01_Matplotlib

December 6, 2019

1 INTRO TO MATPLOTLIB

[0]:
```
%matplotlib inline
```

That line is only for jupyter notebooks, if you are using another editor, you'll use: **plt.show()** at the end of all your plotting commands to have the figure pop up in another window.

[0]:
```
print('Hello world')
```

```
Hello world
```

[0]:
```
a = []
a.append("Hello")
```

[0]:
```
a
```

[0]:
```
['Hello']
```

[0]:
```
a.append("FAV's")
a.append("smart")
a.append("students")
print("The length of list is: ", len(a))
```

```
The length of list is:  4
```

[0]:
```
type(a)
```

[0]:
```
list
```

[0]:
```
print(a)
```

```
['Hello', "FAV's", 'smart', 'students']
```

[0]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id
=947318989803-6bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.googleusercontent.com&redire
```

```
ct_uri=urn%3aietf%3awg%3aoauth%3a2.0%3aoob&response_type=code&scope=email%20http
s%3a%2f%2fwww.googleapis.com%2fauth%2fdocs.test%20https%3a%2f%2fwww.googleapis.c
om%2fauth%2fdrive%20https%3a%2f%2fwww.googleapis.com%2fauth%2fdrive.photos.reado
nly%20https%3a%2f%2fwww.googleapis.com%2fauth%2fpeopleapi.readonly

Enter your authorization code:
ůůůůůůůůůů
Mounted at /content/drive
```

[0]:
```
!pwd
```

```
/content
```

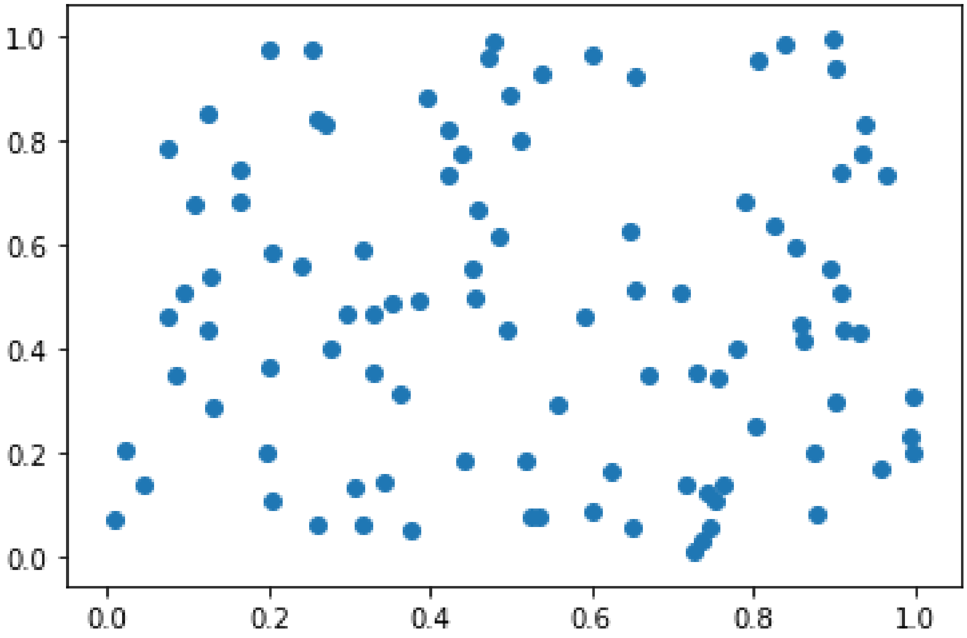
[0]:
```

```

1.0.1 ESEMPI

[0]:
```
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
from numpy.random import normal,rand
```

[0]:
```
a = rand(100)
b = rand(100)
plt.scatter(a,b)
#plt.show()
```

[0]:
```
<matplotlib.collections.PathCollection at 0x7f7013d648d0>
```

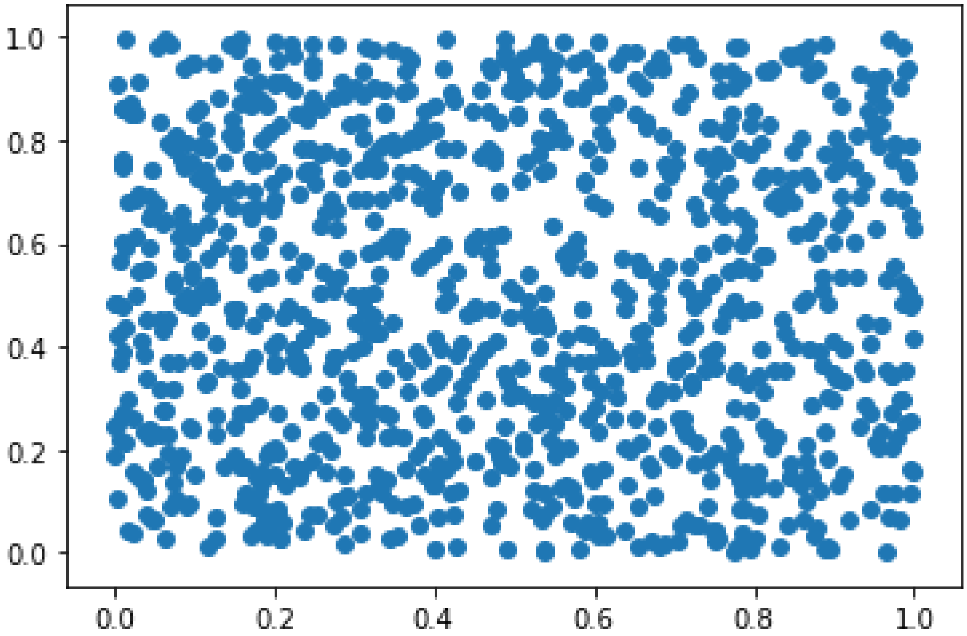


[0]:

```
a = rand(1000)
b = rand(1000)
plt.scatter(a,b)
#plt.show()
```

[0]:

```
<matplotlib.collections.PathCollection at 0x7f701103e1d0>
```



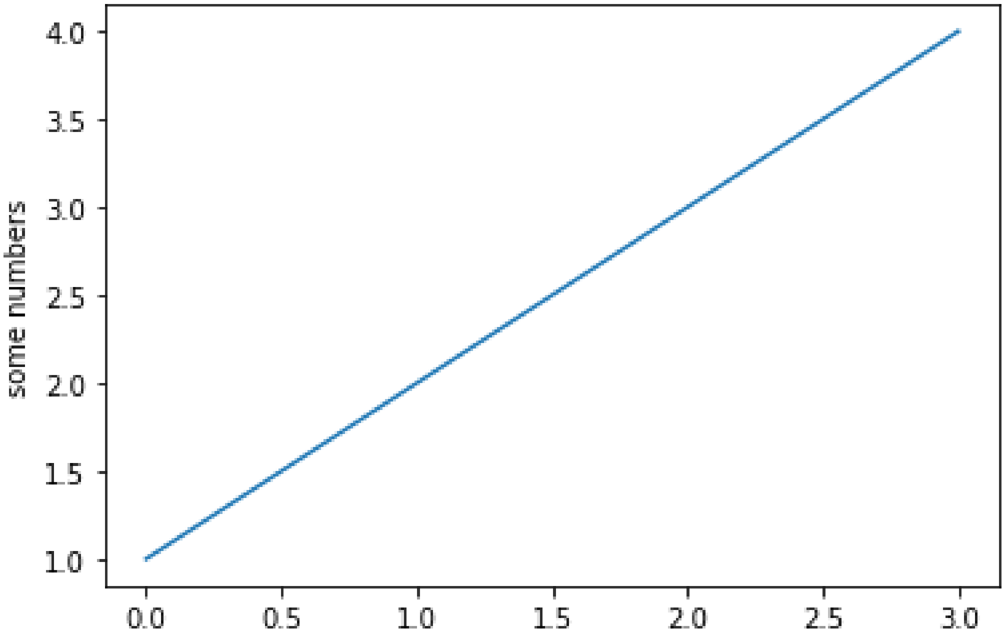
[0]:

```
print(type(a))
```

```
<class 'numpy.ndarray'>
```

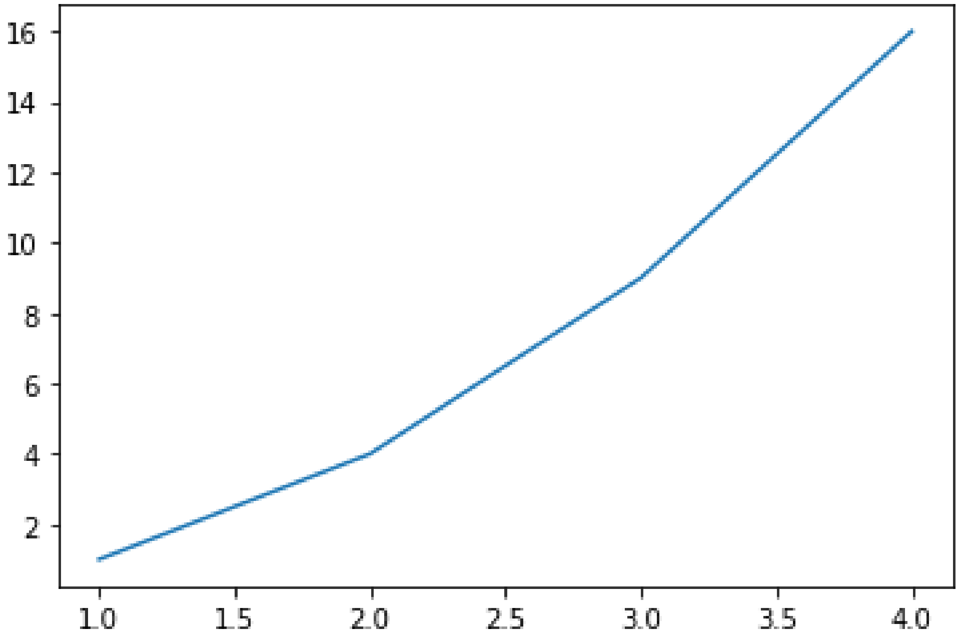
[0]:

```
plt.plot([1, 2, 3, 4])
plt.ylabel('some numbers')
plt.show()
```



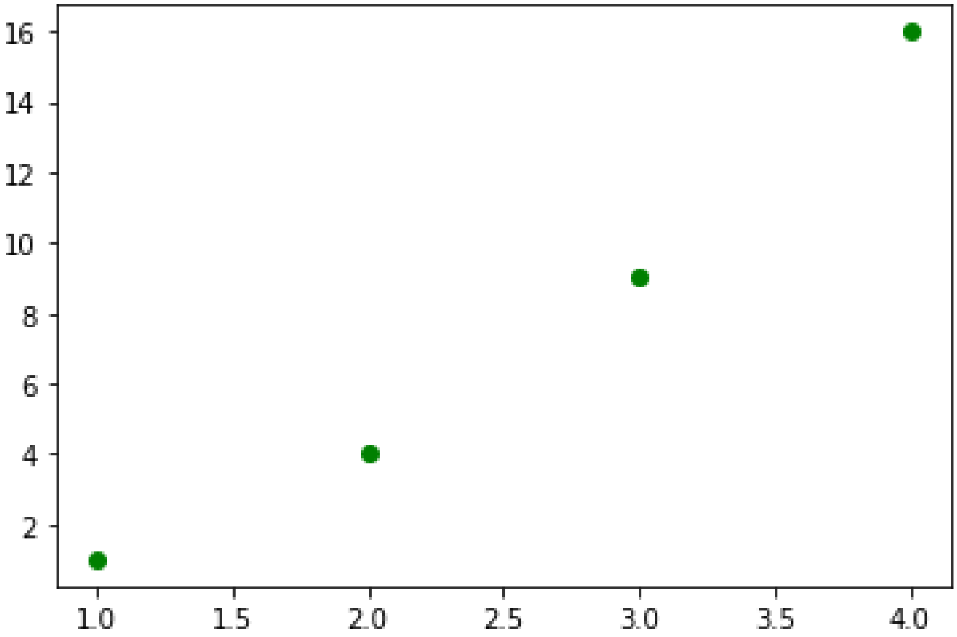
[0]:
```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
```

[0]:
```
[<matplotlib.lines.Line2D at 0x7f7010f8a630>]
```



[0]:

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16], 'go')
#plt.axis([0, 6, 0, 20])
plt.show()
```

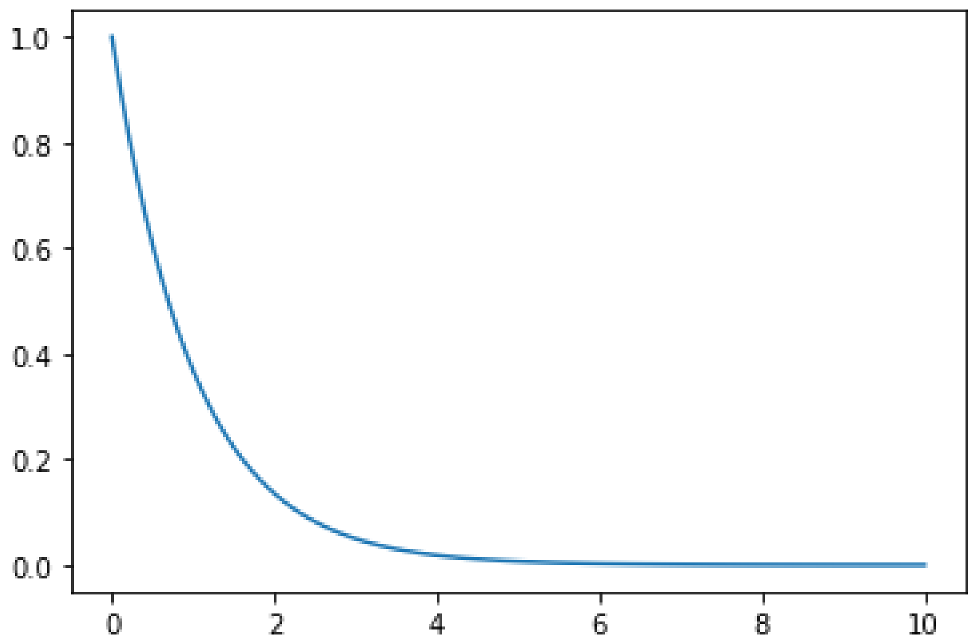


[0]:

```
a = np.linspace(0,10,100)
b = np.exp(-a)
plt.plot(a,b)
#plt.show()
```

[0]:

```
[<matplotlib.lines.Line2D at 0x7f7010e59b38>]
```



scatterplot

[0]:
```
# esercizio
```

[0]:
```
x = [i for i in range (101)]
```

[0]:
```
print(x)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21,
22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41,
42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61,
62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81,
82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
```

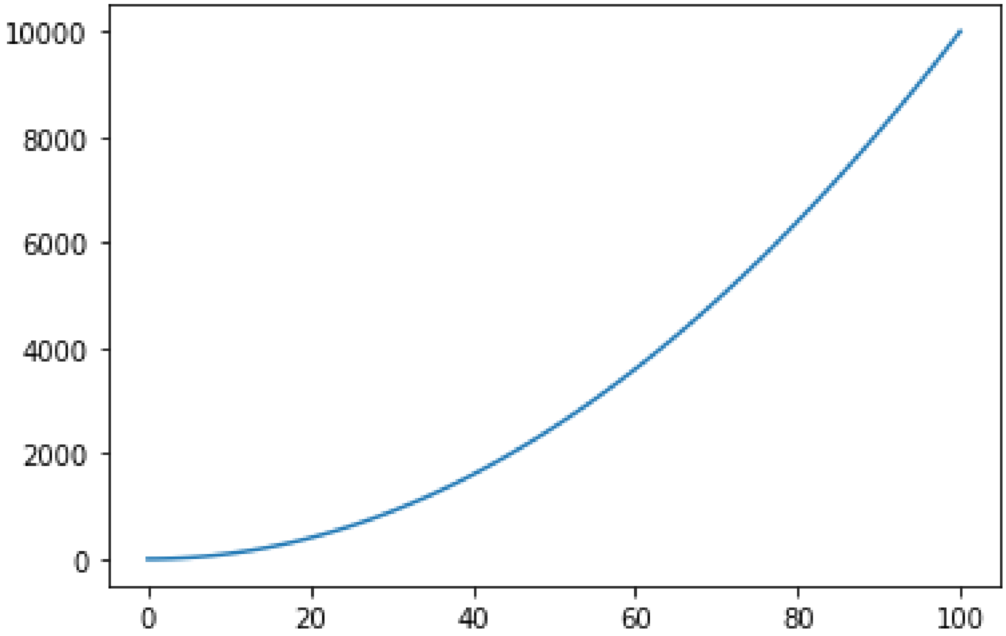
[0]:
```
y = [i**2 for i in x]
```

[0]:
```
print(y)
```

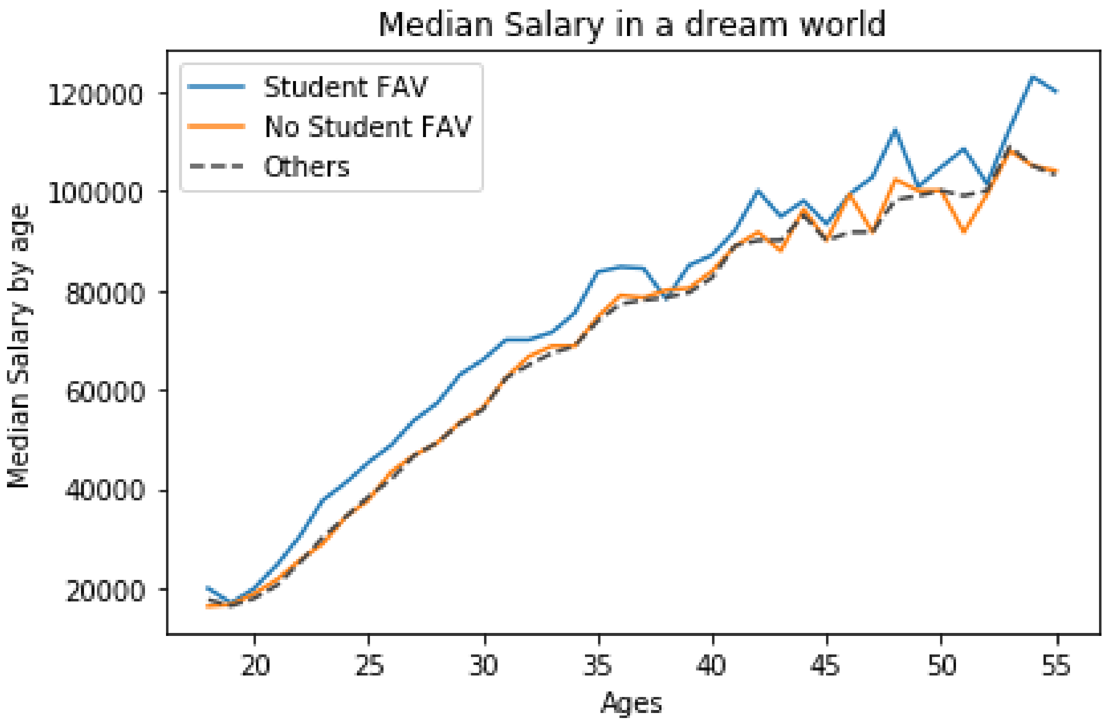
```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289,
324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841, 900, 961, 1024,
1089, 1156, 1225, 1296, 1369, 1444, 1521, 1600, 1681, 1764, 1849, 1936, 2025,
2116, 2209, 2304, 2401, 2500, 2601, 2704, 2809, 2916, 3025, 3136, 3249, 3364,
3481, 3600, 3721, 3844, 3969, 4096, 4225, 4356, 4489, 4624, 4761, 4900, 5041,
5184, 5329, 5476, 5625, 5776, 5929, 6084, 6241, 6400, 6561, 6724, 6889, 7056,
7225, 7396, 7569, 7744, 7921, 8100, 8281, 8464, 8649, 8836, 9025, 9216, 9409,
9604, 9801, 10000]
```

[0]:
```
#plt.figure(figsize=(15,5))
plt.plot(x,y)
```

[0]:
```
[<matplotlib.lines.Line2D at 0x7f7010e3bac8>]
```



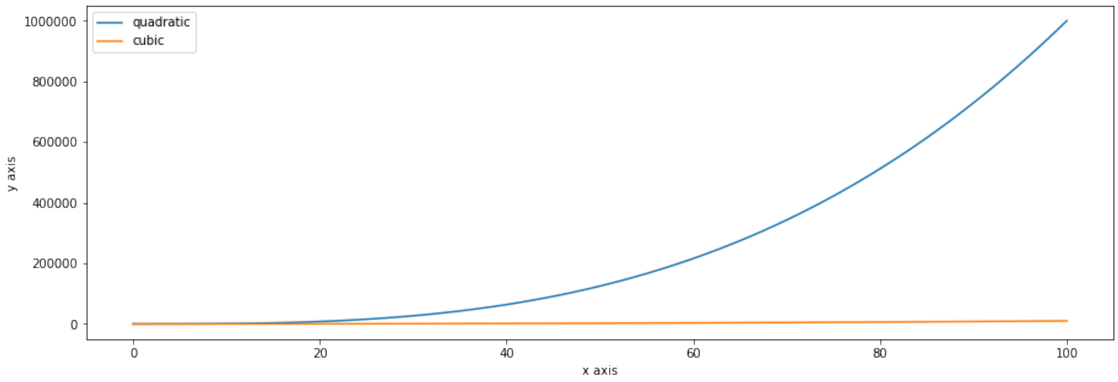
[0]:
```
%matplotlib inline
import  matplotlib.pyplot as plt
from matplotlib import pyplot as plt
import numpy as np



ages_x = [18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,␣
 ↪35,
          36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52,␣
 ↪53, 54, 55]

fav_stud_y = [20046, 17100, 20000, 24744, 30500, 37732, 41247, 45372, 48876,␣
 ↪53850, 57287, 63016, 65998, 70003, 70000, 71496, 75370, 83640, 84666,
            84392, 78254, 85000, 87038, 91991, 100000, 94796, 97962, 93302,␣
 ↪99240, 102736, 112285, 100771, 104708, 108423, 101407, 112542, 122870,␣
 ↪120000]
plt.plot(ages_x, fav_stud_y, label='Student FAV')

nofav_stu_y = [16446, 16791, 18942, 21780, 25704, 29000, 34372, 37810, 43515,␣
 ↪46823, 49293, 53437, 56373, 62375, 66674, 68745, 68746, 74583, 79000,
            78508, 79996, 80403, 83820, 88833, 91660, 87892, 96243, 90000,␣
 ↪99313, 91660, 102264, 100000, 100000, 91660, 99240, 108000, 105000, 104000]
plt.plot(ages_x, nofav_stu_y, label='No Student FAV')

others_y = [17784, 16500, 18012, 20628, 25206, 30252, 34368, 38496, 42000,␣
 ↪46752, 49320, 53200, 56000, 62316, 64928, 67317, 68748, 73752, 77232,
         78000, 78508, 79536, 82488, 88935, 90000, 90056, 95000, 90000, 91633,␣
 ↪91660, 98150, 98964, 100000, 98988, 100000, 108923, 105000, 103117]
plt.plot(ages_x, others_y, color='#444444', linestyle='--', label='Others')


plt.title('Median Salary in a dream world')
plt.xlabel('Ages')
plt.ylabel('Median Salary by age')
plt.legend()
plt.tight_layout()
#plt.savefig('/img/the_salary_I_wonder.png')
#plt.show()

#len(py_dev_y)
#type(ages_x)
```



[0]:
```
y1 = [i**3 for i in x]

y2 = [i**2 for i in x]
```

[0]:
```
plt.figure(figsize=(15,5))
plt.plot(x,y1,y2)
# or plt.plt(x,y1)
#plt.plot(x,y2)
#plt.title( I love fav)
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend(['quadratic','cubic'], loc=0) #1oc = position
```

[0]:
```
<matplotlib.legend.Legend at 0x7f7010e044a8>
```

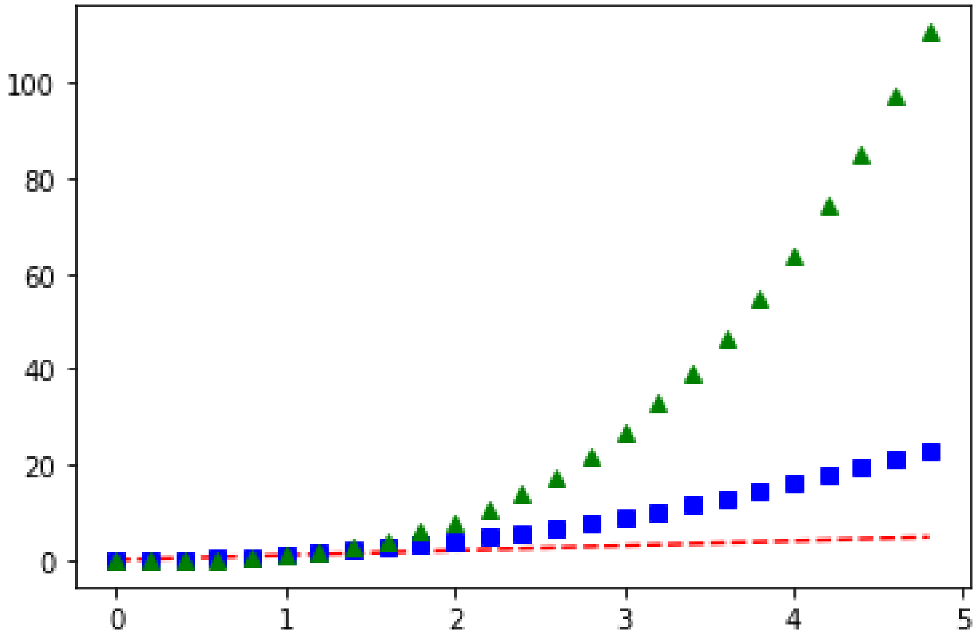


[0]:
```
print(y2)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289,
324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841, 900, 961, 1024,
1089, 1156, 1225, 1296, 1369, 1444, 1521, 1600, 1681, 1764, 1849, 1936, 2025,
2116, 2209, 2304, 2401, 2500, 2601, 2704, 2809, 2916, 3025, 3136, 3249, 3364,
3481, 3600, 3721, 3844, 3969, 4096, 4225, 4356, 4489, 4624, 4761, 4900, 5041,
5184, 5329, 5476, 5625, 5776, 5929, 6084, 6241, 6400, 6561, 6724, 6889, 7056,
7225, 7396, 7569, 7744, 7921, 8100, 8281, 8464, 8649, 8836, 9025, 9216, 9409,
9604, 9801, 10000]
```

[0]:

[0]:
```
# evenly sampled time at 200ms intervals
t = np.arange(0., 5., 0.2)

# red dashes, blue squares and green triangles
plt.plot(t, t, 'r--', t, t**2, 'bs', t, t**3, 'g^')
plt.show()
```

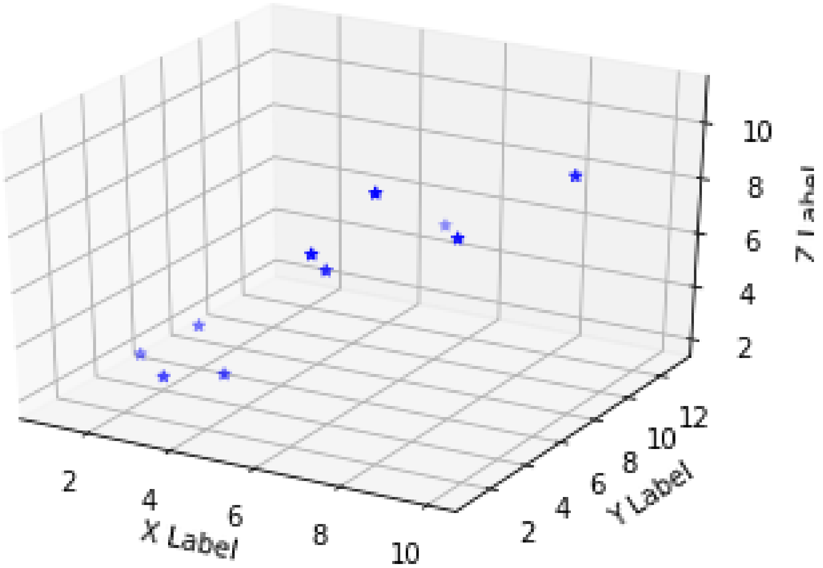


[0]:

```

```

[0]:

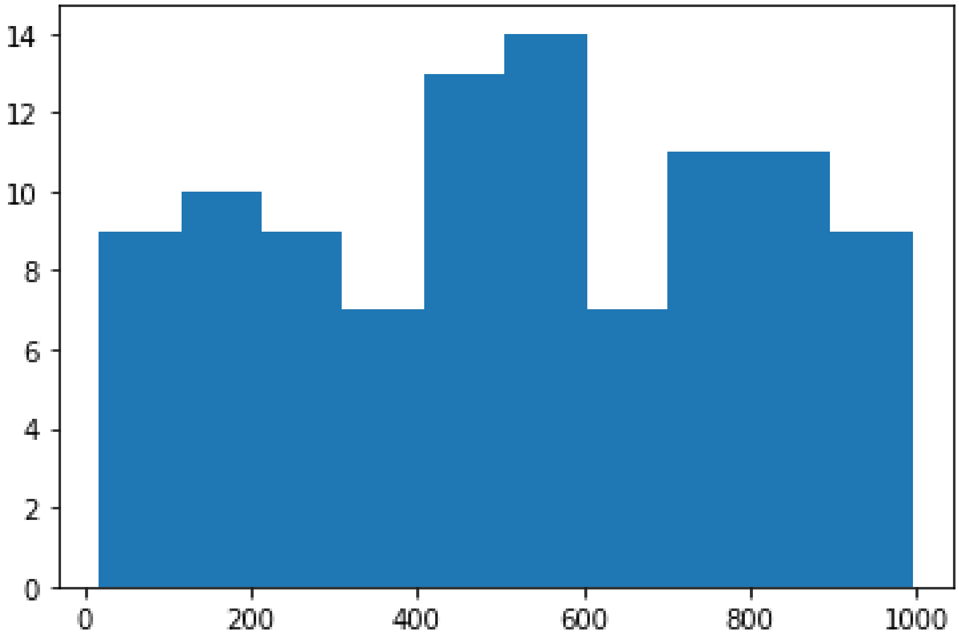
```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_subplot(111,projection='3d' )

x =[1,2,3,4,5,6,7,8,9,10]
y =[5,6,2,3,13,4,1,2,4,8]
z =[2,3,3,3,5,7,9,11,9,10]

ax.scatter(x, y, z, c='b', marker='*' )

ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')

plt.show()
```



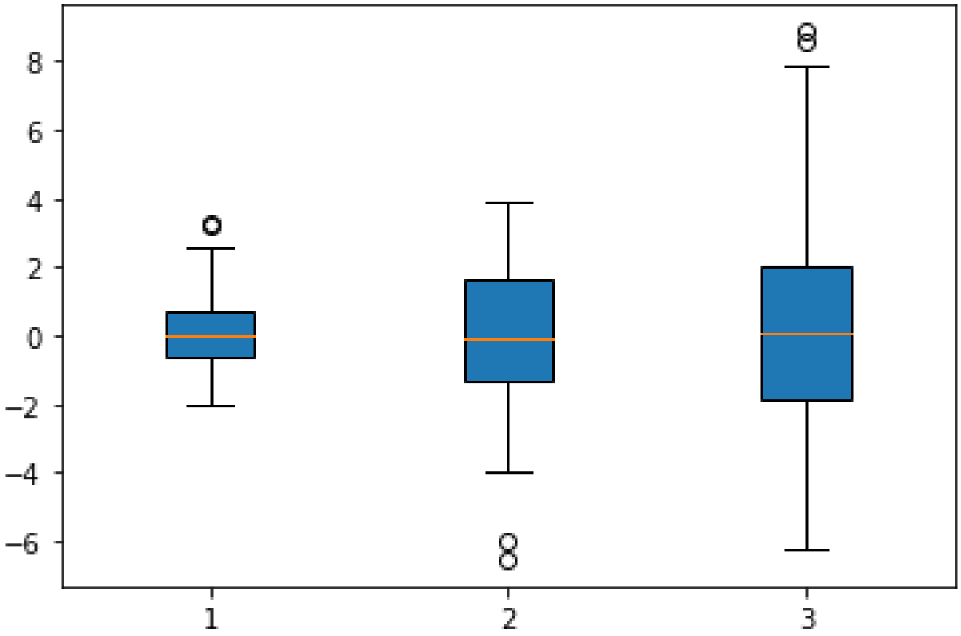
[0]:
```
# BARPLOT + BOXPLOT
```

[0]:
```
from random import sample
import random
random.seed(667)
data = sample(range(1, 1000), 100)
plt.hist(data)
```

[0]:
```
(array([ 9., 10.,  9.,  7., 13., 14.,  7., 11., 11.,  9.]),
 array([ 18. , 115.6, 213.2, 310.8, 408.4, 506. , 603.6, 701.2, 798.8,
        896.4, 994. ]),
 <a list of 10 Patch objects>)
```



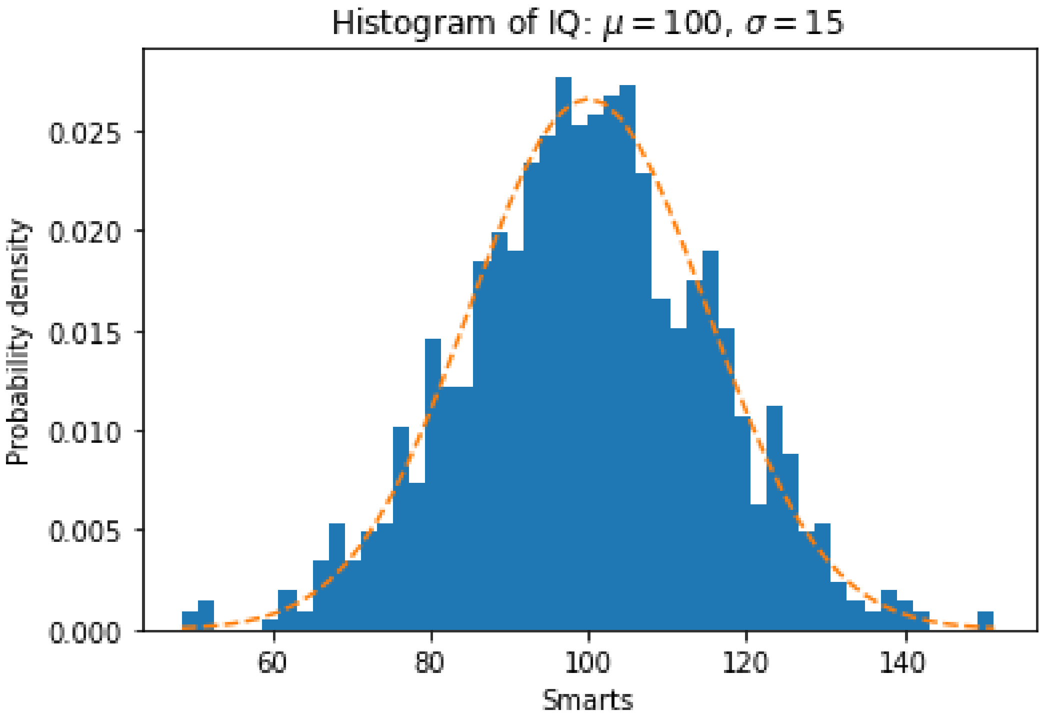
[0]:
```
data = [np.random.normal(0, std, 100) for std in range(1, 4)]

# rectangular box plot
plt.boxplot(data,vert=True,patch_artist=True);
```



[0]:

[0]:

```python
import matplotlib
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(667)

# example data
mu = 100  # mean of distribution
sigma = 15  # standard deviation of distribution
x = mu + sigma * np.random.randn(1000)

num_bins = 50

fig, ax = plt.subplots()

# the histogram of the data
n, bins, patches = ax.hist(x, num_bins, density=1)

# add a 'best fit' line
y = ((1 / (np.sqrt(2 * np.pi) * sigma)) *np.exp(-0.5 * (1 / sigma * (bins - mu))**2))


ax.plot(bins, y, '--')
ax.set_xlabel('Smarts')
ax.set_ylabel('Probability density')
ax.set_title(r'Histogram of IQ: $\mu=100$, $\sigma=15$')

# Tweak spacing to prevent clipping of ylabel
plt.show()
```



[0]:

```
```

[0]:

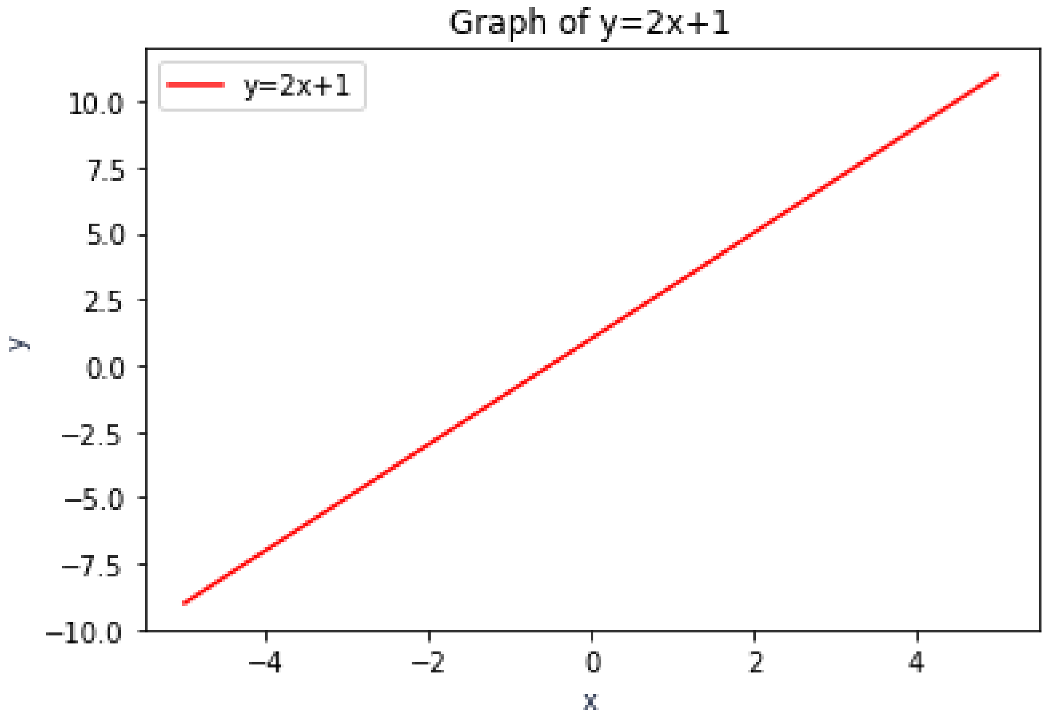
```
```

[0]:

```
## ESERCITAZIONE LAB
```

[0]:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.linspace(-?,?,100)
y = ???????
plt.plot(x, y, '-r', label='y=2x+1')
plt.title('Graph of y=2x+1')
plt.xlabel('x', color='#1C2843' )
plt.ylabel('y', color='#1C2843')
plt.legend(loc='upper left')
#plt.grid()
plt.show()
```

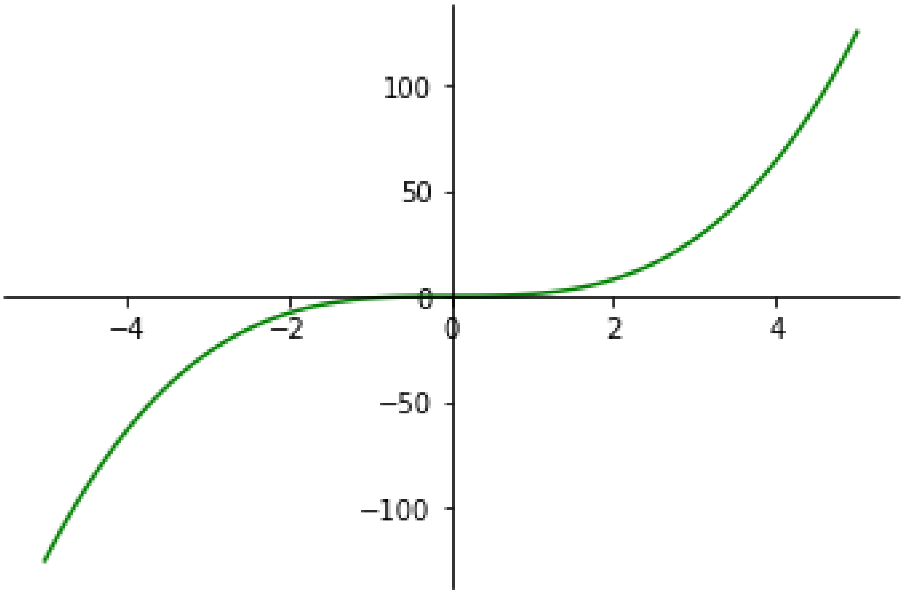


[0]:

```
# 100 linearly spaced numbers
x = np.linspace(?,?,?)

# the function, which is y = ??? here
y = ???


# setting the axes at the centre
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('center')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')


# plot the function
plt.plot(x,y, '?')
```

[0]:

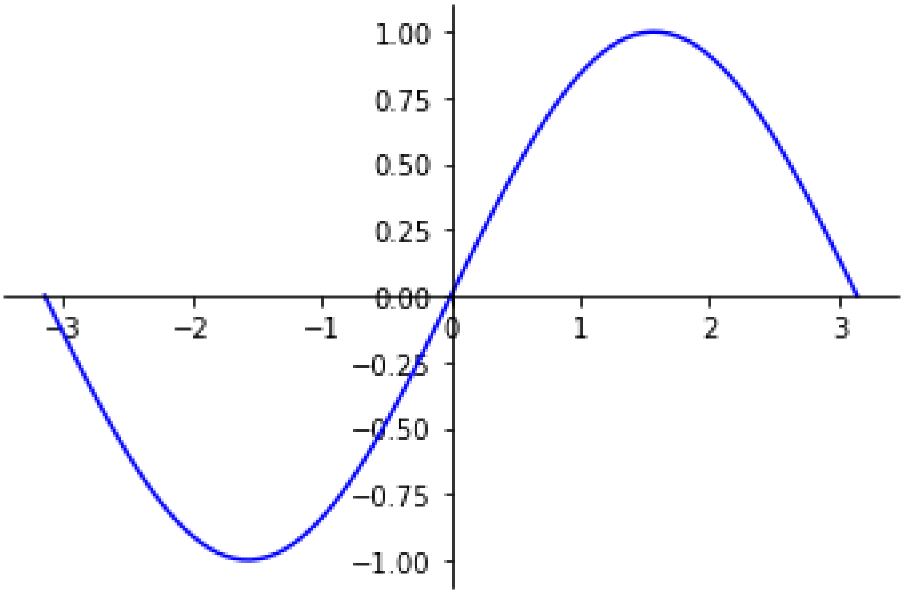
```
[<matplotlib.lines.Line2D at 0x7fb5e46f8208>]
```



[0]:

[0]:

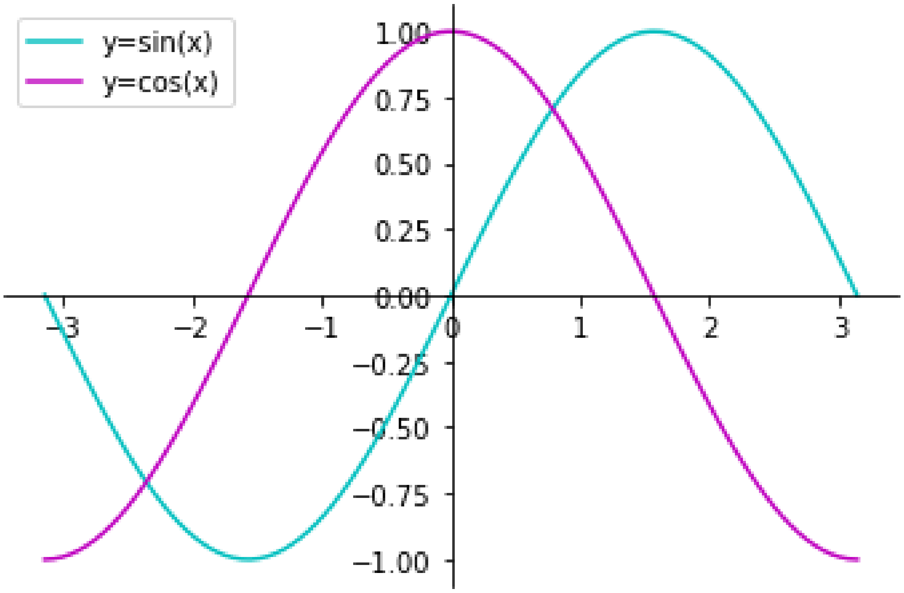
```
# 100 linearly spaced numbers
x = np.linspace(?,?,?)

# the function, which is y = sin(x) here
y = np.??(x)

# setting the axes at the centre
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('center')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')

# plot the function
plt.plot(x,y, 'b')

# show the plot
plt.show()
```



[0]:

```
import matplotlib.pyplot as plt
import numpy as np

# 100 linearly spaced numbers
x = np.linspace(?,?,100)

# the functions, which are y = ? and z = ? here
y = np.?(x)
z = np.?(x)

# setting the axes at the centre
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('center')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')

# plot the functions
plt.plot(x,y, 'c', label='y=sin(x)')
plt.plot(x,z, 'm', label='y=cos(x)')

plt.legend(loc='upper left')

# show the plot
```

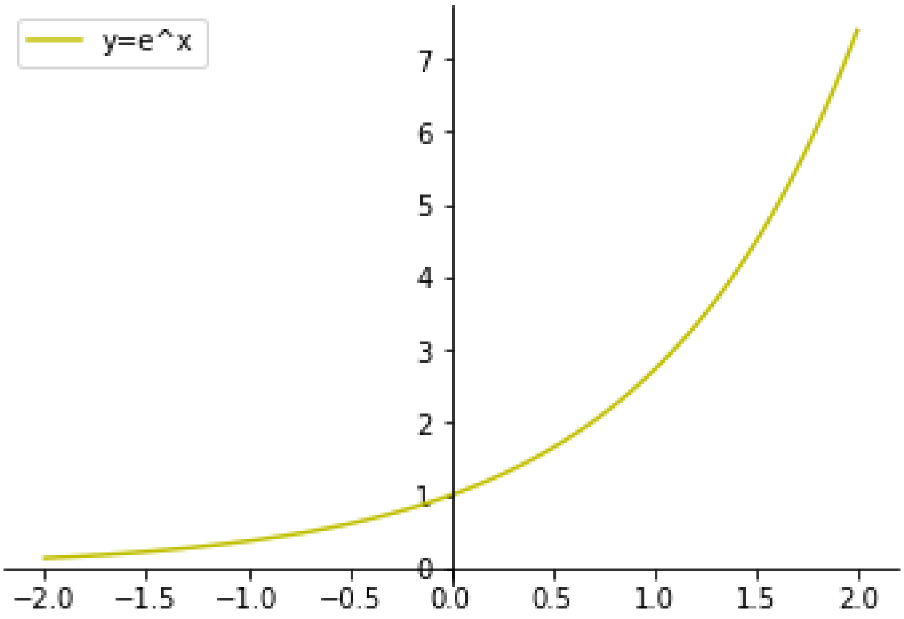
```
plt.show()
```



[0]:
```
np.??
```

[0]:
```
3.141592653589793
```

[0]:
```
# 100 linearly spaced numbers
x = np.linspace(?,?,?)

# the function, which is y = e^x here
y = ????

# setting the axes at the centre
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')

# plot the function
plt.plot(x,y, 'y', label='y=e^x')
plt.legend(loc='upper left')

# show the plot
plt.show()
```



[0]:

[0]:

[0]:

[0]:

[0]: