

10-27_random_walk_and_python_introduction

September 27, 2023

```
[1]: print("hello!")
```

hello!

```
[2]: x = "MME 502"
```

```
[3]: print("hello")  
print(x)
```

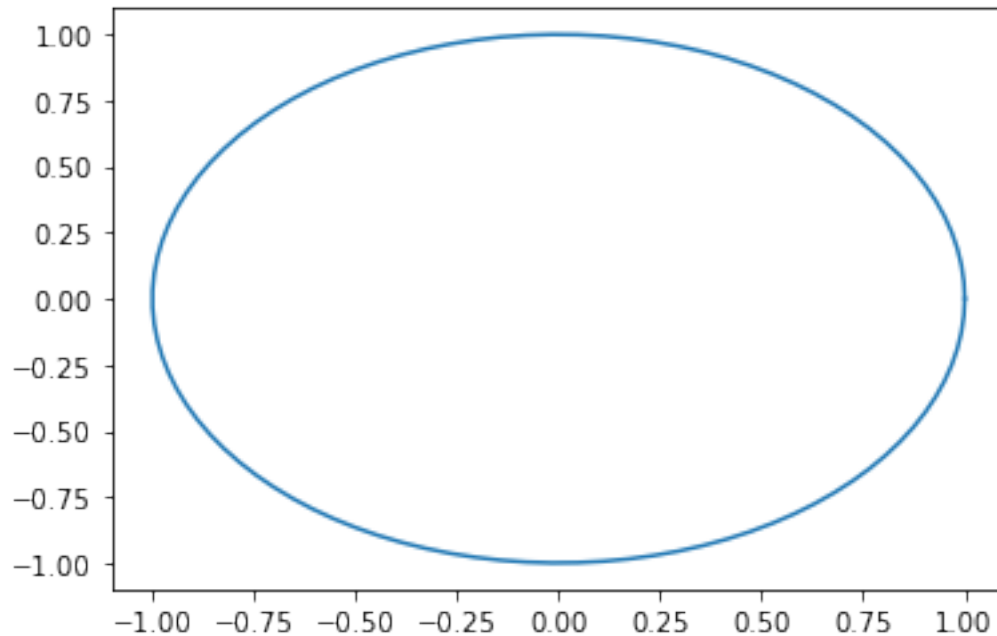
hello
MME 502

```
[4]: import numpy as np  
import matplotlib.pyplot as plt
```

```
[5]: %matplotlib inline
```

```
[6]: t = np.linspace(0, 2 * np.pi, 100)  
x = np.cos(t)  
y = np.sin(t)  
plt.plot(x,y)
```

```
[6]: [<matplotlib.lines.Line2D at 0x7f537129ad60>]
```



```
[7]: # This is comment
```

```
[8]: def random_walk(steps=100):  
    X = np.zeros([2])  
    X_history = np.zeros([2, steps + 1])  
    X_history[:, 0] = X  
    for i in range(steps):  
        X = X + np.random.randn(2)  
        X_history[:, i + 1] = X  
    return X_history
```

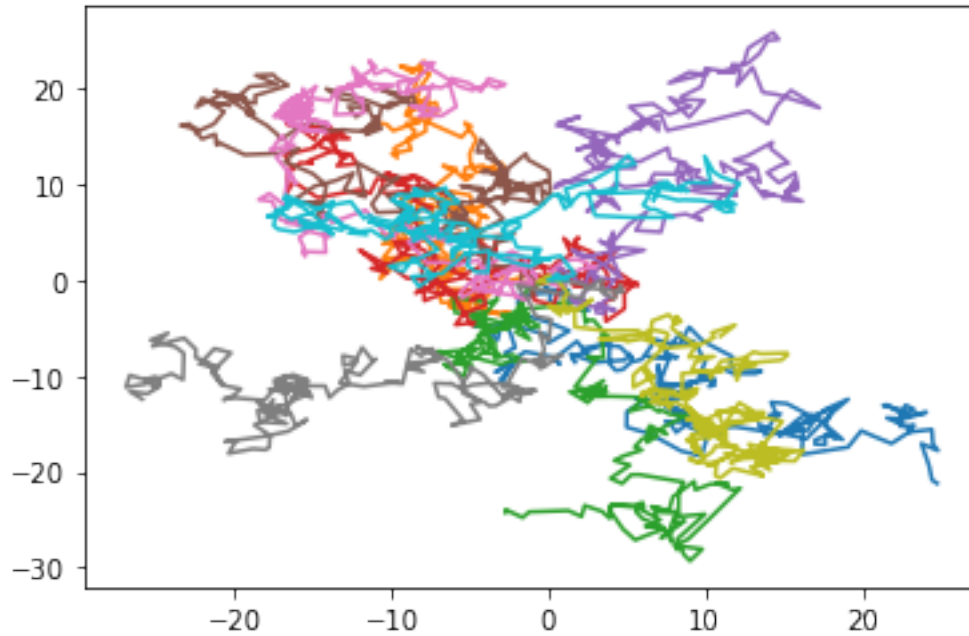
```
[9]: x = np.array([1, 3, 4])  
    y = np.array([6, 3, 7])  
    x + y
```

```
[9]: array([ 7,  6, 11])
```

```
[10]: random_walk().shape
```

```
[10]: (2, 101)
```

```
[11]: for i in range(10):  
    X = random_walk(200)  
    plt.plot(X[0,:], X[1,:])
```



```
[12]: x = 100
      if x < 25:
          print("yes")
      elif x > 75:
          print("no")
      else:
          print("maybe")
```

no

```
[13]: t = 10
      while t > 0:
          print(t)
          t = t - 1
```

10
9
8
7
6
5
4
3
2
1

```
[14]: x = 100.5 # Floating Point Number
      y = 67 # Integer
      z = 54e3 # Evaluates to 54000
      s = "test" # Strings
      s2 = 'test' # Identical to s
```

```
[15]: s3 = "x'(t)+3x'(t)+7x(t)=0"
```

```
[16]: x = random_walk()
```

```
[17]: x[1,::-1]
```

```
[17]: array([-0.12330386, -1.86171507, -1.81514954, -2.05598986, -3.52035963,
          -2.80625334, -1.74387532, -1.57144054, -0.64258159, -0.83656016,
          -0.88258335,  0.31406715, -0.7586129 , -0.2581874 , -0.67861918,
           0.35745849,  0.08615465,  0.6358724 ,  1.43413333,  1.01809002,
           2.95976322,  2.42410696,  1.28358668, -0.18069309,  0.40981067,
           0.59831787,  0.13269334,  0.0848402 ,  0.31860123,  2.01995436,
           1.94033029,  1.19144623,  1.06361374,  1.58748061,  1.60116128,
           2.60248287,  3.14165838,  2.83287629,  3.34375839,  3.11164101,
           2.18909287,  2.24289988,  4.35115102,  4.64286581,  2.79563425,
           5.18410356,  5.53625962,  6.45114351,  7.46588832,  7.41083994,
           7.83864287,  8.54449779,  5.84432169,  5.53737835,  5.43101568,
           5.75298949,  6.40496385,  5.41600717,  3.99289676,  1.98411914,
           1.89043064,  1.64873635,  0.40693777,  2.19276929,  1.90137346,
           0.32090655,  0.39977373,  1.08319431,  2.60905098,  3.33485739,
           4.0950122 ,  3.36668476,  1.91567297,  2.24600133,  2.56979765,
           0.73709235,  1.8988189 ,  3.92150531,  3.61659765,  4.32194251,
           4.00031101,  4.68045706,  3.68433983,  2.91828359,  1.63364958,
           1.96349276,  1.1688064 ,  0.58023465,  1.41255752,  2.66698167,
           4.12419171,  4.41015174,  3.27718809,  1.93300765,  2.15928252,
           1.8707186 ,  3.09514963,  2.45115583,  1.332387 ,  0.32965356,
           0.          ])
```

```
[ ]:
```