



## Commands in MATLAB:

```
>> x = 5
```

```
x =
```

```
5
```

Including a semicolon suppresses output:

```
>> x = 5;
```

# Arrays of numbers:

```
>> v = 0:0.5:2
```

```
v =
```

```
    0    0.5000    1.0000    1.5000    2.0000
```

Arrays of numbers:

```
>> v = 0:0.5:2
```

```
v =
```

```
    0    0.5000    1.0000    1.5000    2.0000
```

First number is starting point, second is step size, third is end point. If the second is omitted, the step size is assumed to be 1.

```
>> v = 0:3
```

```
v =
```

```
    0    1    2    3
```

# Accessing the elements of an array:

```
>> v = 0:0.5:2
```

```
v =
```

```
    0    0.5000    1.0000    1.5000    2.0000
```

# Accessing the elements of an array:

```
>> v = 0:0.5:2
```

```
v =
```

```
    0    0.5000    1.0000    1.5000    2.0000
```

```
>> v(1)
```

```
ans =
```

```
    0
```

# Accessing the elements of an array:

```
>> v = 0:0.5:2
```

```
v =
```

```
    0    0.5000    1.0000    1.5000    2.0000
```

```
>> v(1)
```

```
ans =
```

```
    0
```

```
>> v(5) = 100
```

```
v =
```

```
    0    0.5000    1.0000    1.5000   100.00
```

MATLAB m-files: Two types.

1) MATLAB *scripts*. When called, these execute the commands in the script one at a time.

MATLAB m-files: Two types.

1) MATLAB *scripts*. When called, these execute the commands in the script one at a time.

2) MATLAB *functions*. These define a function such as  $\mathbf{f}(\mathbf{t}, \mathbf{x})$  that can be called from the command line or other m-files.

Example MATLAB function:

```
%function that returns the product of  
%its inputs.  save as myf.m  
function [output] = myf(input1, input2)  
    output = input1 * input2;  
end
```

Example MATLAB function:

```
%function that returns the product of  
%its inputs save as myf.m  
function [output] = myf(input1, input2)  
    output = input1 * input2;  
end
```

```
>> a = myf(3,5)
```

```
a =
```

```
15
```