

# Matplotlib

---

# Модуль Matplotlib

---

Matplotlib - библиотека на языке программирования Python для визуализации данных двумерной (2D) графикой

Пакет поддерживает многие виды графиков и диаграмм:

- Графики (line plot)
- Диаграммы разброса (scatter plot)
- Столбчатые диаграммы (bar chart) и гистограммы (histogram)
- Круговые диаграммы (pie chart)
- Ствол-лист диаграммы (stem plot)
- Контурные графики (contour plot)
- Поля градиентов (quiver)
- Спектральные диаграммы (spectrogram)

# Установка

---

Для того, чтобы установить модуль numpy, необходимо открыть консоль

Win + R □ cmd □ Enter

Далее, в консоли необходимо прописать `pip install matplotlib`

Установка завершена

# Импорт модуля

---

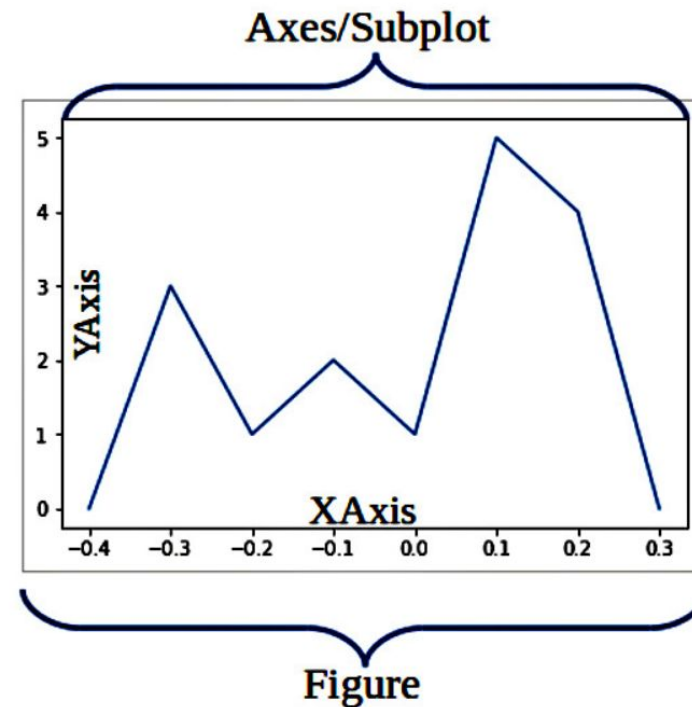
В первой строке прописываем `import matplotlib.pyplot as plt`

- `Import` – подключение модуля
- `matplotlib` – модуль
- `As plt` – используется для сокращенного пользования модулем

# Matplotlib

*Figure* - это контейнер самого верхнего уровня, та область на которой все нарисовано. Таких областей может быть несколько, каждая из которых может содержать несколько контейнеров *Axes*.

*Axes* - это та область на которой чаще всего и отражаются графики (данные в виде графиков), а так же все вспомогательные атрибуты (линии сетки, метки, указатели и т.д.).



# plt.plot()

---

Это обычный график роста

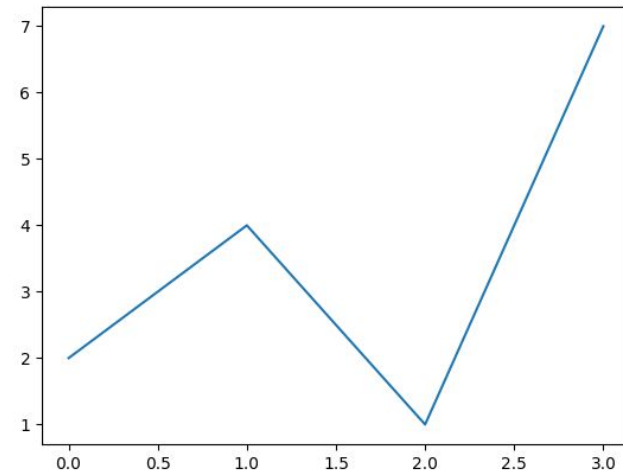
На таком графике, как правило, показывают динамику развития чего либо относительно второго параметра (оси: OX и OY)

```
import matplotlib.pyplot as plt
```

```
arr1=[2,4,1,7]
```

```
plt.plot(arr1)
```

```
plt.show()
```



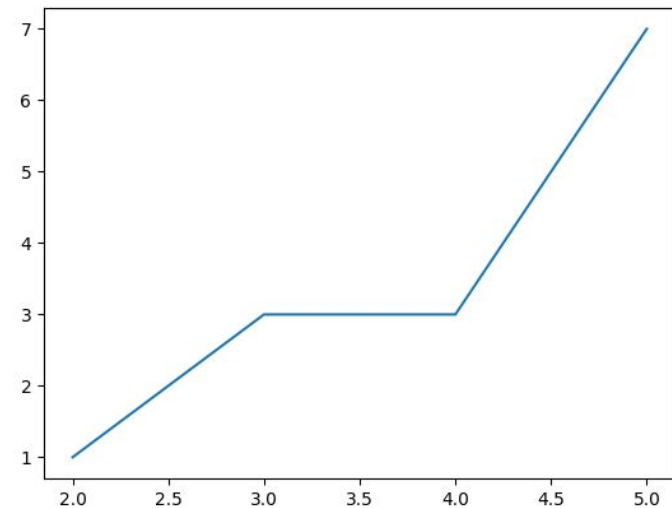
# plt.plot()

---

В качестве параметров в функцию plot можно передавать два списка – первый по оси X, второй по оси Y

Один из списков обязательно должен быть числовым  
plot также работает и с массивами numpy и Series мод

```
import numpy as np
import matplotlib.pyplot as plt
arr1=[2,3,4,5]
arr2=[1,3,3,7]
plt.plot(arr1,arr2)
plt.show()
```



# plt.plot()

---

```
import numpy as np
```

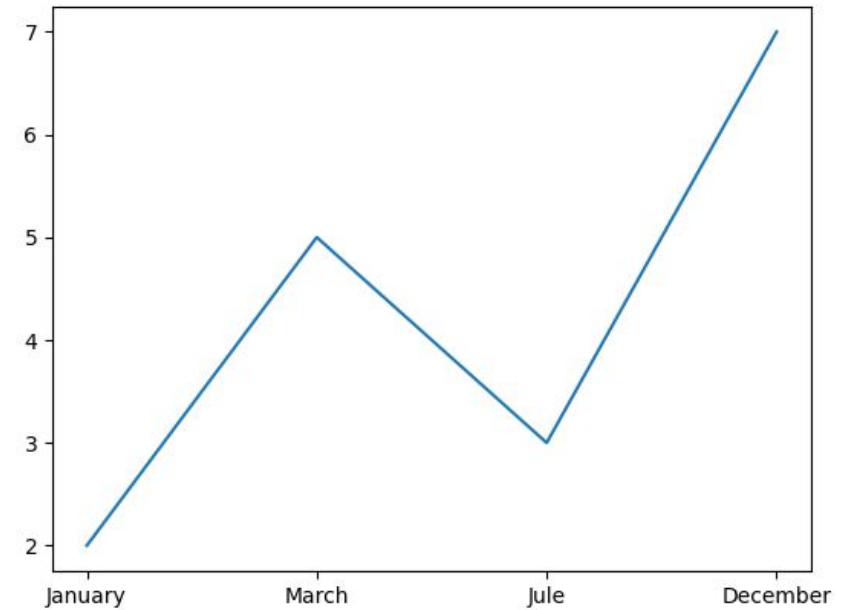
```
import matplotlib.pyplot as plt
```

```
arr1=["January",'March','Jule','December']
```

```
arr2=[2,5,3,7]
```

```
plt.plot(arr1,arr2)
```

```
plt.show()
```





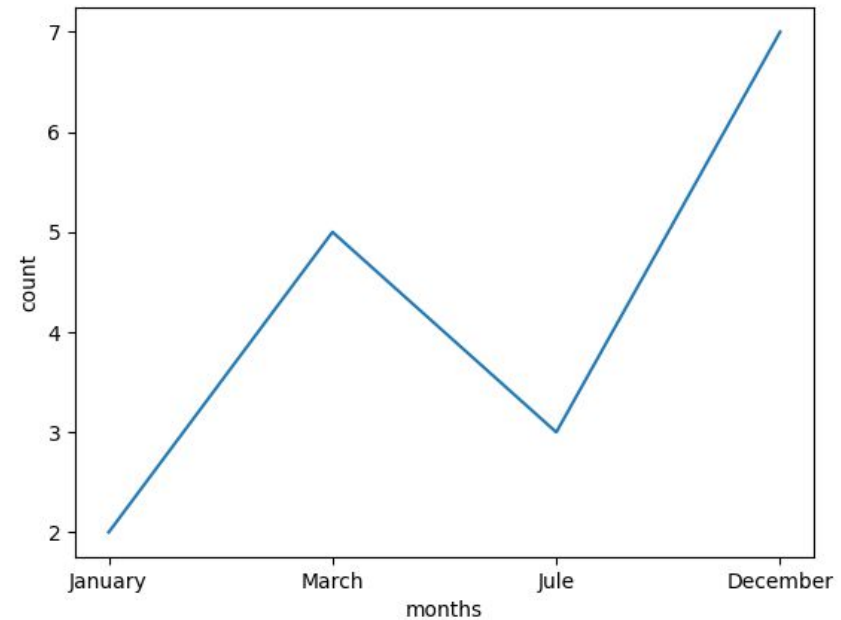
# Подписи осей

---

Подписи осей можно изменить при помощи функций `xlabel` и `ylabel`

```
plt.xlabel('months')
```

```
plt.ylabel('count')
```

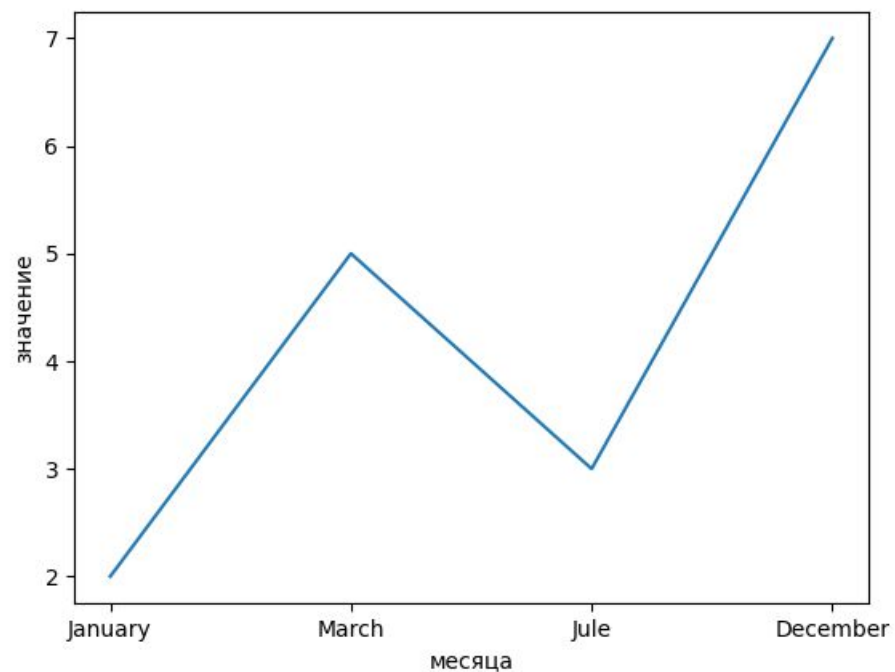


# Подписи осей

---

```
plt.xlabel(u'месяца')
```

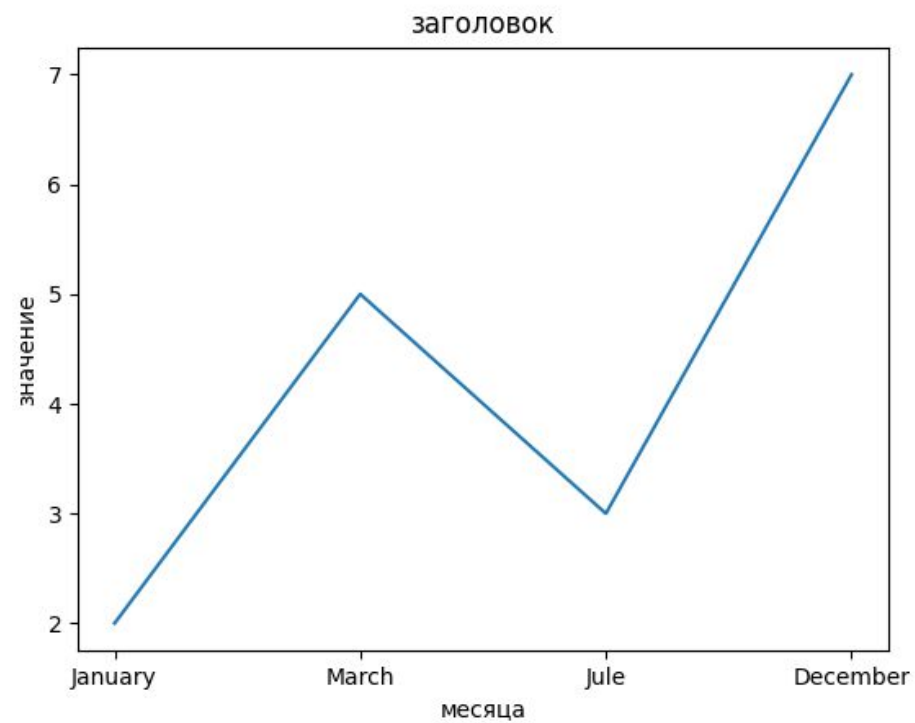
```
plt.ylabel(u'значение')
```



# Подпись заголовок

---

```
plt.title(u'заголовок')
```



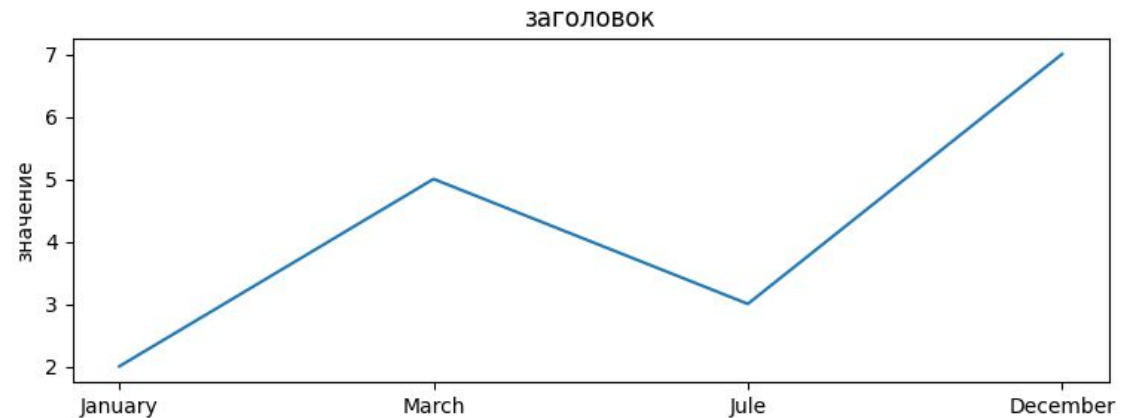
# figure

---

При использовании функции `plot` `figure` создаётся автоматически. Если создавать вручную, можно указать `figsize` – размеры `figure` в дюймах.

Значение по умолчанию – 6.4 и 4.8

```
plt.figure(figsize=(9, 3))
```



# axis()

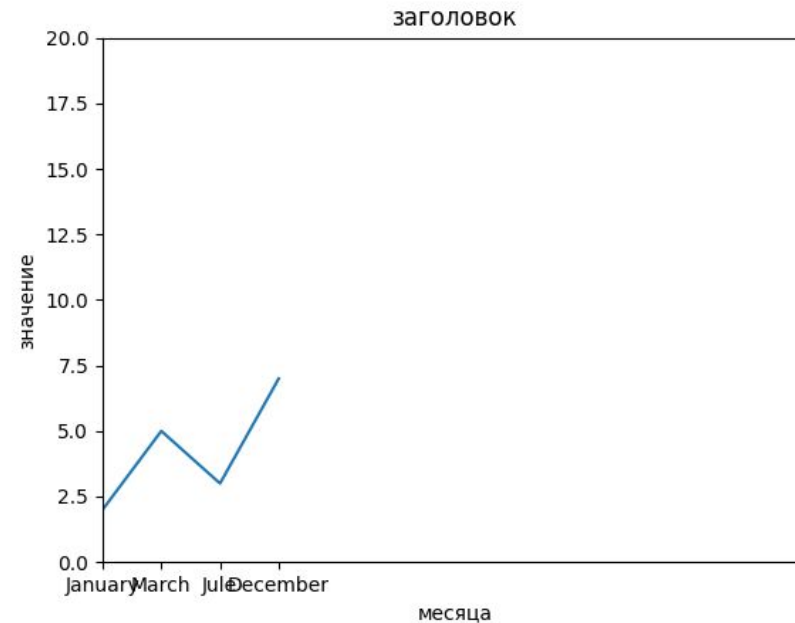
---

Если вас не устраивает, как выглядит график, вы можете самостоятельно задать значения на осях, то есть указать деления на осях X и Y

```
plt.axis([0, 12, 0, 20])
```

0 – 12 деления на оси X

0 – 20 деления на оси Y



# Форматирование графиков

---

После значений есть возможность передать строку, которая должна содержать формат графика.

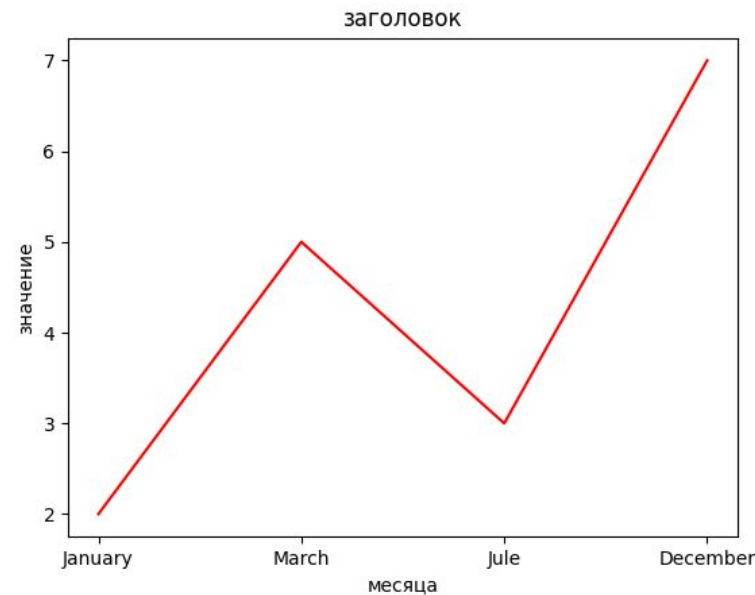
Пример использования:

```
plt.plot(arr1, arr2, 'r*--')
```

r цвет

+ тип соединения

- тип линий



# Возможные значения

---

'b', 'g', 'r', 'c', 'm', 'y', 'k', 'w' – цвета

[https://matplotlib.org/3.1.1/api/markers\\_api.html?highlight=marker#module-matplotlib.markers](https://matplotlib.org/3.1.1/api/markers_api.html?highlight=marker#module-matplotlib.markers) – таблица с типами маркеров

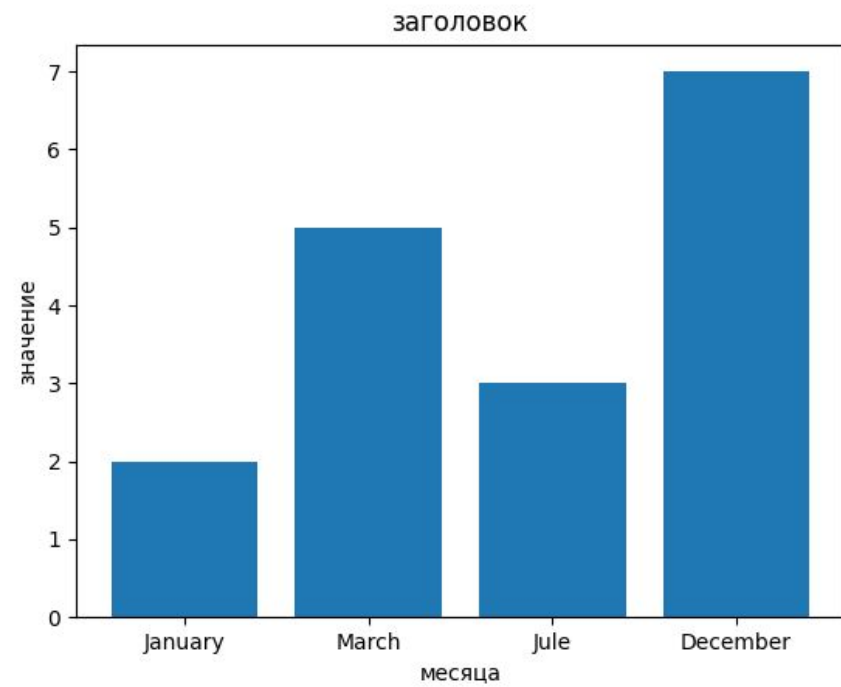
'-', '--', '-.', ':-' - линия

Параметры записываются в любом порядке

# bar()

---

`plt.bar(arr1,arr2)`

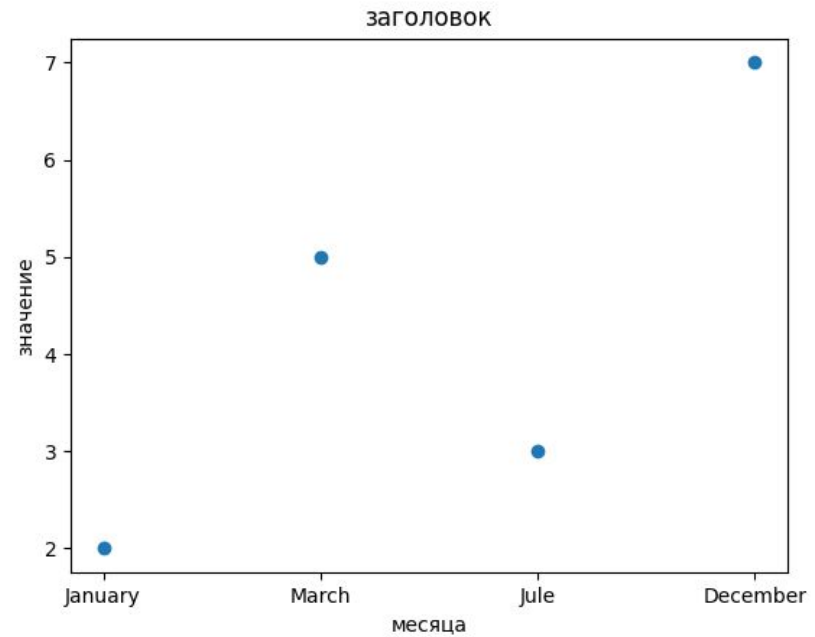




# scatter()

---

```
plt.scatter(arr1,arr2)
```



```
names = ['group_a', 'group_b', 'group_c']  
values = [1, 10, 100]
```

---

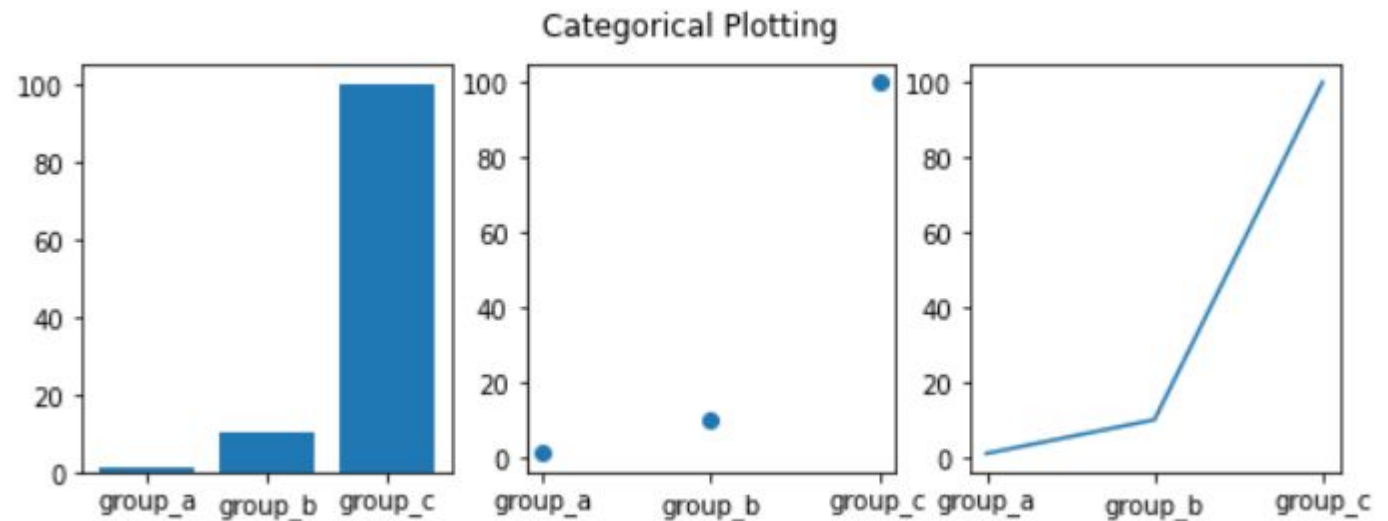
```
plt.figure(figsize=(9, 3))
```

```
plt.subplot(131)  
plt.bar(names, values)
```

```
plt.subplot(132)  
plt.scatter(names, values)
```

```
plt.subplot(133)  
plt.plot(names, values)
```

```
plt.suptitle('Categorical Plotting')  
plt.show()
```



# Добавление графиков

---

Добавление нескольких графиков в один выполняется при помощи команды `subplot`

В `subplot` нужно вписать либо 3 параметра, либо трехзначное число.

Трехзначное число делится на отдельные цифры и представляют собой отдельные измерения для расположения графика

Например:

```
plt.subplot(132)
```

1 – это количество строк

3 – это количество столбцов

2 – это индекс размещения

# Задачи

---

1. Построить график функции  $y = x^{**} 2$  на 10 значениях
2. Найти информацию о суточной температуре за последнюю неделю и построить график
3. Построить график цен на бензин за последний год ежемесячно
4. Построить графики стоимости доллара, евро и рубля за последний год (subplot)
5. Построить график ваших оценок(subplot)