Sandeep Sharma

Date : - 9/4/2020

Matplotlib

Matplotlib is a python library originated from MATLAB, which is being used for Data Visualization.By using Matplotlib, we can show the insights found from the analysis of the data. In data science visualization is the important step. By using visualization we can easily understand than how data is split. It was created to replicate MatLab's plotting capabilities in Python.

- Easy to use
- Custom labels and texts
- Great control over every element in a figure

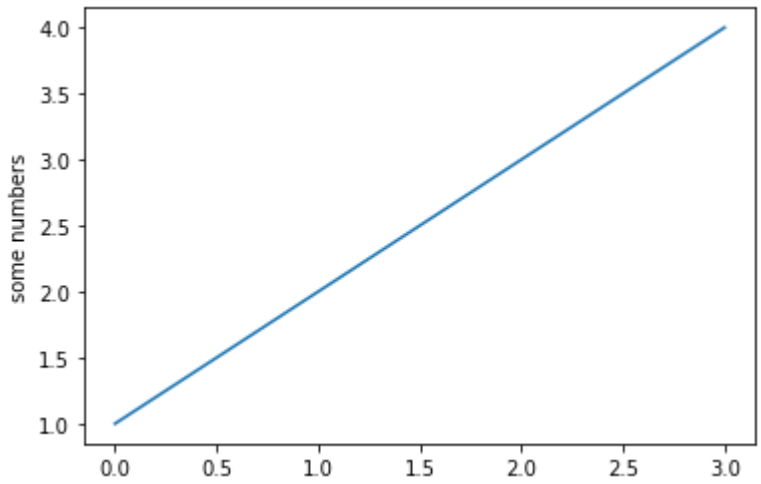
In [1]:
```
!pip install matplotlib
```

```
Requirement already satisfied: matplotlib in c:\users\sshar127\anaconda3\lib
\site-packages (3.2.2)
Requirement already satisfied: numpy>=1.11 in c:\users\sshar127\anaconda3\lib
\site-packages (from matplotlib) (1.18.5)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in
c:\users\sshar127\anaconda3\lib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: cycler>=0.10 in c:\users\sshar127\anaconda3\li
b\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\sshar127\anacond
a3\lib\site-packages (from matplotlib) (1.2.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\sshar127\anac
onda3\lib\site-packages (from matplotlib) (2.8.1)
Requirement already satisfied: six in c:\users\sshar127\anaconda3\lib\site-pa
ckages (from cycler>=0.10->matplotlib) (1.15.0)
```

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

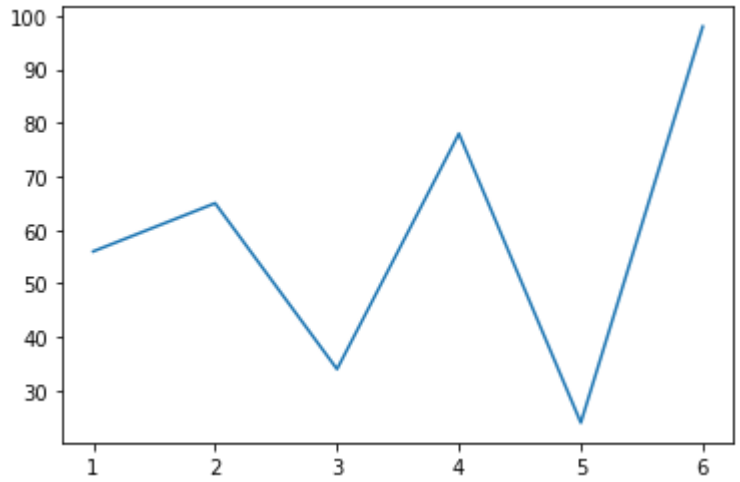
In [3]:
```
%matplotlib inline
```

In [5]:
```
plt.plot([1,2,3,4])
plt.ylabel('some numbers')
plt.show()
```



In [6]:
```
x1 = [1,2,3,4,5,6]
y1 = [56,65,34,78,24,98]
plt.plot(x1,y1)
```

Out[6]:
```
[<matplotlib.lines.Line2D at 0x242c38ab070>]
```



In [7]:
```
x = np.linspace(1,50,20)
print (x)
```

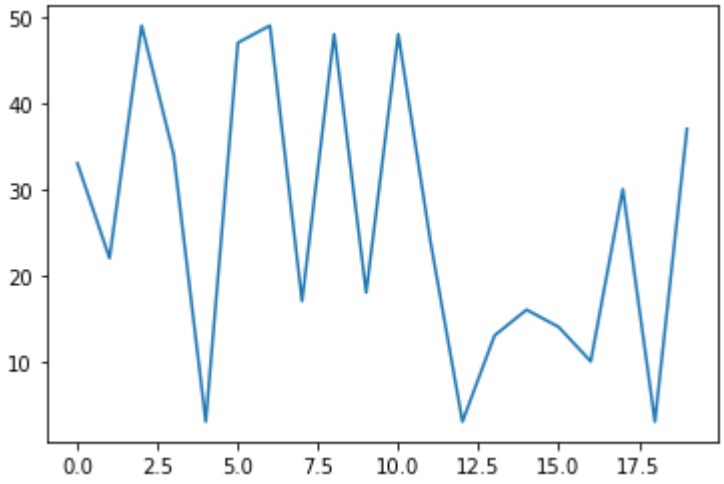
```
[ 1.          3.57894737  6.15789474  8.73684211 11.31578947 13.89473684
 16.47368421 19.05263158 21.63157895 24.21052632 26.78947368 29.36842105
 31.94736842 34.52631579 37.10526316 39.68421053 42.26315789 44.84210526
 47.42105263 50.        ]
```

In [8]:
```
y = np.random.randint(1, 50, 20)
print (y)
```

```
[33 22 49 34  3 47 49 17 48 18 48 24  3 13 16 14 10 30  3 37]
```

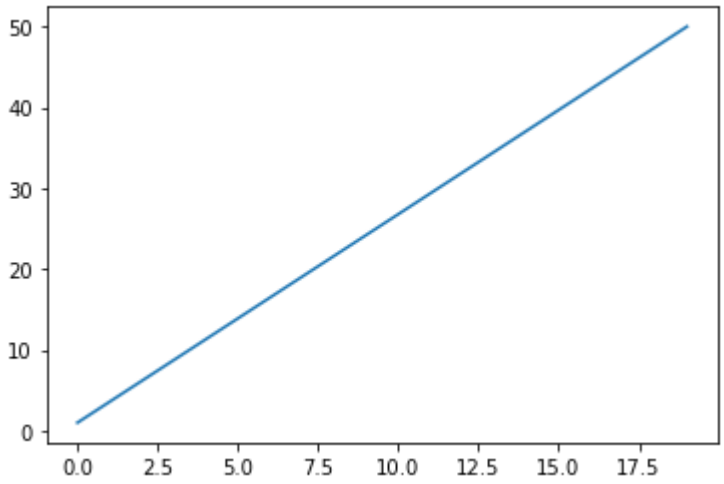
In [9]:
```
plt.plot(y)
```

Out[9]:
```
[<matplotlib.lines.Line2D at 0x242c3907520>]
```



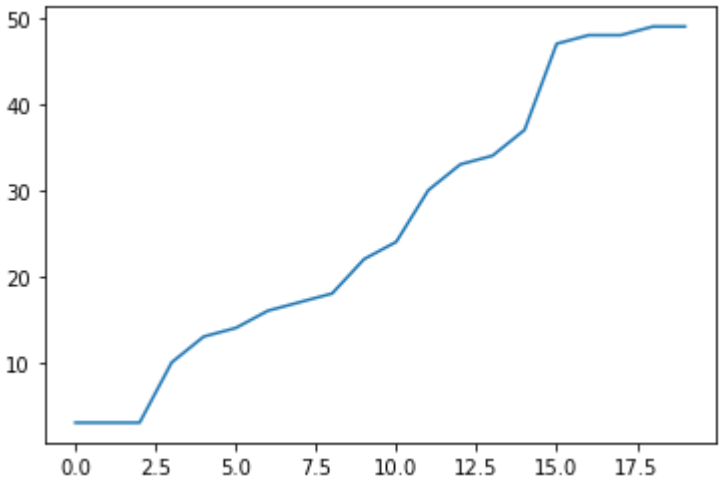
In [10]:
```
plt.plot(x)
```

Out[10]:
```
[<matplotlib.lines.Line2D at 0x242c395f1c0>]
```



In [11]:
```
# Sorting
y = np.sort(y)
plt.plot(y)
```

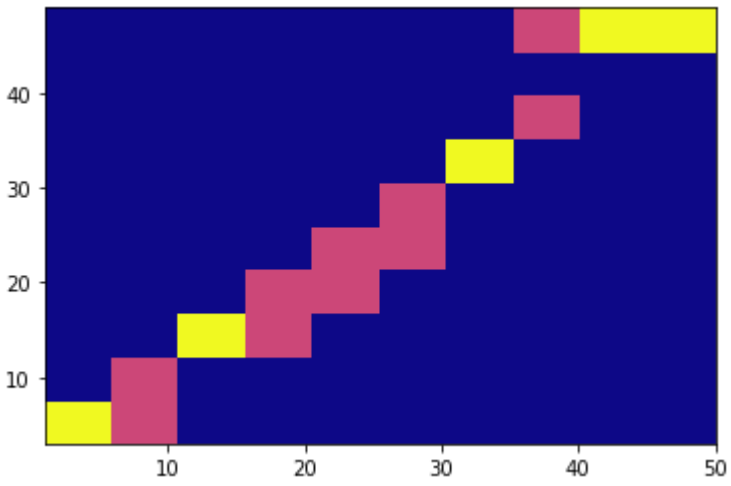
Out[11]:
```
[<matplotlib.lines.Line2D at 0x242c39b4430>]
```



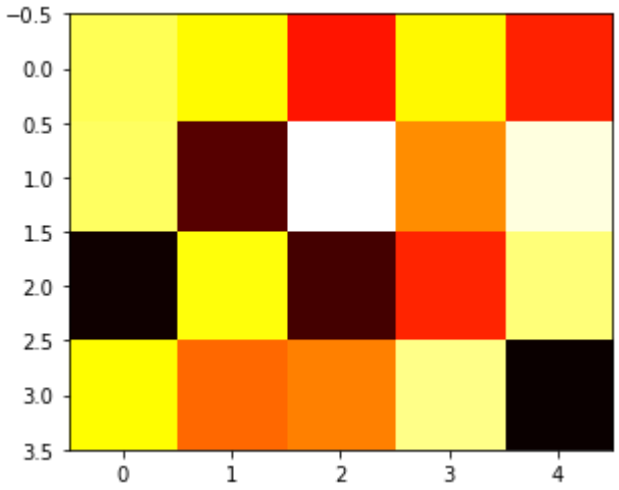
In [12]:
```
# Histogram 2d
plt.hist2d(x, y, bins=10, normed=False, cmap='plasma')
```

```
<ipython-input-12-bb84d7458ed0>:2: MatplotlibDeprecationWarning: The 'normed'
parameter of hist2d() has been renamed 'density' since Matplotlib 3.1; suppor
t for the old name will be dropped in 3.3.
  plt.hist2d(x, y, bins=10, normed=False, cmap='plasma')
```

Out[12]:
```
(array([[2., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [1., 1., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 2., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 1., 1., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 1., 1., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 1., 1., 0., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 2., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 0., 1., 0., 1.],
        [0., 0., 0., 0., 0., 0., 0., 0., 0., 2.],
        [0., 0., 0., 0., 0., 0., 0., 0., 0., 2.]]),
 array([ 1. ,  5.9, 10.8, 15.7, 20.6, 25.5, 30.4, 35.3, 40.2, 45.1, 50. ]),
 array([ 3. ,  7.6, 12.2, 16.8, 21.4, 26. , 30.6, 35.2, 39.8, 44.4, 49. ]),
 <matplotlib.collections.QuadMesh at 0x242c3a01fa0>)
```

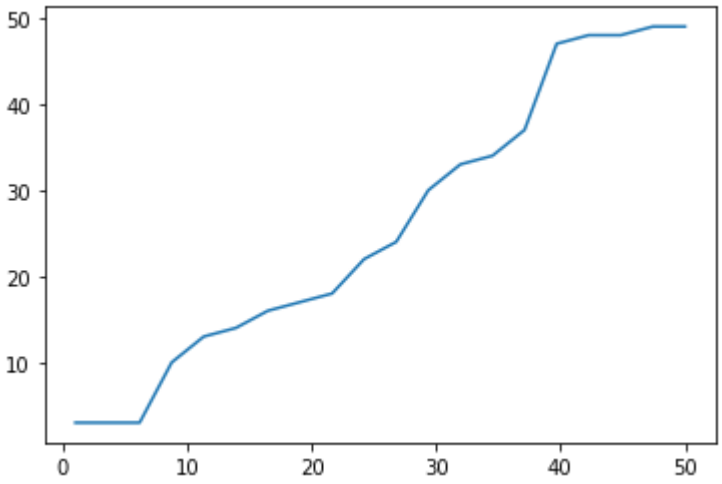


In [13]:
```
# Heatmap
a = np.random.random((4, 5))
plt.imshow(a, cmap='hot', interpolation='nearest')
plt.show()
```



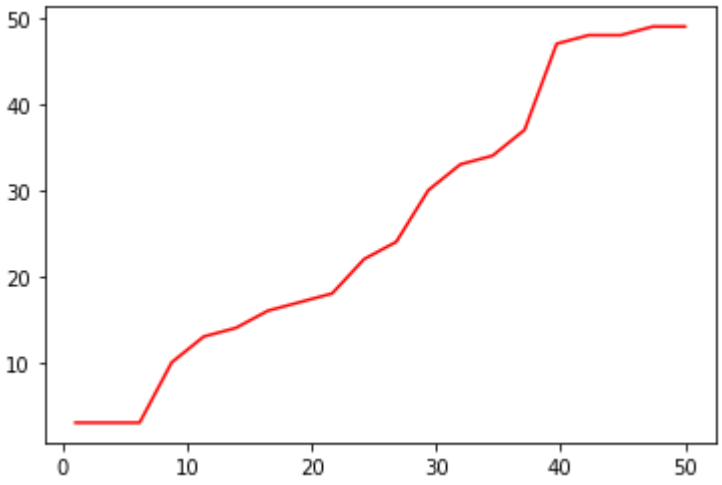
In [14]:
```
# for x-axis and y-axis
plt.plot(x, y)
```

Out[14]:
```
[<matplotlib.lines.Line2D at 0x242c3ab8a90>]
```



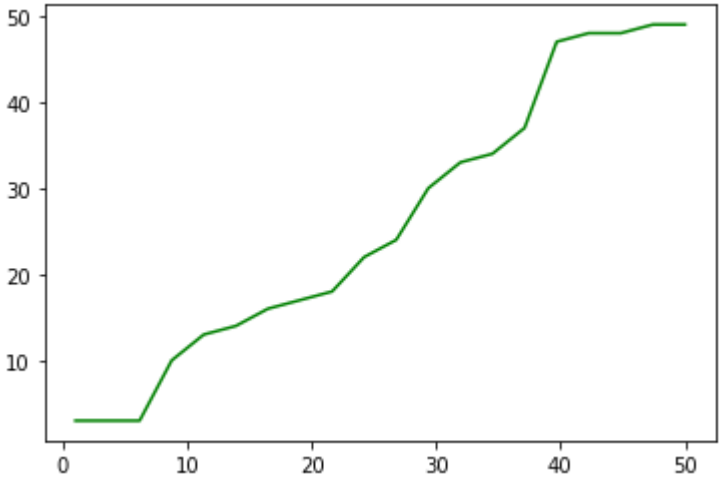
In [15]:
```
# to change color
plt.plot(x, y, 'r')
```

Out[15]:
```
[<matplotlib.lines.Line2D at 0x242c3b08c10>]
```



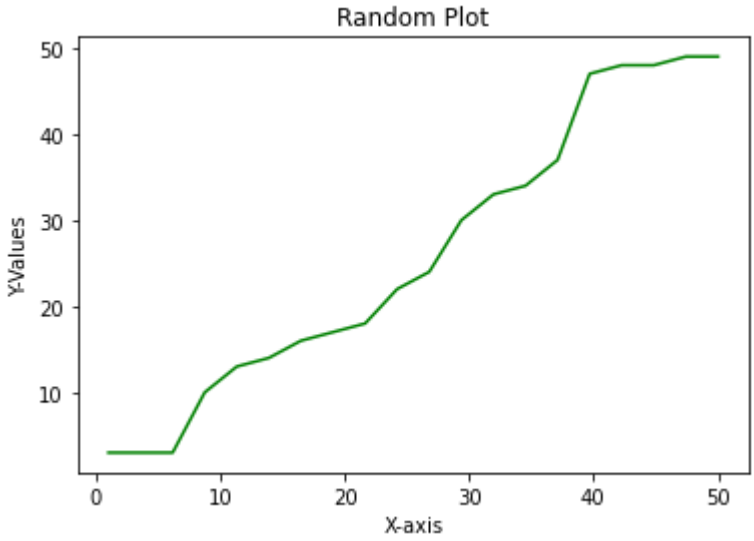
In [16]:
```
plt.plot(x, y, color = 'g')
```

Out[16]:
```
[<matplotlib.lines.Line2D at 0x242c3b5ad60>]
```



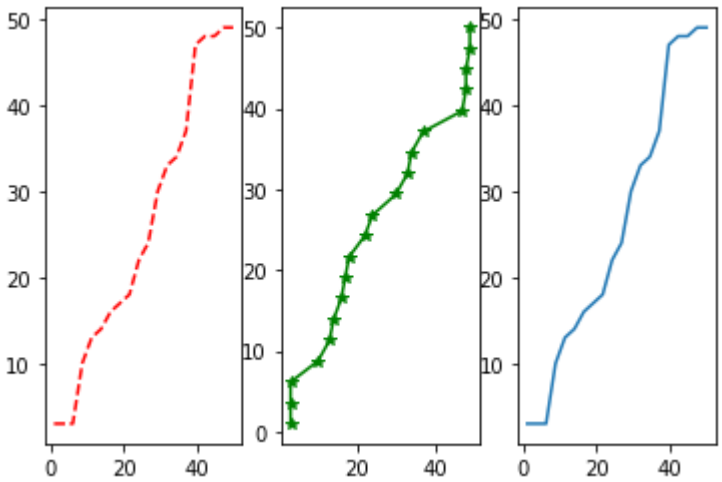
In [19]:
```
# labeling X-Axis, y-Axis and Title
plt.plot(x, y, color = 'g')
plt.xlabel('X-axis')
plt.ylabel('Y-Values')
plt.title('Random Plot')
```

Out[19]:
```
Text(0.5, 1.0, 'Random Plot')
```



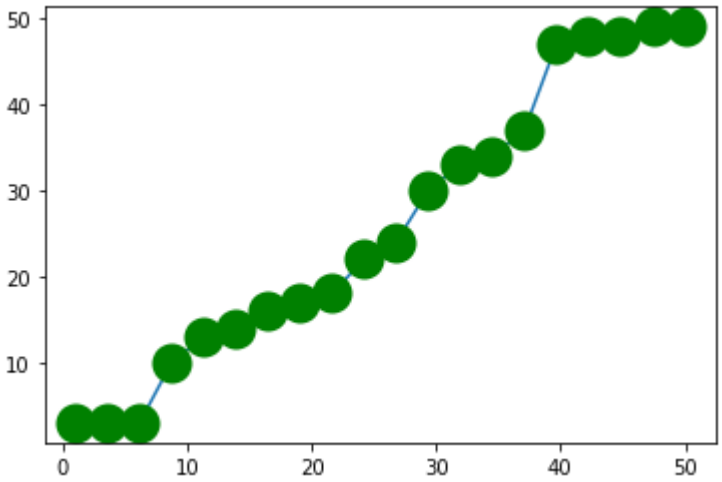
In [20]:
```
# plt.subplot(nrows, ncols, plot number)
plt.subplot(1,3,1)
plt.plot(x, y, 'r--') # More on color options later
plt.subplot(1,3,2)
plt.plot(y, x, 'g*-')
plt.subplot(1,3,3)
plt.plot(x,y)
```

Out[20]:
```
[<matplotlib.lines.Line2D at 0x242c3c41c40>]
```



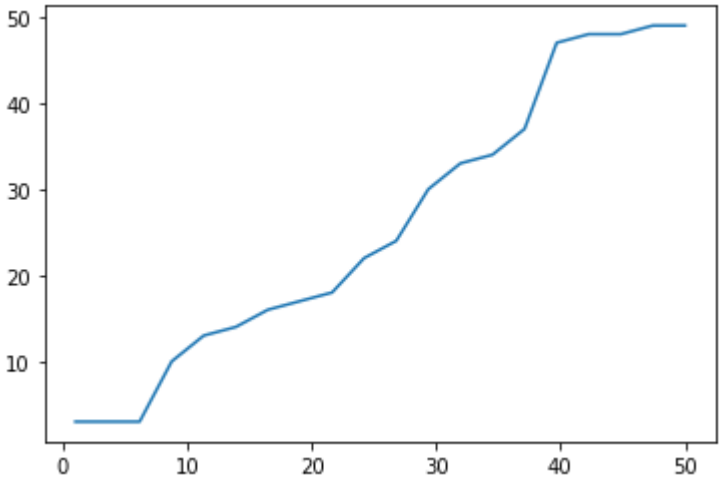
In [21]:
```
plt.plot(x,y, marker = 'o', markersize = 10, markerfacecolor = 'r', markeredge
color = 'g',
         markeredgewidth = 10)
```

Out[21]:
```
[<matplotlib.lines.Line2D at 0x242c3cb2a90>]
```



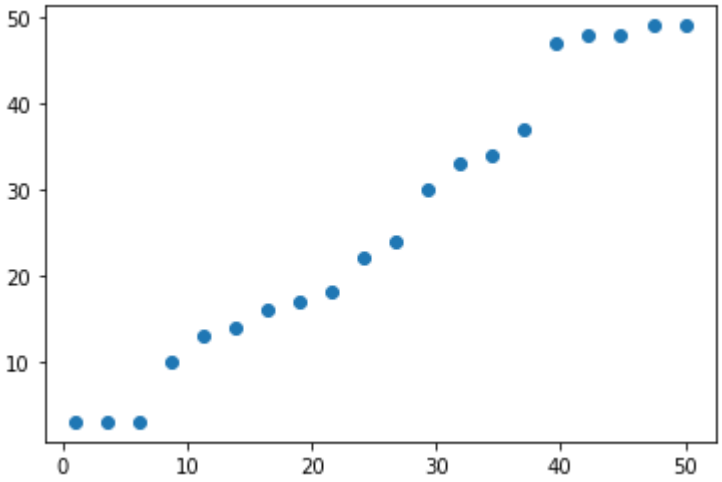
In [22]:
```
plt.plot(x,y)
```

Out[22]:
```
[<matplotlib.lines.Line2D at 0x242c3d03850>]
```



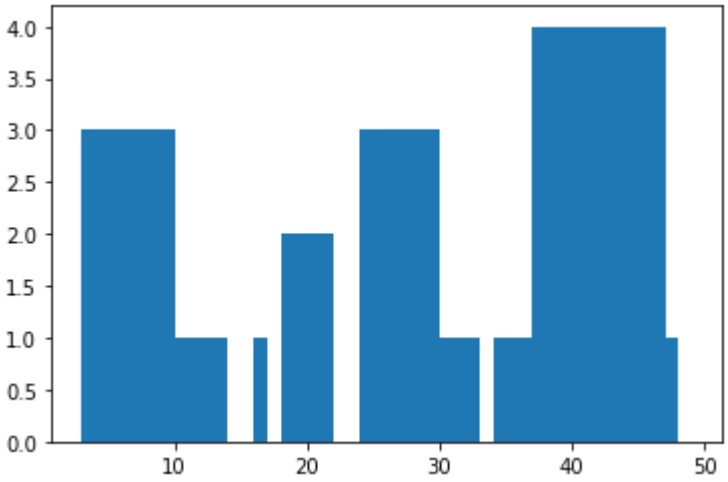
In [23]:
```
plt.scatter(x, y)
```

Out[23]:
```
<matplotlib.collections.PathCollection at 0x242c3d52820>
```



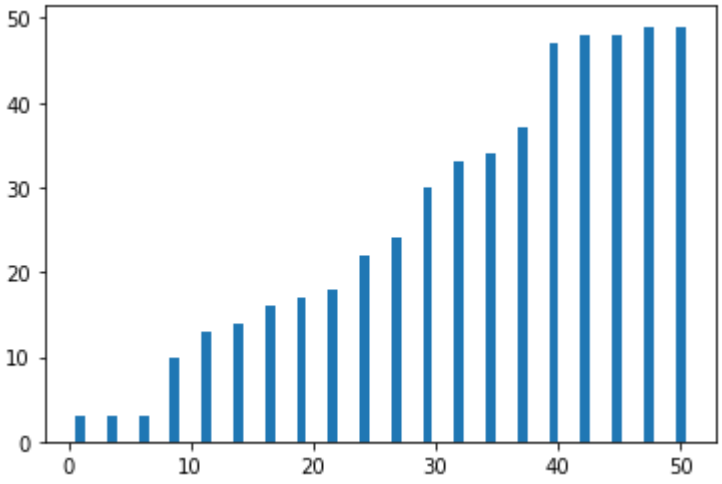
In [24]:
```
plt.hist(x, y)
```

Out[24]:
```
(array([0., 0., 3., 1., 1., 0., 1., 0., 2., 0., 3., 1., 0., 1., 4., 1., 0.,
        0., 0.]),
 array([ 3,  3,  3, 10, 13, 14, 16, 17, 18, 22, 24, 30, 33, 34, 37, 47, 48,
        48, 49, 49]),
 <a list of 19 Patch objects>)
```



In [25]:
```
plt.bar(x,y)
```

Out[25]:
```
<BarContainer object of 20 artists>
```



In [26]:
```
plt.fill(x,y)
```

Out[26]:
```
[<matplotlib.patches.Polygon at 0x242c4e81e80>]
```