

# 1\_basic\_plot\_exercise

May 20, 2019

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

plt.rcParams["figure.figsize"] = (10,5)

x = range(100)
y = [value**2 for value in x]

plt.plot(x,y)
plt.savefig('test.png')
```

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

plt.rcParams["figure.figsize"] = (10,5)

x = range(100)
y = [value**2 for value in x]

plt.plot(x,y)
plt.savefig('test.png')
```

```
In [ ]: import numpy as np

X_1 = range(100)
Y_1 = [np.cos(value) for value in X_1]

X_2 = range(100)
Y_2 = [np.sin(value) for value in X_2]

plt.plot(X_1, Y_1)
plt.plot(X_2, Y_2)

plt.show()
```

```
In [ ]: fig = plt.figure() # figure
fig.set_size_inches(10,5) #
```

```

ax_1 = fig.add_subplot(1,2,1) # plot
ax_2 = fig.add_subplot(1,2,2) # plot

ax_1.plot(X_1, Y_1, c="b") # plot
ax_2.plot(X_2, Y_2, c="g") # plot
plt.show() # show & flush

```

```
In [ ]: plt.show()
```

### 0.0.1 Set color

- [http://matplotlib.org/2.0.2/api/colors\\_api.html](http://matplotlib.org/2.0.2/api/colors_api.html)

```

In [ ]: X_1 = range(100)
        Y_1 = [value for value in X_1]

        X_2 = range(100)
        Y_2 = [value + 50 for value in X_2]

        plt.plot(X_1, Y_1, color="#000000")
        plt.plot(X_2, Y_2, c="c")

        plt.show()

```

### 0.0.2 Set linestyle

- [https://matplotlib.org/examples/lines\\_bars\\_and\\_markers/linestyles.html](https://matplotlib.org/examples/lines_bars_and_markers/linestyles.html)

```

In [ ]: plt.plot(X_1, Y_1, c="b", linestyle="dashed")
        plt.plot(X_2, Y_2, c="r", ls="dotted")

        plt.show()

In [ ]: plt.plot(X_1, Y_1, color="b", linestyle="dashed")
        plt.plot(X_2, Y_2, color="r", linestyle="dotted")

        plt.title("Two lines")
        plt.show()

In [ ]: plt.plot(X_1, Y_1, color="b", linestyle="dashed")
        plt.plot(X_2, Y_2, color="r", linestyle="dotted")

        plt.title('$y = \frac{ax + b}{test}$')
        plt.show()

In [ ]: plt.plot(X_1, Y_1, color="b", linestyle="dashed")
        plt.plot(X_2, Y_2, color="r", linestyle="dotted")

        plt.title('$y = ax+b$')

```

```

plt.xlabel('$x_{line}$')
plt.ylabel('$y_{line}$')

plt.show()

In [ ]: plt.plot(X_1, Y_1, color="b", linestyle="dashed")
plt.plot(X_2, Y_2, color="r", linestyle="dotted")

plt.text(50, 70, "Line_1")
plt.annotate(
    'line_2', xy=(50, 100), xytext=(30, 110),
    arrowprops=dict(facecolor='black', shrink=0.1))

plt.title('$y = ax+b$')
plt.xlabel('$x_{line}$')
plt.ylabel('$y_{line}$')

plt.show()

In [ ]: plt.plot(X_1, Y_1, color="b", linestyle="dashed", label='line_1')
plt.plot(X_2, Y_2, color="r", linestyle="dotted", label='line_2')
plt.legend(shadow=True, fancybox=False, loc="upper right")

plt.title('$y = ax+b$')
plt.xlabel('$x_{line}$')
plt.ylabel('$y_{line}$')

plt.show()

```

### 0.0.3 Scatter

```

In [ ]: data_1 = np.random.rand(512, 2)
data_2 = np.random.rand(512, 2)

In [ ]: plt.scatter(data_1[:,0], data_1[:,1], c="b", marker="x")
plt.scatter(data_2[:,0], data_2[:,1], c="r", marker="o")

plt.show()

In [ ]: N = 50
x = np.random.rand(N)
y = np.random.rand(N)
colors = np.random.rand(N)
area = np.pi * (15 * np.random.rand(N))**2
plt.scatter(x, y, s=area, c=colors, alpha=0.5)
plt.show()

```

#### 0.0.4 Bar chart

```
In [ ]: data = [[5., 25., 50., 20.],
                [4., 23., 51., 17],
                [6., 22., 52., 19]]

X = np.arange(0,8,2)

In [ ]: plt.bar(X + 0.00, data[0], color = 'b', width = 0.50)
plt.bar(X + 0.50, data[1], color = 'g', width = 0.50)
plt.bar(X + 1.0, data[2], color = 'r', width = 0.50)
plt.xticks(X+0.50, ("A","B","C", "D"))
plt.show()

In [ ]: data = np.array([[5., 25., 50., 20.],
                        [4., 23., 51., 17],
                        [6., 22., 52., 19]])

color_list = ['b', 'g', 'r']
data_label = ["A","B","C"]
X = np.arange(data.shape[1])
X

In [ ]: data = np.array([[5., 5., 5., 5.],
                        [4., 23., 51., 17],
                        [6., 22., 52., 19]])

for i in range(3):
    plt.bar(X, data[i], bottom = np.sum(data[:i], axis=0),
            color = color_list[i], label=data_label[i])
plt.legend()
plt.show()

In [ ]: A = [5., 30., 45., 22.]
B = [5, 25, 50, 20]

X = range(4)

plt.bar(X, A, color = 'b')
plt.bar(X, B, color = 'r', bottom = 60)
plt.show()

In [ ]: women_pop = np.array([5, 30, 45, 22])
men_pop = np.array([5, 25, 50, 20])
X = np.arange(4)

plt.barh(X, women_pop, color = 'r')
plt.barh(X, -men_pop, color = 'b')
plt.show()
```

```
In [ ]: X = np.arange(100)
        plt.hist(X,bins=5)
        plt.show()
```

```
In [ ]: data = np.random.randn(100,5)
        plt.boxplot(data)
        plt.show()
```

```
In [ ]:
```