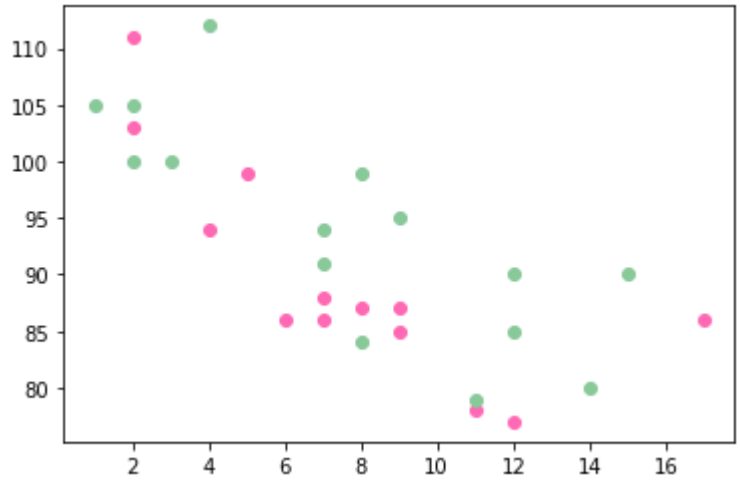
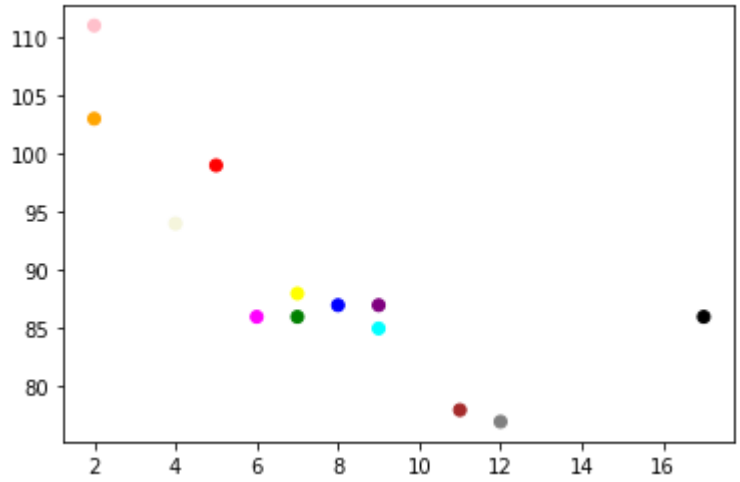
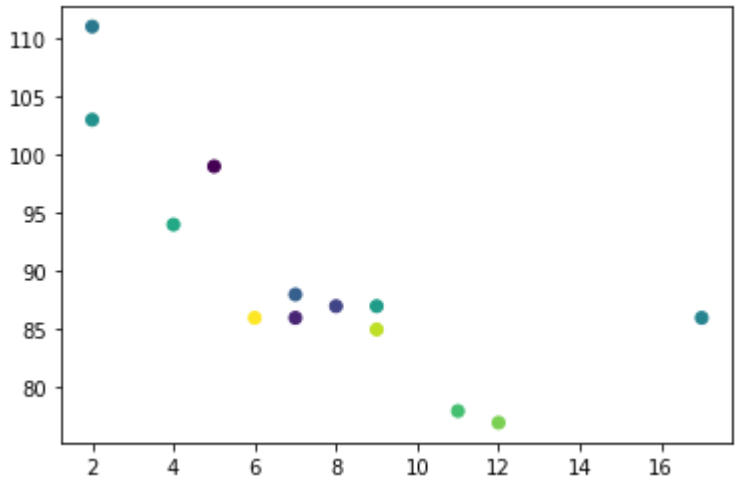
In [24]:
```
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
plt.scatter(x, y, color = 'hotpink')

x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])
plt.scatter(x, y, color = '#88c999')

plt.show()
```



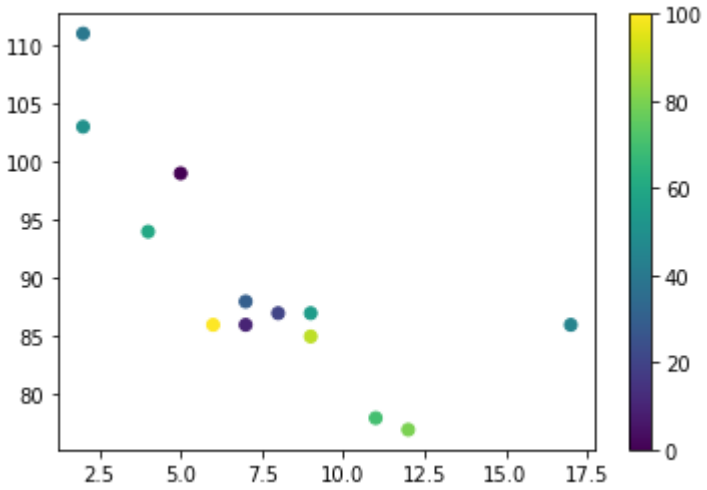
In [25]:
```
#color each dot
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
colors = np.array(["red","green","blue","yellow","pink","black","orange","purple","beig

plt.scatter(x, y, c=colors)

plt.show()
```



In [26]:
```
#Create a color array, and specify a colormap in the scatter plot
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
colors = np.array([0, 10, 20, 30, 40, 45, 50, 55, 60, 70, 80, 90, 100])

plt.scatter(x, y, c=colors, cmap='viridis')

plt.show()
```

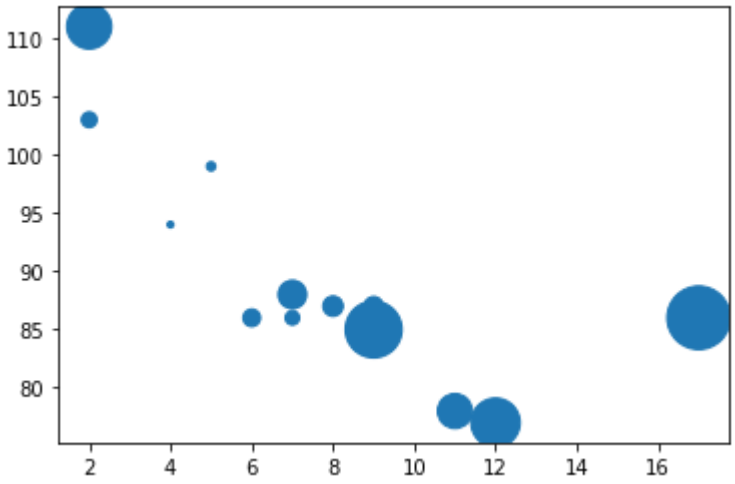


In [27]:
```
#Include the actual colormap:
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
colors = np.array([0, 10, 20, 30, 40, 45, 50, 55, 60, 70, 80, 90, 100])

plt.scatter(x, y, c=colors, cmap='viridis')

plt.colorbar()

plt.show()
```



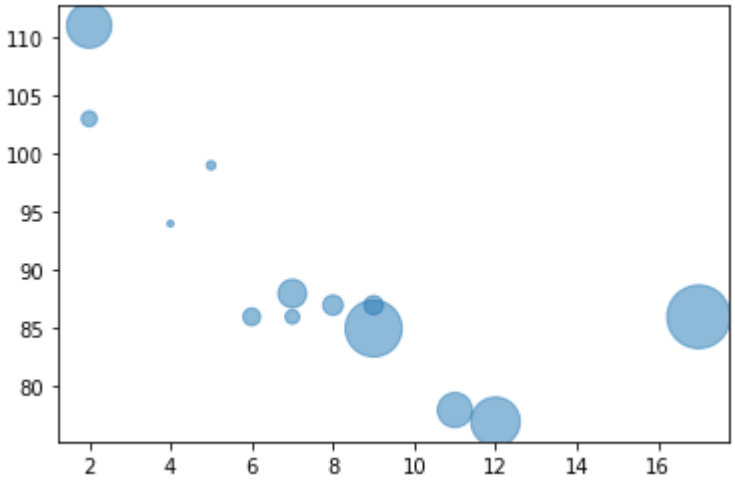
In [28]:

```
#set your own size of the markers
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])

plt.scatter(x, y, s=sizes)

plt.show()
```



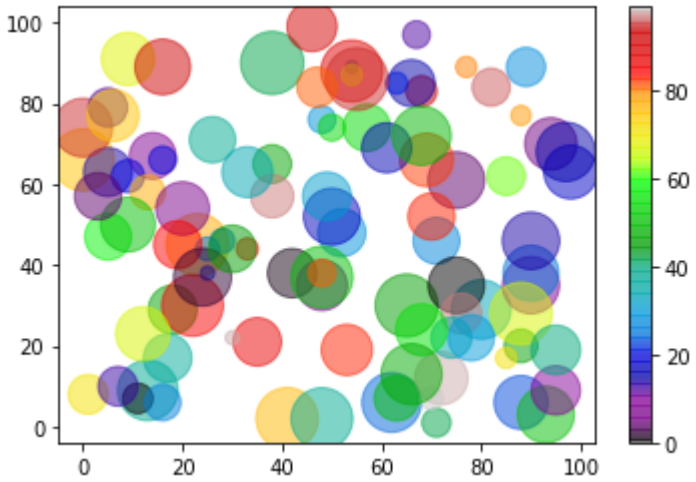
In [29]:

```
#adjust the transparency of the dots with the alpha argument
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])

plt.scatter(x, y, s=sizes, alpha=0.5)

plt.show()
```



In [30]:

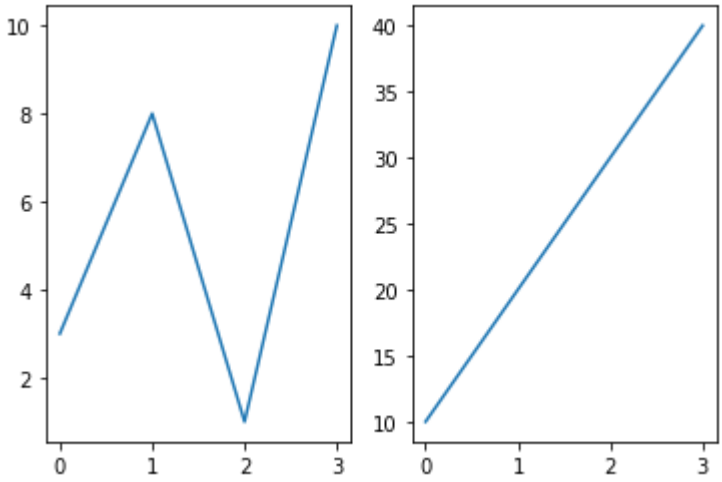
```
#Create random arrays with 100 values for x-points, y-points, colors and sizes
import matplotlib.pyplot as plt
import numpy as np

x = np.random.randint(100, size=(100))
y = np.random.randint(100, size=(100))
colors = np.random.randint(100, size=(100))
sizes = 10 * np.random.randint(100, size=(100))

plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='nipy_spectral')

plt.colorbar()

plt.show()
```



In [19]:

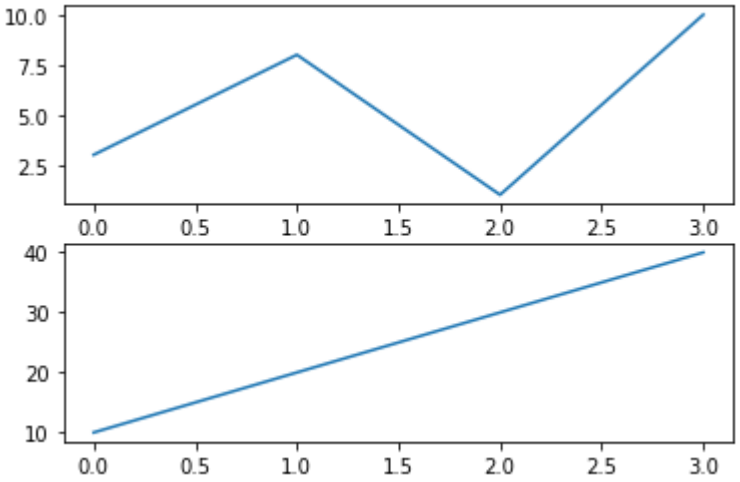
```
import matplotlib.pyplot as plt
import numpy as np

#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(1, 2, 1)
plt.plot(x,y)

#plot 2:
```

```
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(1, 2, 2)
plt.plot(x,y)

plt.show()
```



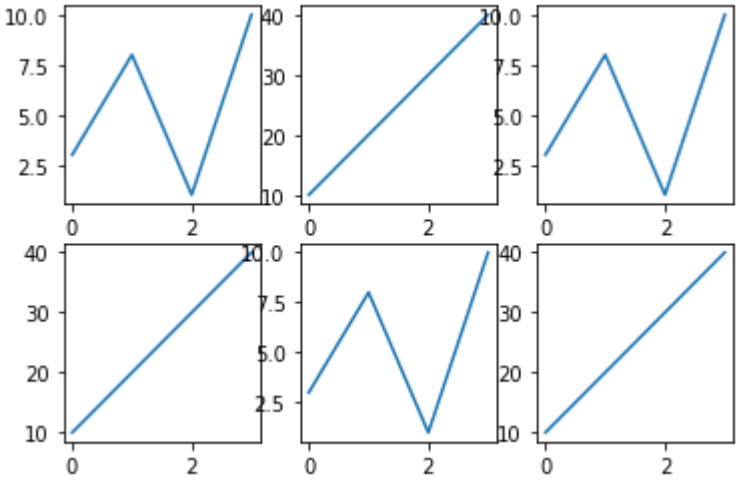
In [20]:

```
#Draw 2 plots on top of each other:
import matplotlib.pyplot as plt
import numpy as np

#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 1, 1)
plt.plot(x,y)

#plot 2:
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 1, 2)
plt.plot(x,y)

plt.show()
```



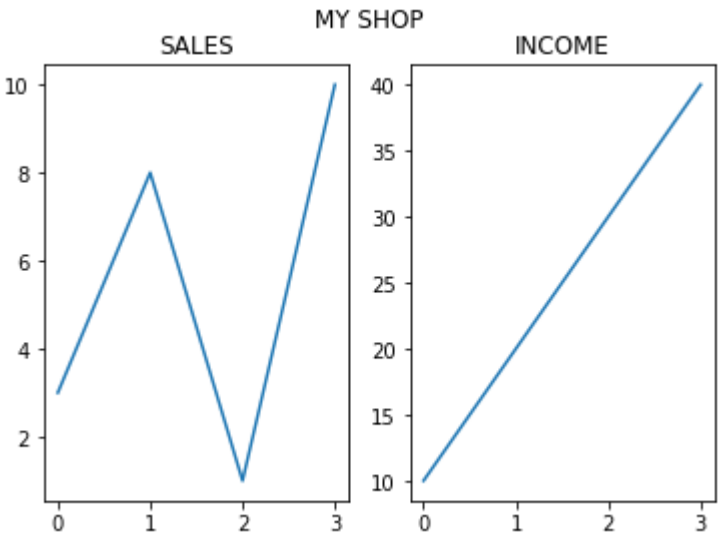
In [21]:

```
#Draw 6 plots:
import matplotlib.pyplot as plt
import numpy as np

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 1)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 2)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 3)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 4)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 5)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 6)
plt.plot(x,y)

plt.show()
```



In [22]:

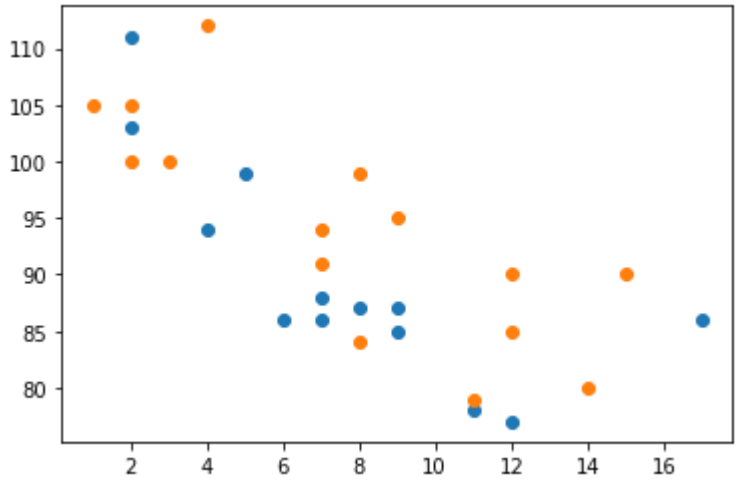
```
import matplotlib.pyplot as plt
import numpy as np

#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(1, 2, 1)
plt.plot(x,y)
plt.title("SALES")

#plot 2:
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(1, 2, 2)
plt.plot(x,y)
plt.title("INCOME")

plt.suptitle("MY SHOP")
plt.show()
```



In [23]:

```
#Compare Plots
import matplotlib.pyplot as plt
```

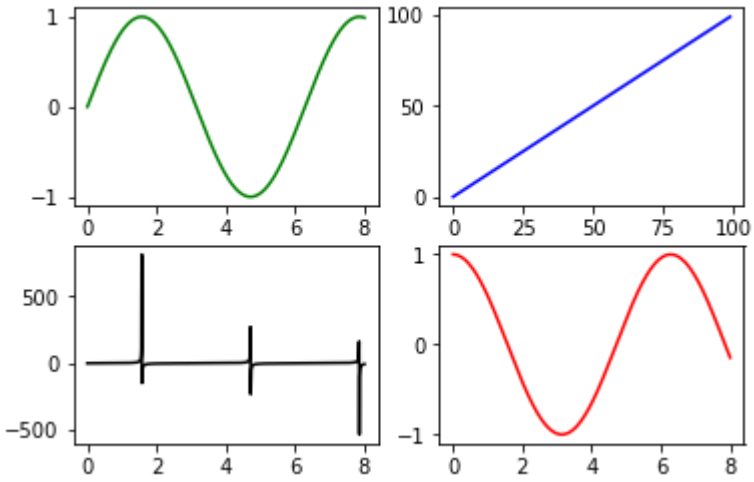
```
import numpy as np

#day one, the age and speed of 13 cars:
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
plt.scatter(x, y)

#day two, the age and speed of 15 cars:
x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])
plt.scatter(x, y)

plt.show()
```



In [2]:

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

fig, ax = plt.subplots(2, 2)
x = np.linspace(0, 8, 1000)

ax[0, 0].plot(x, np.sin(x), 'g') #row=0, col=0
ax[1, 0].plot(x, np.tan(x), 'k') #row=1, col=0
ax[0, 1].plot(range(100), 'b') #row=0, col=1
ax[1, 1].plot(x, np.cos(x), 'r') #row=1, col=1
fig.show();
```

```
<ipython-input-2-9eb047bef8a6>:11: UserWarning: Matplotlib is currently using module://i
pykernel.pylab.backend_inline, which is a non-GUI backend, so cannot show the figure.
  fig.show();
```



In [ ]:

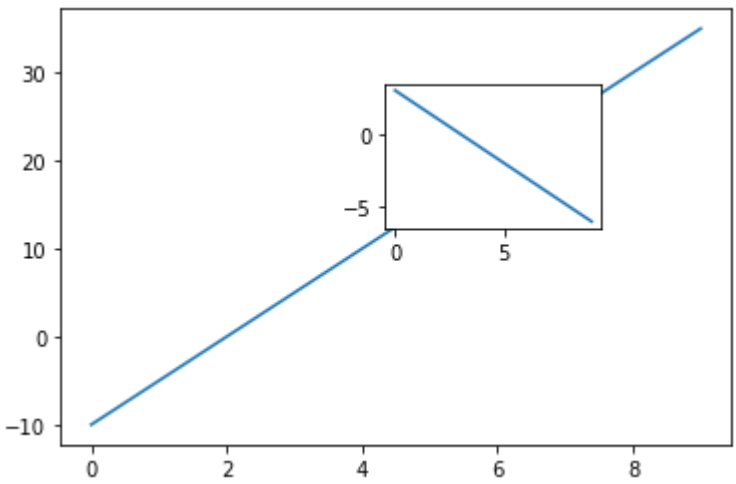
```
```

In [3]:

```
# Different functions in different axis
x= np.arange(0,10,1)
y1 = 5*x -10
y2 = -1*x +3

# plot
ax1 = plt.axes() # standard axes
ax2 = plt.axes([0.5, 0.5, 0.25, 0.25])
ax1.plot(x,y1)
ax2.plot(x,y2)
```

Out[3]:

```
[<matplotlib.lines.Line2D at 0x219c52143a0>]
```

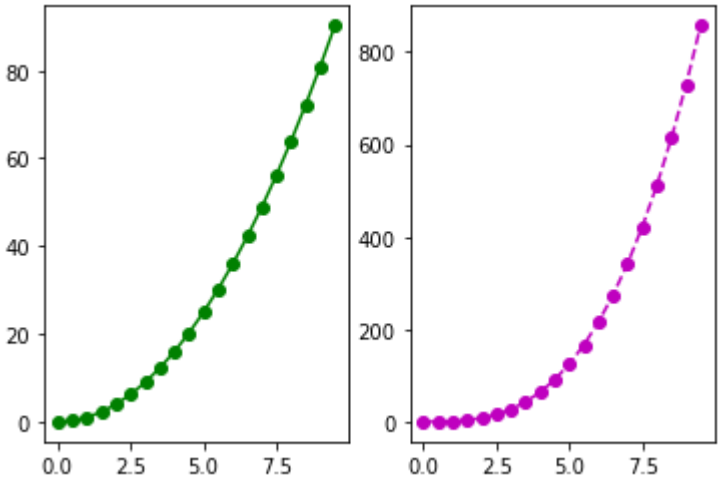


In [16]:

```
x =np.arange(0,10,0.5)
y1 = x*x
y2 = x*x*x
fig, axes = plt.subplots(1, 2 )
axes[0].plot(x, y1, 'g-o')
axes[1].plot(x, y2, 'm--o')
```

Out[16]:

```
[<matplotlib.lines.Line2D at 0x219c5196ca0>]
```

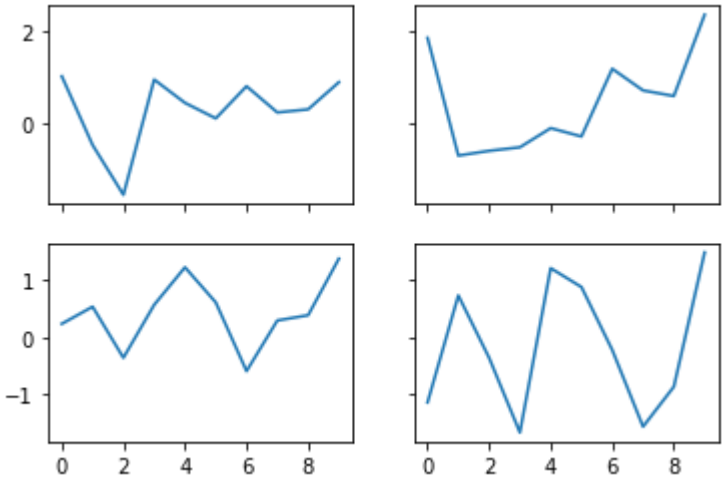


In [9]:

```
x  = np.arange(0,10,1)
y1 = np.random.randn(10)
y2 = np.random.randn(10)
y3 = np.random.randn(10)
y4 = np.random.randn(10)

# Create subplots
#fig, ax = plt.subplots(2, 2);
fig, ax = plt.subplots(2, 2, sharex='col', sharey='row');
y1 = np.random.randn(10)
y2 = np.random.randn(10)
y3 = np.random.randn(10)
y4 = np.random.randn(10)
ax[0][0].plot(x,y1)
ax[0][1].plot(x,y2)
ax[1][0].plot(x,y3)
ax[1][1].plot(x,y4)
```

Out[9]:

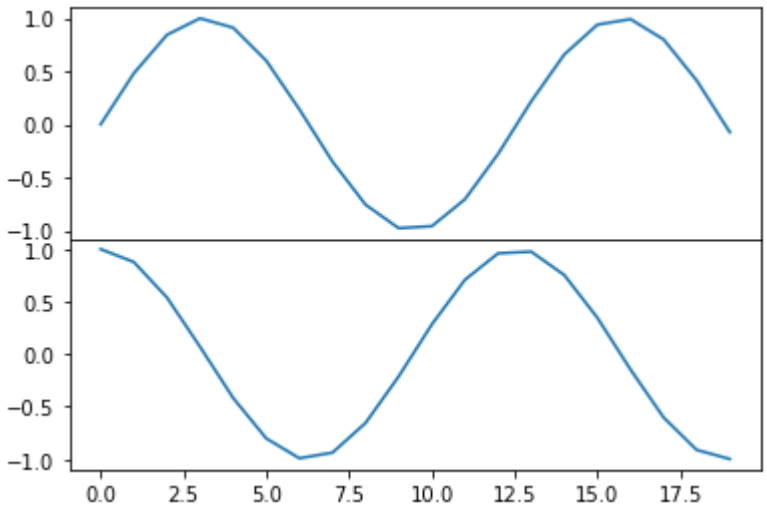
```
[<matplotlib.lines.Line2D at 0x219c5669a30>]
```



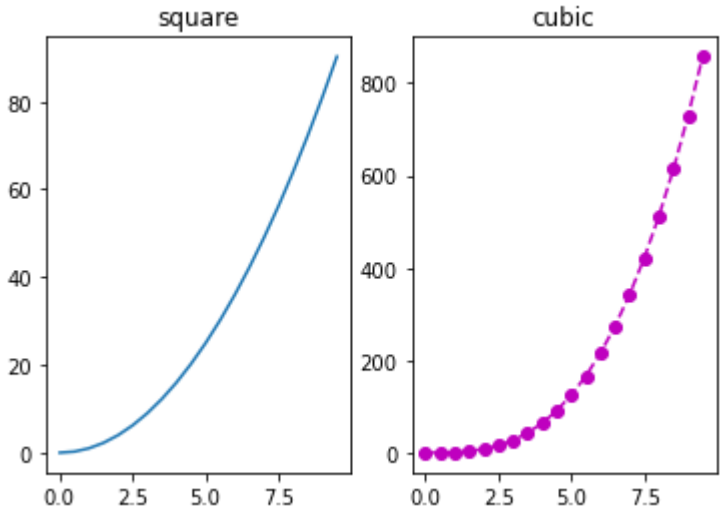
In [10]:

```
fig = plt.figure()
ax1 = fig.add_axes([0.1, 0.5, 0.8, 0.4])
ax2 = fig.add_axes([0.1, 0.1, 0.8, 0.4])
x = np.arange(0, 10,0.5)
ax1.plot(np.sin(x))
ax2.plot(np.cos(x))
```

Out[10]:
```
[<matplotlib.lines.Line2D at 0x219c57221c0>]
```



In [18]:
```
x =np.arange(0,10,0.5)
y1 = x*x
y2= x*x*x
fig, axes = plt.subplots(1, 2)
axes[0].plot(x, y1, '-')
axes[0].set_title('square')
axes[1].plot(x, y2, 'm--o')
axes[1].set_title('cubic');
```



In [ ]: