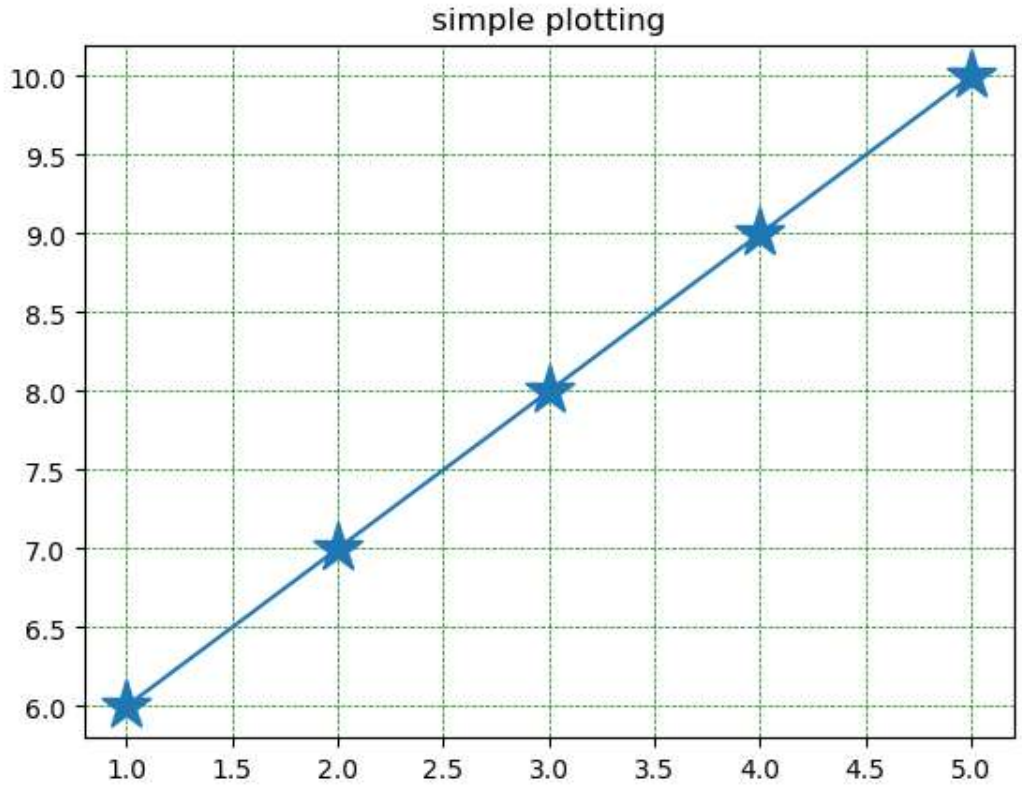
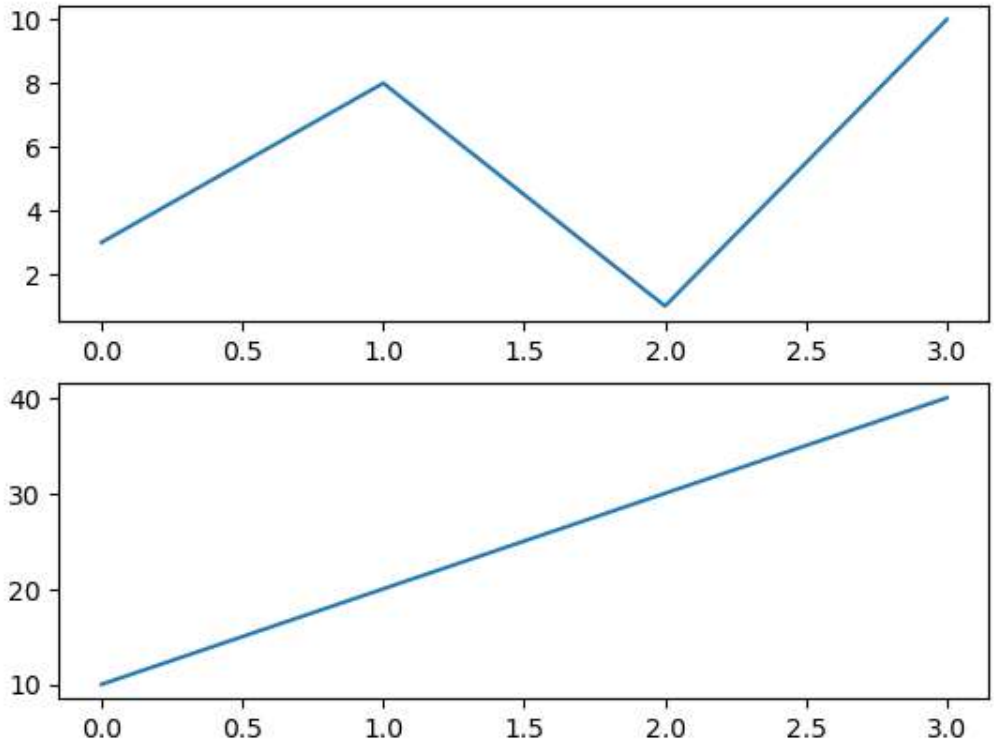
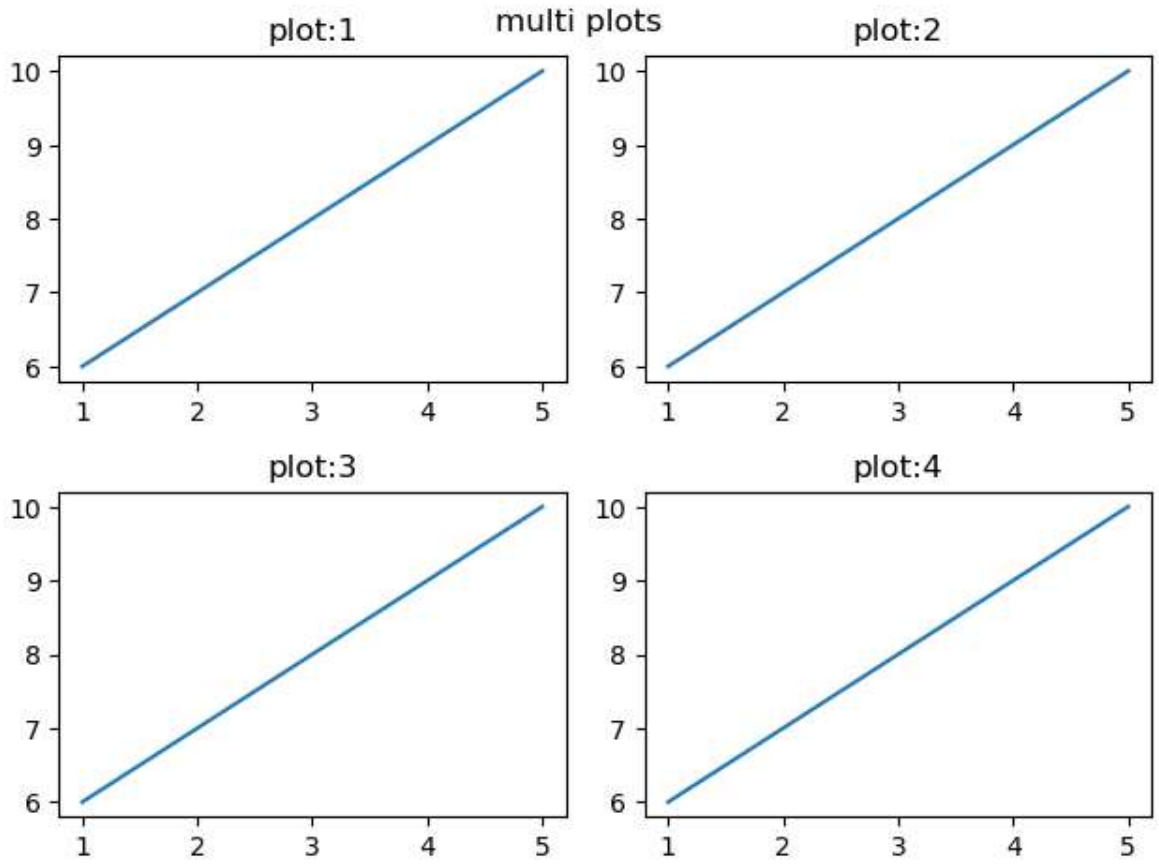
In [136]:
```
# Grid
from matplotlib import pyplot as plt
x=[1,2,3,4,5]
y=[6,7,8,9,10]
plt.plot(x,y,marker="*",ms="20")
plt.title("simple plotting")
plt.grid(axis="both")
plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)
plt.show()
```



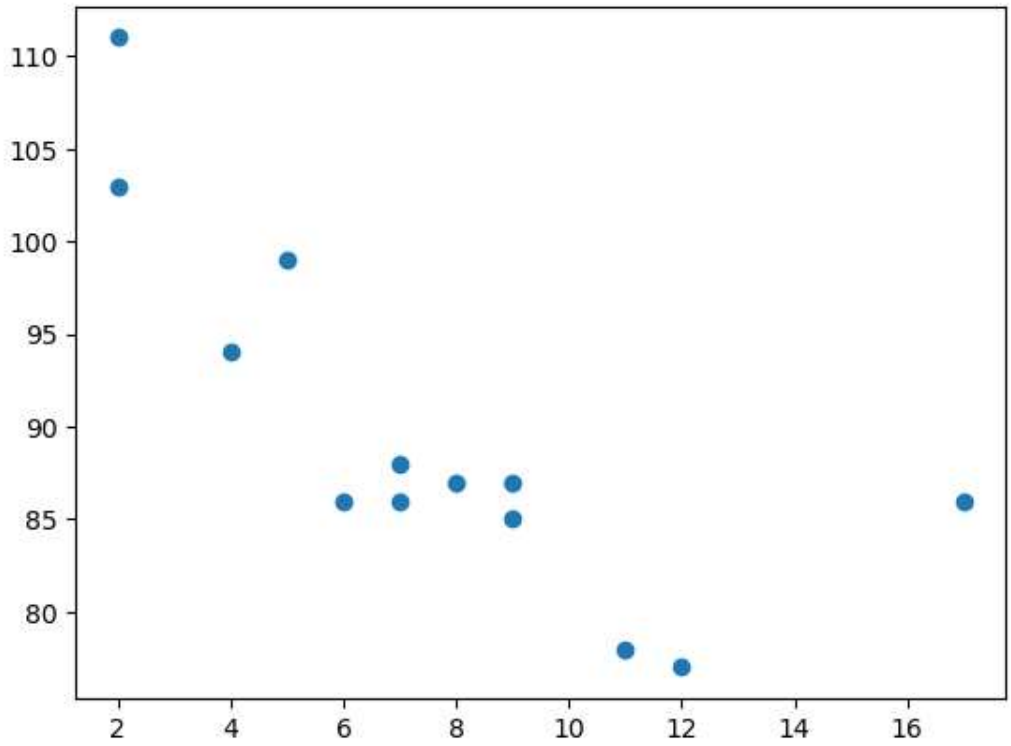
In [15]:
```
# sub plots(): function you can draw multiple plots in one figure.
# we have 3 arguments (row,column,index)

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()
```



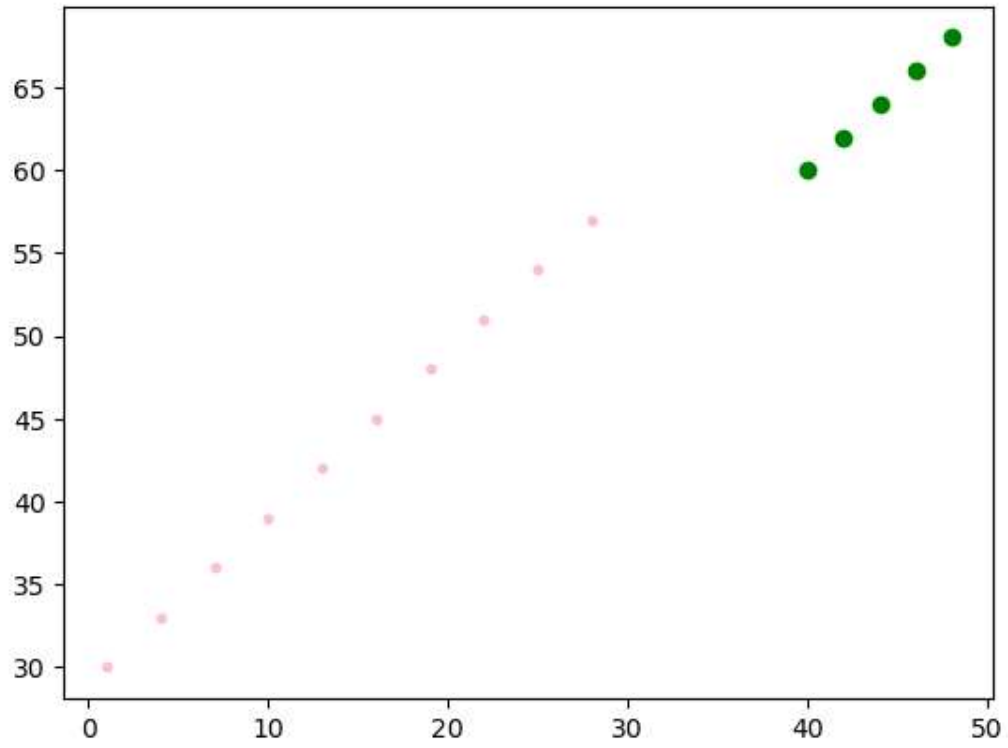
In [61]:
```
# multiple plots
from matplotlib import pyplot as plt
import random
x=[1,2,3,4,5]
y=[6,7,8,9,10]
plt.subplot(2,2,1)
plt.title("plot:1")
plt.plot(x,y)
#plot 2:
x=[1,2,3,4,5]
y=[6,7,8,9,10]
plt.subplot(2,2,2)
plt.title("plot:2")
plt.plot(x,y)
#plot 3:
x=[1,2,3,4,5]
y=[6,7,8,9,10]
plt.subplot(2,2,3)
plt.title("plot:3")
plt.plot(x,y)
#plot 4:
x=[1,2,3,4,5]
y=[6,7,8,9,10]
plt.subplot(2,2,4)
plt.title("plot:4")
plt.plot(x,y)
plt.tight_layout() # spaces between plots
plt.suptitle("multi plots")
plt.show()
```



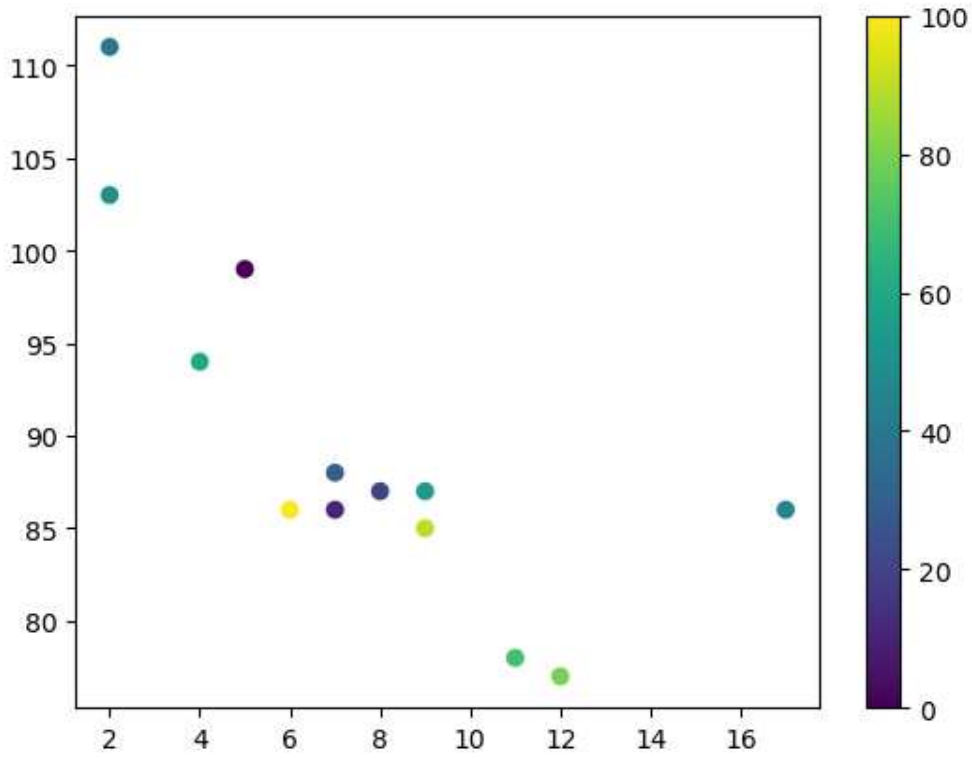
In [37]:
```
# scatter :function plots one dot for each observation needs two arrays with same lengt
from matplotlib import pyplot as plt
import numpy as np

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)
plt.show()
```



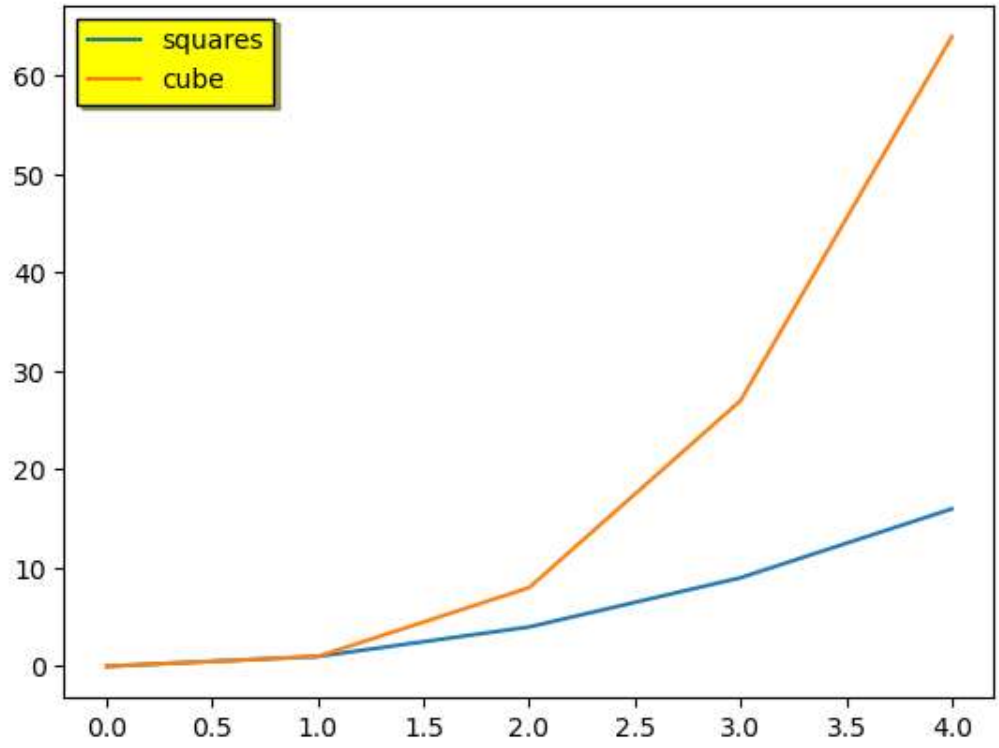
In [104]:
```
# giving colour to each scatter set and  two plots in same figure
import matplotlib.pyplot as plt
import numpy as np

x = range(1,30,3)
y = range(30,60,3)
plt.scatter(x, y, color = 'pink',s=10)

x = range(40,50,2)
y = range(60,70,2)
plt.scatter(x, y, color = 'green')

plt.show()
```



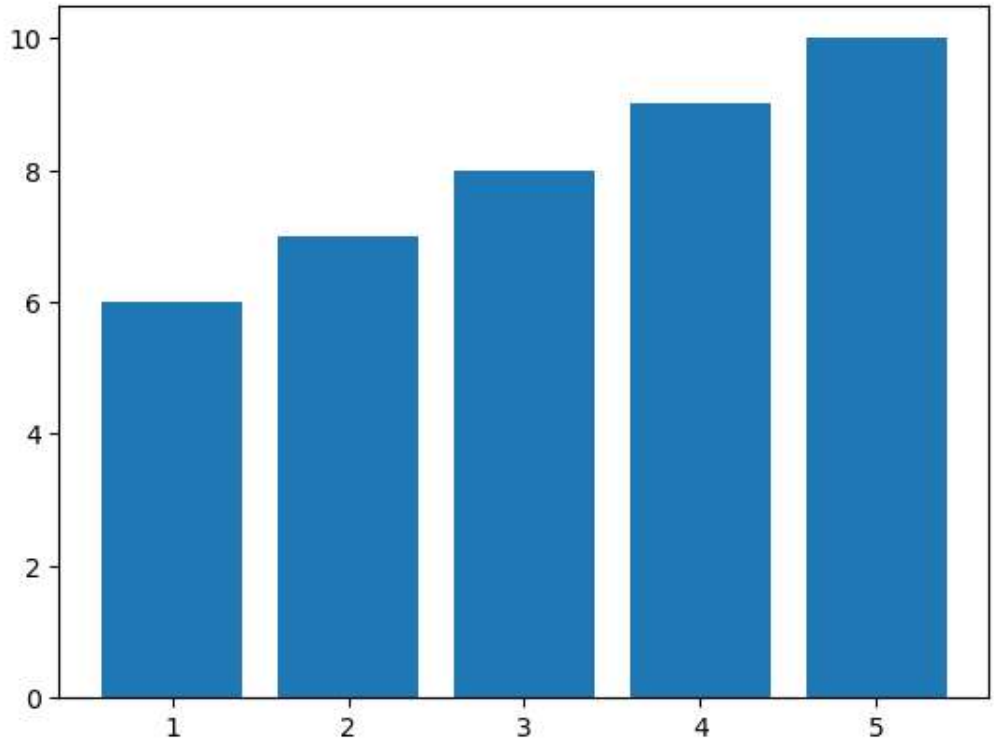
In [107]:
```
#giving different colors to the range
import matplotlib.pyplot as plt
import numpy as np

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()
```



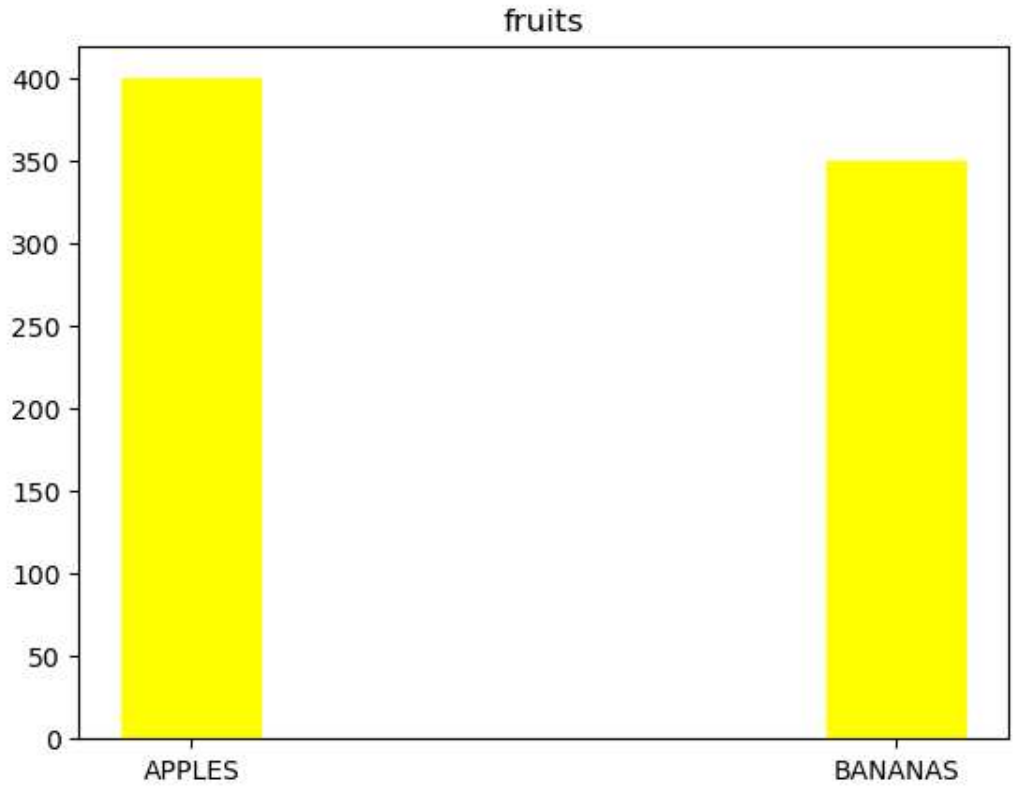
In [96]:
```
#legend: gives functionality of plotting

from matplotlib import pyplot as plt
import numpy as np
x=np.arange(0,5,1)
y1=x**2
y2=x**3
colo
#plt.plot(x,y1,x,y2)
#plt.legend(["squares","cubes"])
plt.plot(x,y1,label="squares")
x=np.arange(0,5,1)
y1=x**2
y2=x**3
plt.plot(x,y2, label="cube")
plt.legend(loc="upper left",framealpha=1,facecolor="yellow",edgecolor="black",fancybox=
plt.show()
```



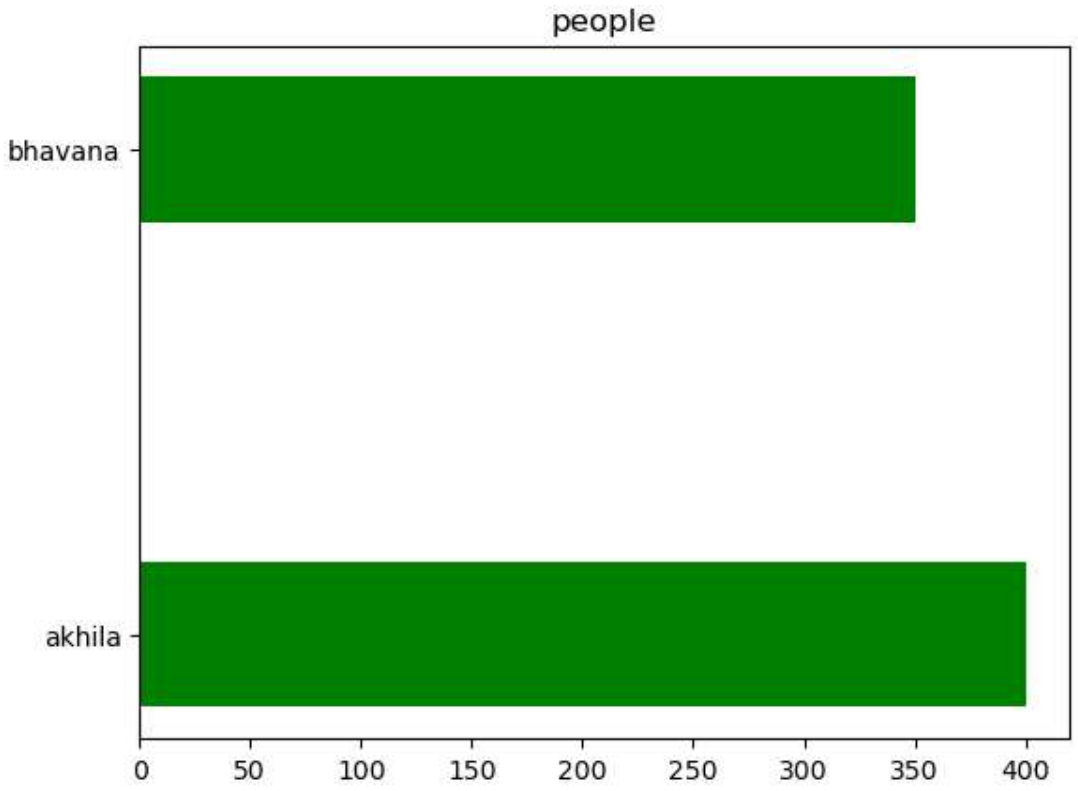
In [64]:
```
# bar
from matplotlib import pyplot as plt
d=[1,2,3,4,5]
x=[6,7,8,9,10]
plt.bar(d,x,)
plt.show()
```



In [87]:
```
# vertical bar : we can control the width
from matplotlib import pyplot as plt
x = ["APPLES", "BANANAS"]
y = [400, 350]
plt.title("fruits")
plt.bar(x, y,color="yellow",width=0.2)
plt.show()
```

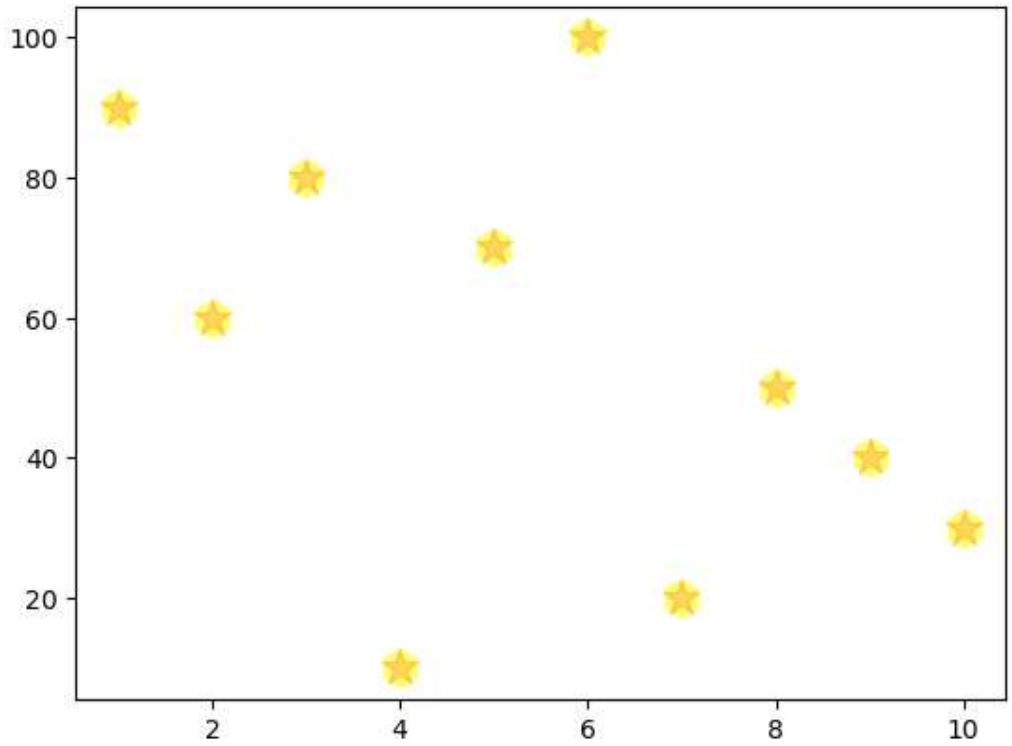


In [137]:
```
# horizontal bar: we can control the height
from matplotlib import pyplot as plt
x = ["akhila", "bhavana"]
y = [400, 350]
plt.title("people")
plt.barh(x, y,color="green",height=0.3)
plt.show()
```

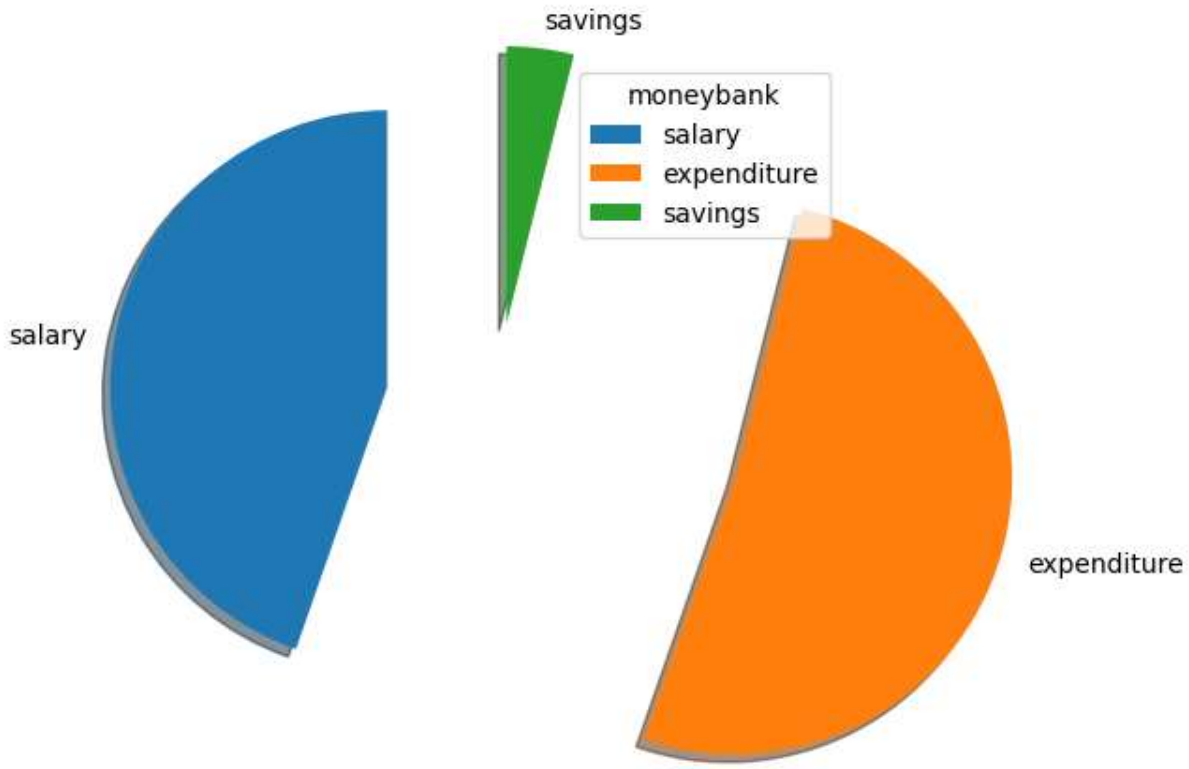


In [119]:
```
# scatterplot
from matplotlib import pyplot as plt
import random
x=[1,2,3,4,5,6,7,8,9,10]
y=[10,20,30,40,50,60,70,80,90,100]
random.shuffle(y)
print(x,y)
plt.scatter(x,y,color="red",marker="*",s=200,linewidth=1,alpha=0.3)
x1=[1,2,3,4,5,6,7,8,9,10]
y1=[10,20,30,40,50,60,70,80,90,100]
random.shuffle(y1)
print(x1,y1)
plt.scatter(x,y,color="yellow",marker="h",s=200,linewidth=1,alpha=0.5)
plt.show()
```

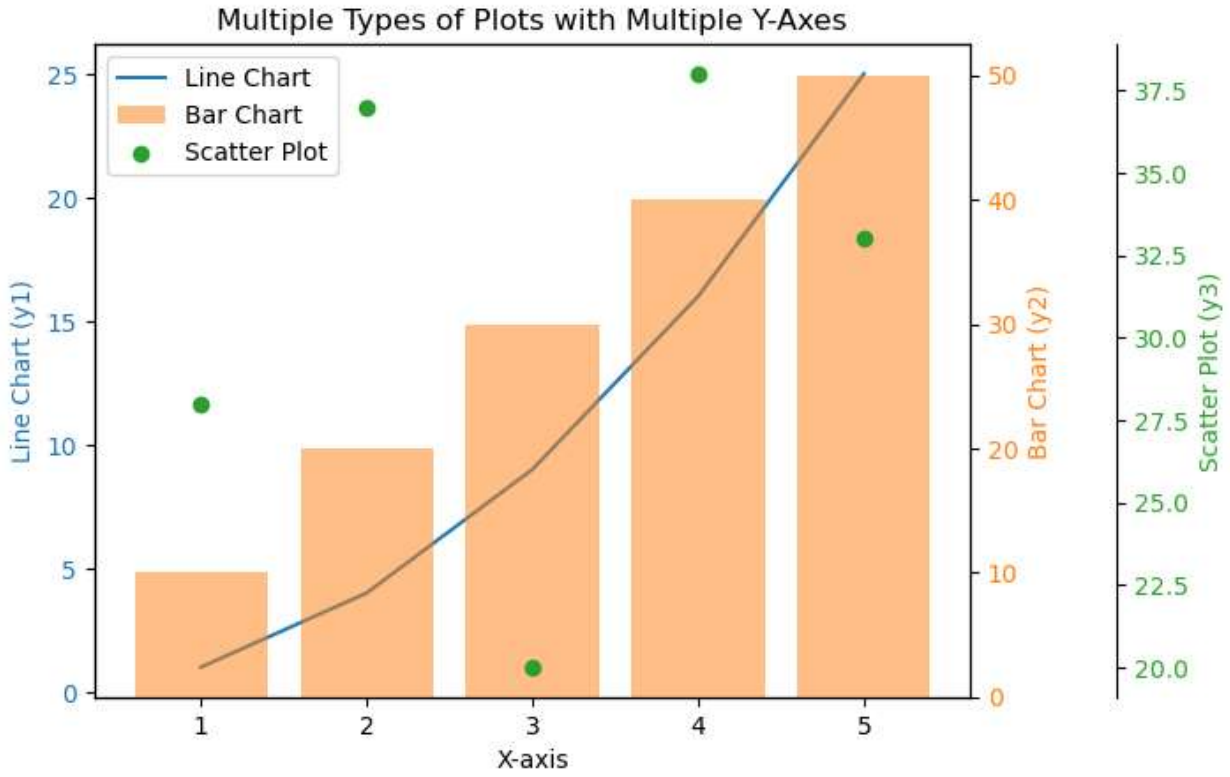
```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10] [90, 60, 80, 10, 70, 100, 20, 50, 40, 30]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10] [90, 80, 30, 60, 10, 70, 20, 40, 100, 50]
```



In [131]:
```
# pie charts:using startangle,explode and shadow
import matplotlib.pyplot as plt
import numpy as np

y = np.array([68.5,78.9,6])
label=["salary","expenditure","savings"]
exp_lode=[0.4,0.9,0.3]

plt.pie(y,labels=label,startangle=90,shadow=True,explode=exp_lode)
plt.legend(loc="upper right",title="moneybank")
plt.show()
```



In [132]:
```
#scatterplot,barchart,linechart in one visualization
import matplotlib.pyplot as plt
import numpy as np

# Sample data
x = np.arange(1, 6)
y1 = x**2
y2 = x*10
y3 = np.random.randint(20, 40, size=5)

# Create the first plot (line chart)
fig, ax1 = plt.subplots()

ax1.set_xlabel('X-axis')
ax1.set_ylabel('Line Chart (y1)', color='tab:blue')
ax1.plot(x, y1, color='tab:blue', label='Line Chart')
ax1.tick_params(axis='y', labelcolor='tab:blue')

# Create the second plot (bar chart)
ax2 = ax1.twinx()

ax2.set_ylabel('Bar Chart (y2)', color='tab:orange')
ax2.bar(x, y2, color='tab:orange', alpha=0.5, label='Bar Chart')
ax2.tick_params(axis='y', labelcolor='tab:orange')

# Create the third plot (scatter plot)
ax3 = ax1.twinx()

ax3.spines['right'].set_position(('outward', 60))  # Adjust the position of the third p
ax3.set_ylabel('Scatter Plot (y3)', color='tab:green')
ax3.scatter(x, y3, color='tab:green', marker='o', label='Scatter Plot')
ax3.tick_params(axis='y', labelcolor='tab:green')

# Combine legends
lines1, labels1 = ax1.get_legend_handles_labels()
lines2, labels2 = ax2.get_legend_handles_labels()
lines3, labels3 = ax3.get_legend_handles_labels()
lines = lines1 + lines2 + lines3
labels = labels1 + labels2 + labels3
plt.legend(lines, labels, loc='upper left')

plt.title('Multiple Types of Plots with Multiple Y-Axes')
plt.show()
```


Multiple Types of Plots with Multiple Y-Axes
Line Chart
Bar Chart
Scatter Plot
Line Chart (y1)
Bar Chart (y2)
Scatter Plot (y3)
X-axis
0
5
10
15
20
25
0
10
20
30
40
50
20.0
22.5
25.0
27.5
30.0
32.5
35.0
37.5
1
2
3
4
5

In [ ]: