```
disp('Hello world')
```

```
Hello world
```

```
%print('Hello world')

help print
```

```
  print Print or save a figure or model.
    A subset of the available options is presented below. For more details
    see the documentation.
 
    print, by itself, prints the current figure to your default printer.
    Use the -s option to print the current model instead of the current figure.
      print         % print the current figure to the default printer
      print -s      % print the current model to the default printer
 
    print(filename, formattype) saves the current figure to a file in the
    specified format. Vector graphics, such as PDF ('-dpdf'), and encapsulated
    PostScript ('-depsc'), as well as images such as JPEG ('-djpeg') and PNG ('-dpng')
    can be created. Use '-d' to specify the formattype option
      print(fig, '-dpdf', 'myfigure.pdf'); % save to the 'myfigure.pdf' file
    The full list of formats is documented here.
 
    print(printer, ...) prints the figure or model to the specified printer.
    Use '-P' to specify the printer option.
      print(fig, '-Pmyprinter'); % print to the printer named 'myprinter'
 
    print(resize,...) resizes the figure to fit the page when printing.
    The resize options are valid only for figures, and only for page
    formats (PDF, and PS) and printers. Specify resize as either
      '-bestfit'  to preserve the figure's aspect ratio or
      '-fillpage' to ignore the aspect ratio.
 
  The documentation contains additonal details and examples, including how to
  specify the figure or model to print, adjust the output size and
  resolution, save to the clipboard, and specify the renderer to use.
 
  See also saveas, printpreview, savefig.

  Reference page for print
```

```
help disp
```

```
 disp Display array.
    disp(X) displays array X without printing the array name or
    additional description information such as the size and class name.
    In all other ways it's the same as leaving the semicolon off an
    expression except that nothing is shown for empty arrays.
 
    If X is a string or character array, the text is displayed.
 
    See also fprintf, sprintf, int2str, num2str, rats, format, details.

    Reference page for disp
    Other functions named disp
```

welcome to Matlab!

```
a = 2
```

```
a = 2
```

```
a * 2
```

```
     4
```

Programming :

- Saving variables into memory
- operating on the saved variables

```
b = a * 2  %% the * operation is a built in matlab operation
```

```
b = 4
```

```
disp(b)
```

```
     4
```

```
disp(1i)  %% some variables are already saved in matlab
```

```
   0.0000 + 1.0000i
```

You have to know the landscape

```
a = 3
```

```
a = 3
```

```
disp(b)
```

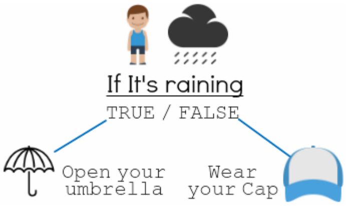
Line by line execution!

Branching

Getting different parts of your code based on different conditions

```
clear;

imshow('assets/if_logic.png')
```



How to "branch" your code

```
myinput = input('Put in a number please')
```

```
myinput = 19
```

```

if myinput > 20
    disp('high!')
end
```

```
Low
```

```

myinput = input('Put in a number please')

if myinput > 20
    disp('high!')
else
    disp('Low')
end
```

Iterations

Repeat the same segment of code multiple times

```
clear

disp(' Not a loop')
disp(1)
disp(2)
disp(3)
disp(4)
disp(5)
disp(6)
disp(7)
disp(8)
disp(9)
disp(10)

Mylist = [1,2,3,4,5,6,7,8,9,10];

disp('My first Loop')
```

```
My first Loop
```

```
for i = Mylist
    disp(i)
end
```

```
     1

     2

     3

     4

     5

     6

     7

     8

     9

    10
```

```


disp('My second Loop')
```

```
My second Loop
```

```


for i = 1:10
    disp(i)
end
```

```
     1

     2

     3

     4

     5

     6

     7

     8
     9

    10
```

```
disp('My third Loop')
```

```
My third Loop
```

```
for i = 1:2:length(Mylist)
    disp(i-1)
end
```

```
     0

     2

     4

     6

     8
```