

Le module matplotlib permet de générer des graphiques depuis Python. Il est l'outil complémentaire de NumPy, scipy ou pandas (que l'on verra juste après) lorsqu'on veut faire de l'analyse de données.

Il est très facile de trouver sur internet des exemples de représentation de données, de changement de couleurs... (<https://matplotlib.org/gallery/index.html> (<https://matplotlib.org/gallery/index.html>)) <http://www.python-simple.com/python-matplotlib/matplotlib-intro.php> (<http://www.python-simple.com/python-matplotlib/matplotlib-intro.php>)

## Réaliser des graphiques simples

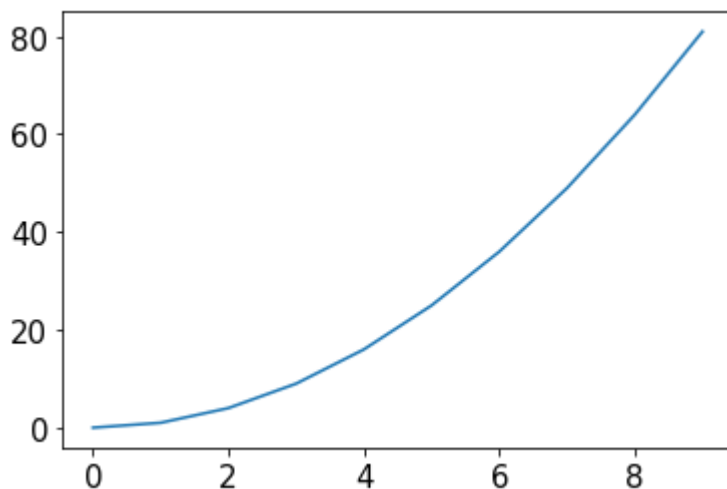
Commençons par étudier un cas simple, comme tracer la courbe d'une fonction.

La variable fig correspond à un conteneur qui contient tous les objets (axes, labels, données, etc). Les axes contiendront par la suite les données du graphe.

In [4]:

```
import matplotlib.pyplot as plt
import numpy as np
fig = plt.figure()
ax = plt.axes()

x = np.array([0,1,2,3,4,5,6,7,8,9])
ax.plot(x, x**2);
```



In [3]:

```
# Chanegr la taille de police par défaut
plt.rcParams.update({'font.size': 15})

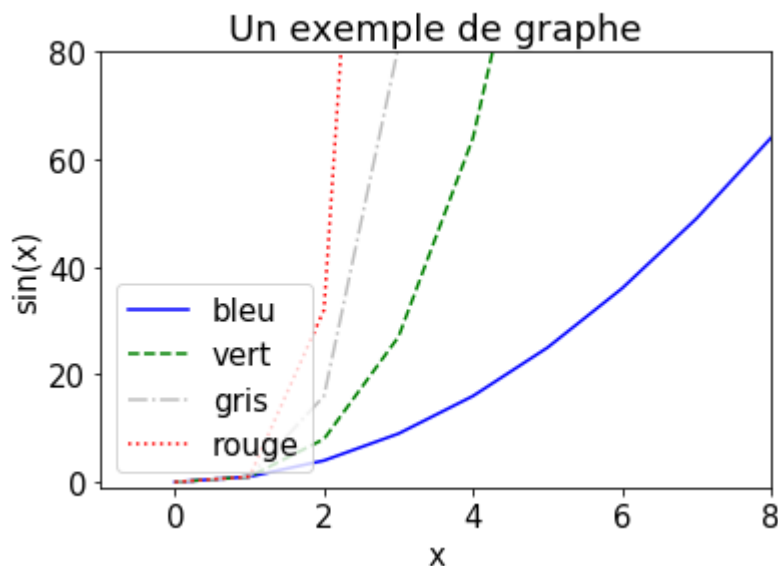
fig = plt.figure()
ax = plt.axes()
# Couleur spécifiée par son nom, ligne solide
plt.plot(x, x**2, color='blue', linestyle='solid', label='bleu')
# Nom court pour la couleur, ligne avec des traits
plt.plot(x, x**3, color='g', linestyle='dashed', label='vert')
# Valeur de gris entre 0 et 1, des traits et des points
plt.plot(x, x**4, color='0.75', linestyle='dashdot', label='gris')
# Couleur spécifié en RGB, avec des points
plt.plot(x, x**5, color='#FF0000', linestyle='dotted', label='rouge')

# Les limites des axes, essayez aussi les arguments 'tight' et 'equal'
# pour voir leur effet
plt.axis([-1, 8, -1, 80]);

# Les labels
plt.title("Un exemple de graphe")

# La légende est générée à partir de l'argument label de la fonction
# plot. L'argument loc spécifie le placement de la légende
plt.legend(loc='lower left');

# Titres des axes
ax = ax.set(xlabel='x', ylabel='sin(x)')
```



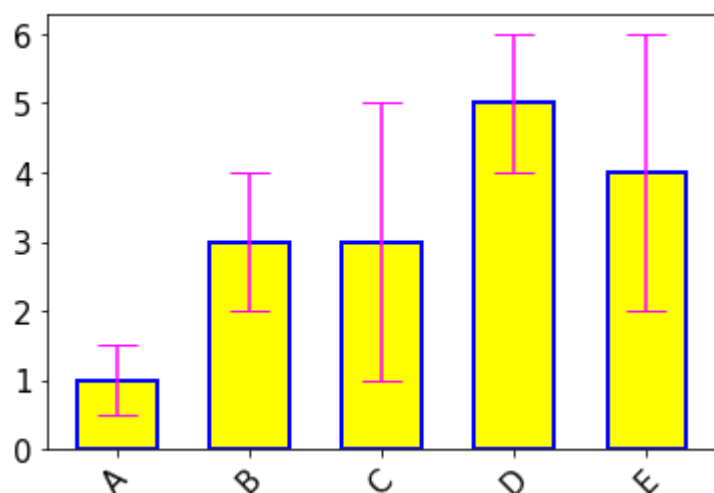
In [5]:

```
# Barplot
```

```
plt.bar(range(5), [1, 3, 3, 5, 4], width = 0.6, color = 'yellow',  
        edgecolor = 'blue', linewidth = 2, yerr = [0.5, 1, 2, 1, 2],  
        ecolor = 'magenta', capsize = 10)  
plt.xticks(range(5), ['A', 'B', 'C', 'D', 'E'],  
           rotation = 45)
```

Out[5]:

```
([<matplotlib.axis.XTick at 0x275b28b0a90>,  
 <matplotlib.axis.XTick at 0x275b27ac978>,  
 <matplotlib.axis.XTick at 0x275b28a1198>,  
 <matplotlib.axis.XTick at 0x275b2909550>,  
 <matplotlib.axis.XTick at 0x275b29099e8>],  
 <a list of 5 Text xticklabel objects>)
```

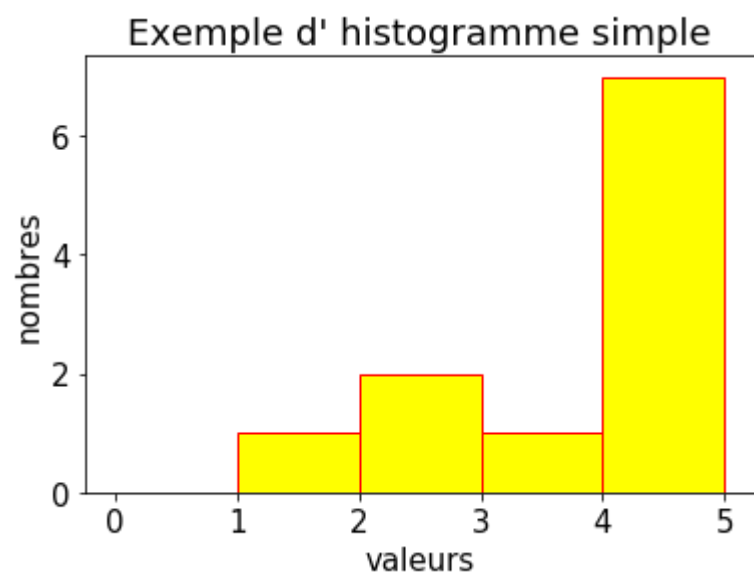


In [7]:

```
# histogramme  
  
x = [1, 2, 2, 3, 4, 4, 4, 4, 4, 5, 5]  
plt.hist(x, range = (0, 5), bins = 5, color = 'yellow',  
         edgecolor = 'red')  
plt.xlabel('valeurs')  
plt.ylabel('nombres')  
plt.title('Exemple d\' histogramme simple')
```

Out[7]:

```
Text(0.5, 1.0, "Exemple d' histogramme simple")
```



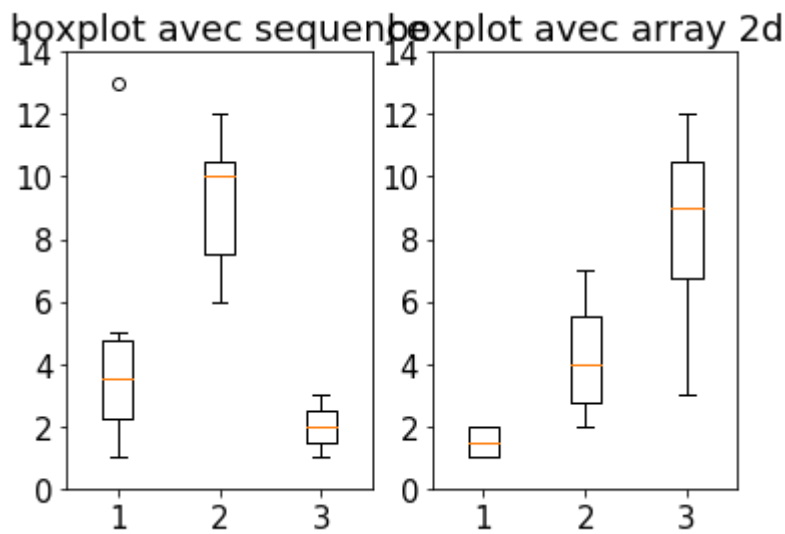
In [8]:

```
# Boxplot
import numpy
plt.subplot(121)
plt.boxplot([[1, 2, 3, 4, 5, 13], [6, 7, 8, 10, 10, 11, 12], [1, 2, 3]])
plt.ylim(0, 14)
plt.title('boxplot avec sequence')

plt.subplot(122)
plt.boxplot(numpy.array([[1, 2, 3], [2, 7, 8], [1, 3, 10], [2, 5, 12]]))
plt.ylim(0, 14)
plt.title('boxplot avec array 2d')
```

Out[8]:

Text(0.5, 1.0, 'boxplot avec array 2d')



In [12]:

```
plt.figure(figsize = (8, 8))  
x = [1, 2, 3, 4, 10]  
plt.pie(x, labels = ['A', 'B', 'C', 'D', 'E'])  
plt.legend()
```

Out[12]:

<matplotlib.legend.Legend at 0x275b2c7f7f0>

