In [2]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
```

Vertical bar charts

Create a vertical bar chart for the values by years.

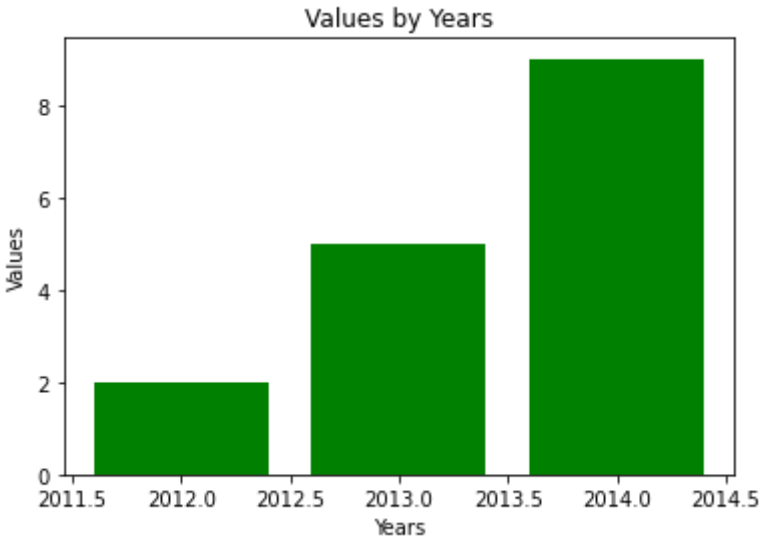
Its ok to have years split i.e, (2012.0, 2012.5...)

In [35]:
```
# Using
years = np.arange(2012, 2015)
values = [2, 5, 9]
# Create a vertical bar chart values by years.

# Code goes below:

```

In [36]:
```
plt.title("Values by Years")
plt.xlabel('Years')
plt.ylabel('Values')
plt.bar(years,values,color ="green");
```

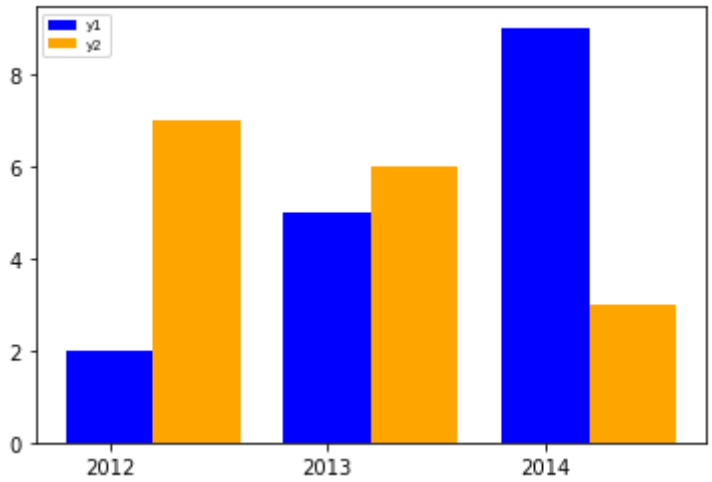


Create a multiseries bar chart with:

- category1_values: Color = Blue, No edgecolor, width = 0.4, align = center and a label of y1
- category2_values: Color = Orange, No edgecolor, width = 0.4, align = center and a label of y2

In [30]:
```
years = np.arange(2012, 2015)
category1_values = [2, 5, 9]
category2_values = [7, 6, 3]
# spacing bars out
xposCat1 = np.arange(len(years))
xposCat2 = xposCat1 + 0.4
```

In [33]:
```
plt.bar(xposCat1, category1_values,color = 'blue', edgecolor='none', width
plt.bar(xposCat2, category2_values,color = 'orange', edgecolor='none', widt
plt.legend(loc = "upper left",fontsize = 'x-small')
# replace x indexes by the given values
plt.xticks(ticks = xposCat1,labels = years);
```

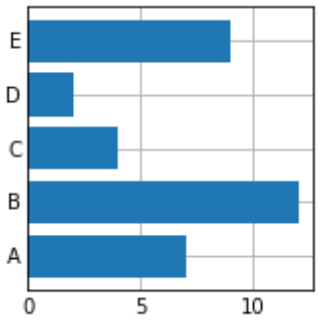


Create a horizontal bar chart for the given values:

- categories = ['A', 'B', 'C', 'D', 'E']
- values = [7, 12, 4, 2, 9]
- Label each as the letter from the categories parameter... i.e. A, B, C ...

In [79]:
```
categories = ['A', 'B', 'C', 'D', 'E']
values = [7, 12, 4, 2, 9]
y_pos = np.arange(len(categories))
```

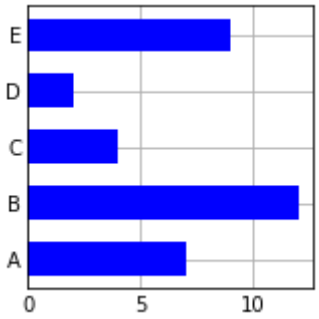
In [88]:
```
plt.barh(y_pos,values)
plt.yticks(y_pos,categories)
plt.show()
```



Create a horizontal bar chart with these values :

- Color = Blue, no edgecolor, height of 0.6, center aligned

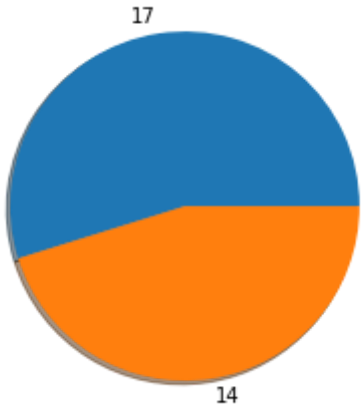
In [91]:
```
plt.barh(y_pos,values, color ="blue",edgecolor='none',height = 0.6,align ='
plt.yticks(y_pos,categories)
plt.show()
```



Create a pie chart

- For the given counts = [17, 14]

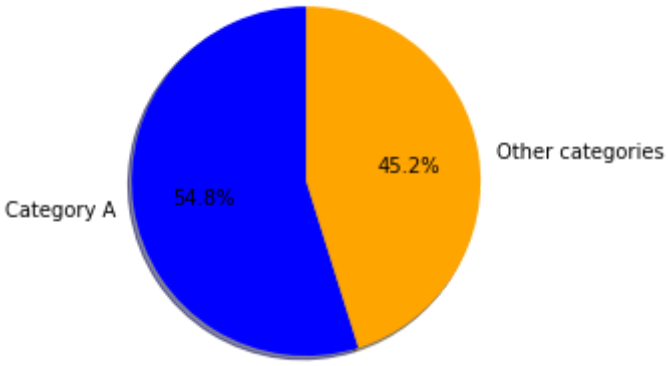
In [29]:
```
counts =  [17, 14]
lab = ["17","14"]
plt.pie(counts,shadow = True, labels= lab);
```



Create a pie chart for counts = [17, 14] using:

- Colors of blue and orange
- Labels of 'Category A' and 'Other categories'
- Set the angle at 90
- Set the percentage display format of 1.1

In [25]:
```
counts =  [17, 14]
colors = ['blue','orange']
lab = ["Category A","Other categories"]
plt.pie(counts,shadow = True, labels= lab, colors = colors,startangle = 90,
```

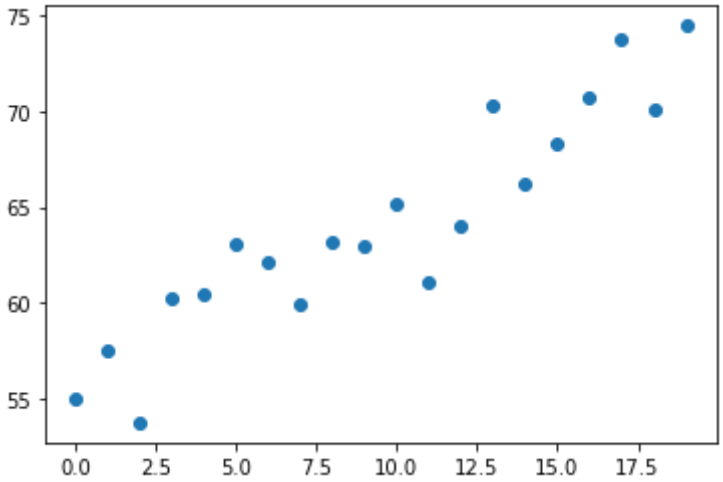


Create a scatter plot for the given data

In [18]:
```
x = range(20)
y = np.arange(50, 70) + (np.random.random(20) * 10. )
x
```

Out[18]:
```
range(0, 20)
```

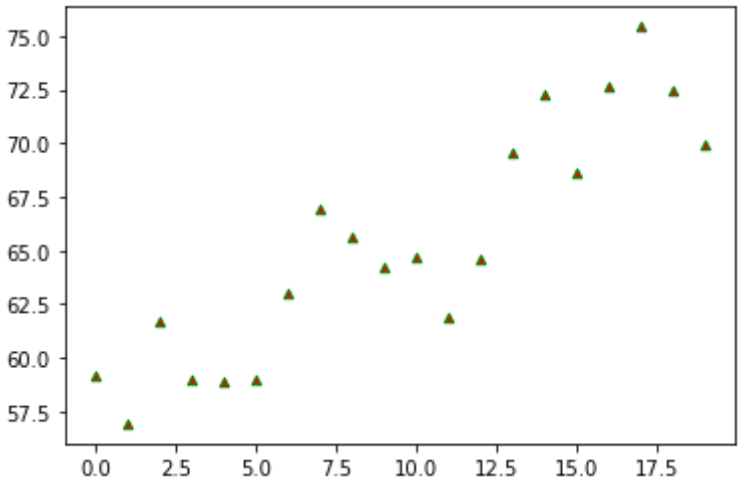
In [16]:
```
plt.scatter(x,y)
plt.show()
```



Create a scatter plot with the given data with:

- Color of red
- Marker
- `c` : Set the color of the markers.
- `s` : Set the size of the markers.
- `marker` : Set the marker style, e.g., circles, triangles, or squares.
- `edgecolor` : Set the color of the lines on the edges of the markers.

In [24]:
```
x = np.arange(20)
y = np.arange(50, 70) + (np.random.random(20) * 10.)
plt.scatter(x,y,marker = '^', c ='red', edgecolor = 'green',s = 20, )
plt.show()
```



END

In [ ]: