**Matplotlib.**

Установка:

Чтобы установить библиотеку Matplotlib введите в терминале команды:

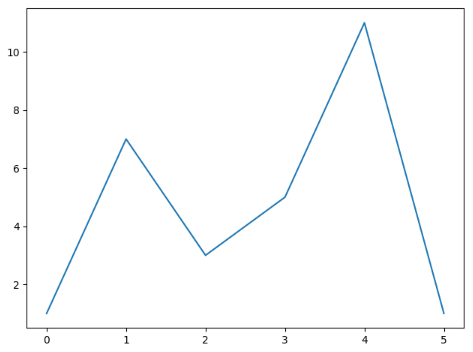
python -m pip install -U pip

python -m pip install -U matplotlib

1. **Построение графиков.**
   1. Построение одного графика.

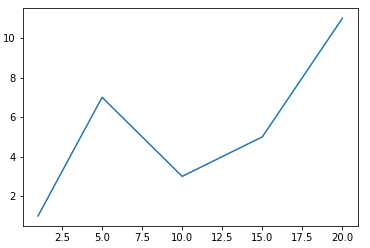
**import** matplotlib.pyplot **as** plt  
plt.plot([1, 7, 3, 5, 11, 1])  
plt.show()

Если в качестве параметра функции plot() передать список, то значения из этого списка будут отложены по оси ординат (ось y), а по оси абсцисс (ось x) будут отложены индексы элементов массива:



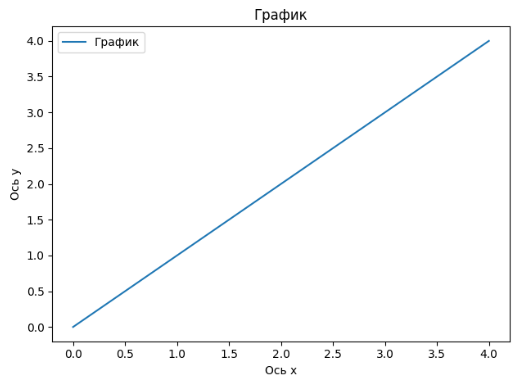
Для того, чтобы задать значения по осям x и y необходимо в plot() передать два списка:

plt.plot([1, 5, 10, 15, 20], [1, 7, 3, 5, 11])



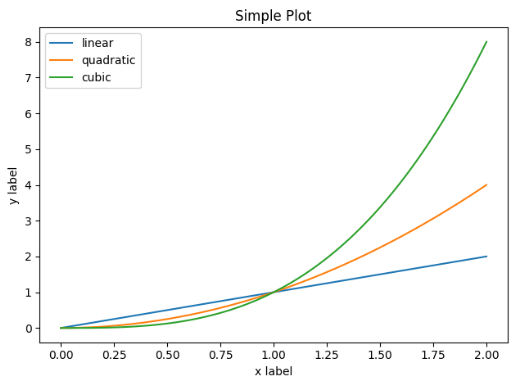
Оформление графика:

**import** matplotlib.pyplot **as** plt  
**import** numpy **as** np  
n = [0, 1, 2, 3, 4]  
fig, ax = plt.subplots() *# Получение фигуры и оси*ax.plot(n, n, label=**"График"**) *# Начертить график*ax.set\_xlabel(**"Ось x"**) *# Подписи осей*ax.set\_ylabel(**"Ось y"**)  
ax.set\_title(**"График"**) *# Заголовок фигуры*ax.legend() *# Показать легенду*plt.show()



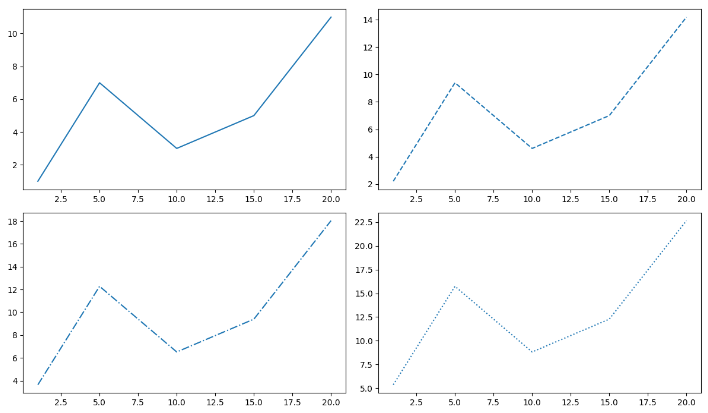
* 1. Построение нескольких графиков на одной фигуре

**import** matplotlib.pyplot **as** plt  
**import** numpy **as** np  
n = np.linspace(0, 2, 100) *# 100 чисел в интервале [0, 2]*fig, ax = plt.subplots()  
ax.plot(n, n, label=**"linear"**) *# 1 график*ax.plot(n, n\*\*2, label=**"quadratic"**) *# 2 график*ax.plot(n, n\*\*3, label=**"cubic"**) *# 3 график*ax.set\_xlabel(**"x label"**)  
ax.set\_ylabel(**"y label"**)  
ax.set\_title(**"Simple Plot"**)  
ax.legend()  
plt.show()



* 1. Размещение графиков на разных полях

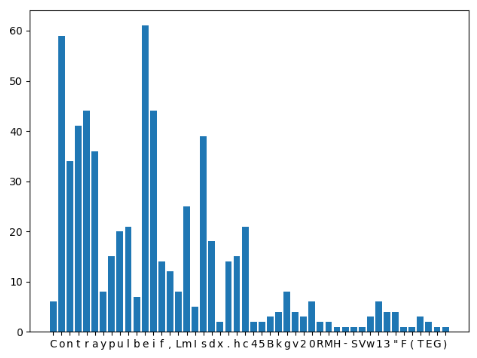
**import** matplotlib.pyplot **as** plt  
*# Исходный набор данных*x = [1, 5, 10, 15, 20]  
y1 = [1, 7, 3, 5, 11]  
y2 = [i\*1.2 + 1 **for** i **in** y1]  
y3 = [i\*1.2 + 1 **for** i **in** y2]  
y4 = [i\*1.2 + 1 **for** i **in** y3]  
*# Настройка размеров подложки*plt.figure(figsize=(12, 7))  
*# Вывод графиков*plt.subplot(2, 2, 1)  
plt.plot(x, y1, **'-'**)  
plt.subplot(2, 2, 2)  
plt.plot(x, y2, **'--'**)  
plt.subplot(2, 2, 3)  
plt.plot(x, y3, **'-.'**)  
plt.subplot(2, 2, 4)  
plt.plot(x, y4, **':'**)  
plt.show()



1. **Построение гистограммы.**

Напишем программу, которая подсчитывает количество различных символов и выводит гистограмму с их частотами.

**import** matplotlib.pyplot **as** plt  
d = {}  
**with** open(**"data.txt"**, **"r"**) **as** f:  
 **for** line **in** f:  
 **for** c **in** line:  
 **if** c != **' ' and** c != **'\n'**:  
 **if** c **not in** d:  
 d.update({c : 1})  
 **else**:  
 d[c] += 1  
  
fig, ax = plt.subplots()  
plt.bar(d.keys(), d.values())  
plt.show()



1. **Построение 3D.**

Построение треугольника в 3D фигуре:

**from** mpl\_toolkits.mplot3d **import** Axes3D  
**import** matplotlib.pyplot **as** plt  
**from** mpl\_toolkits.mplot3d.art3d **import** Poly3DCollection  
fig = plt.figure()  
ax = Axes3D(fig)  
triangle = [((0.8,0.5,0), (0.5,0,0), (0.5, 0.5, 1))]  
ax.add\_collection(Poly3DCollection(triangle))  
plt.show()

