophisticated than just `counter=counter+1`:

`>> var=var+2;` Add 2 to the existing value of 'var'

`>> var=var*2;` Double the existing value of 'var'

`>> var=var^3;` Cube the existing value of 'var'

All are recursive: 'var' on both sides of =

ME123 Computer Programming

Fancy recursive assignment

Note: for recursive assignments like doubling or cubing it doesn't make sense to initialize to zero

```
Command Window
>> var=0;
>> var=var*2

var =

    0
```

```
Command Window
>> var=0;
>> var=var^3

var =

    0
```

Fancy recursive assignment

Recursive assignments may include other variables too:

```
>> var=var+2*loopvariable
```

```
>> var=var+loopvariable^2
```

```
>> var=var+b^loopvariable
```

```
>> var=var*loopvariable
```

All are recursive: 'var' on both sides of =