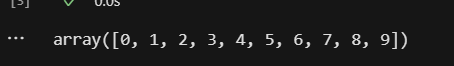
import matplotlib.pyplot as plt

import numpy as np

data = np.arange(10)

data



plt.plot(data)

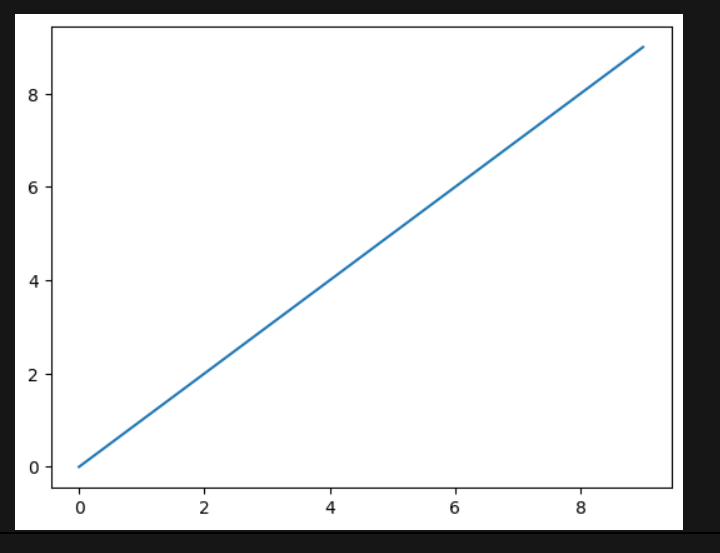
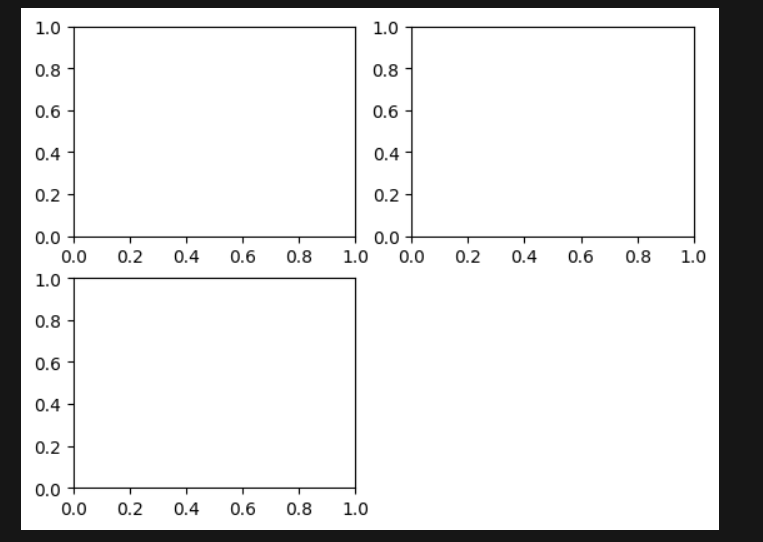


fig = plt.figure()

ax1 = fig.add\_subplot(2, 2, 1)

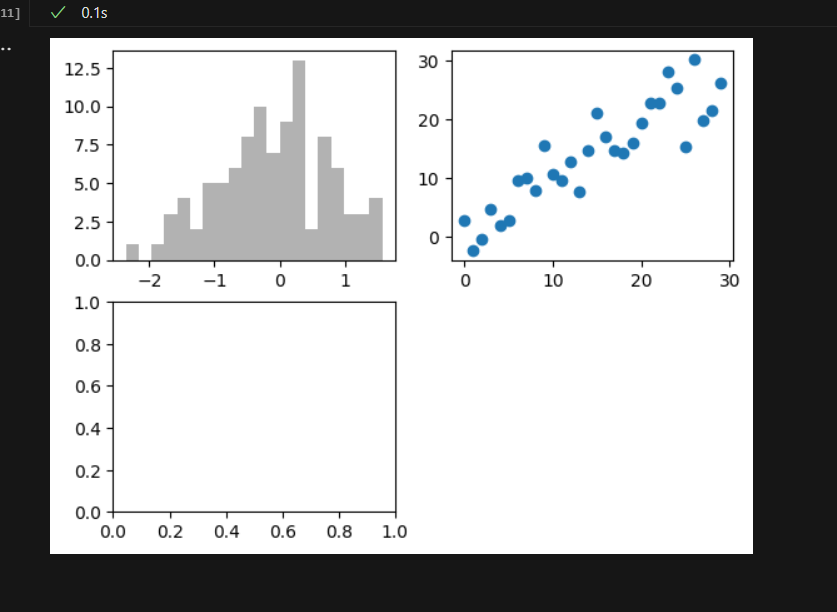
ax2 = fig.add\_subplot(2, 2, 2)

ax3 = fig.add\_subplot(2, 2, 3)



\_ = ax1.hist(np.random.randn(100), *bins*=20, *color*='k', *alpha*=0.3)

ax2.scatter(np.arange(30), np.arange(30) + 3 \* np.random.randn(30))



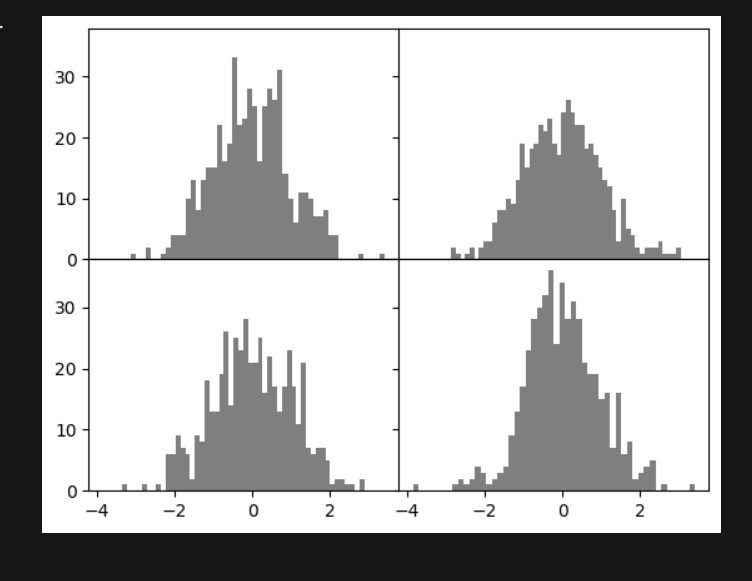
fig, axes = plt.subplots(2, 2, *sharex*=True, *sharey*=True)

for i in range(2):

for j in range(2):

axes[i, j].hist(np.random.randn(500), *bins*=50, *color*='k', *alpha*=0.5)

plt.subplots\_adjust(*wspace*=0, *hspace*=0)



data = np.random.randn(30).cumsum()

plt.plot(data, 'k--', *label*='Default')

plt.plot(data, 'k-', *drawstyle*='steps-post', *label*='steps-post')

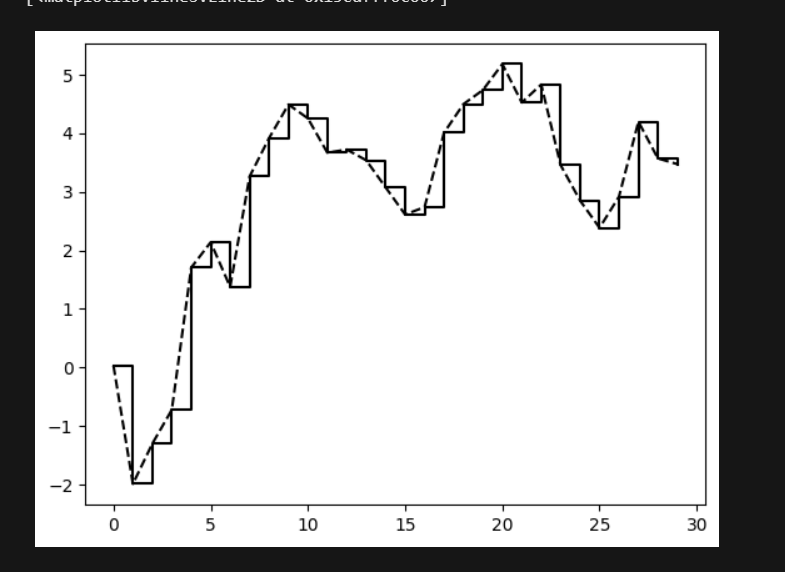
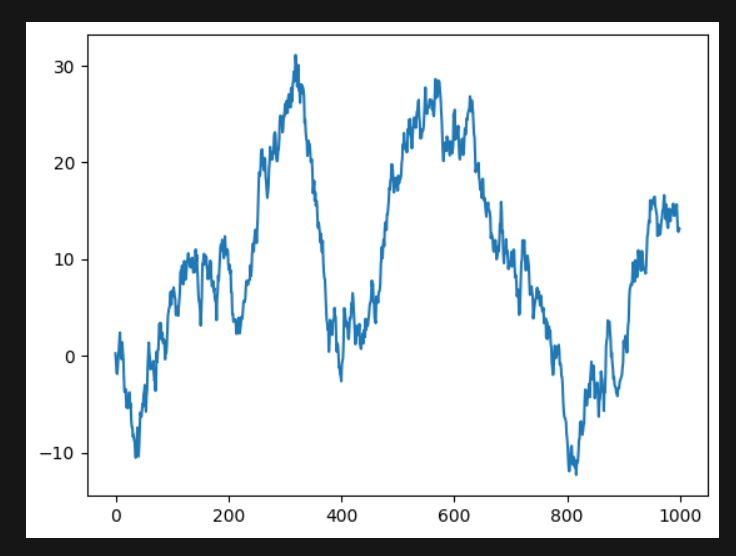


fig = plt.figure()

ax = fig.add\_subplot(1, 1, 1)

ax.plot(np.random.randn(1000).cumsum())



from numpy.random import randn

fig = plt.figure(); ax = fig.add\_subplot(1, 1, 1)

ax.plot(randn(1000).cumsum(), 'k', *label*='one')

ax.plot(randn(1000).cumsum(), 'k--', *label*='two')

ax.plot(randn(1000).cumsum(), 'k.', *label*='three')

ax.legend(*loc*='best')

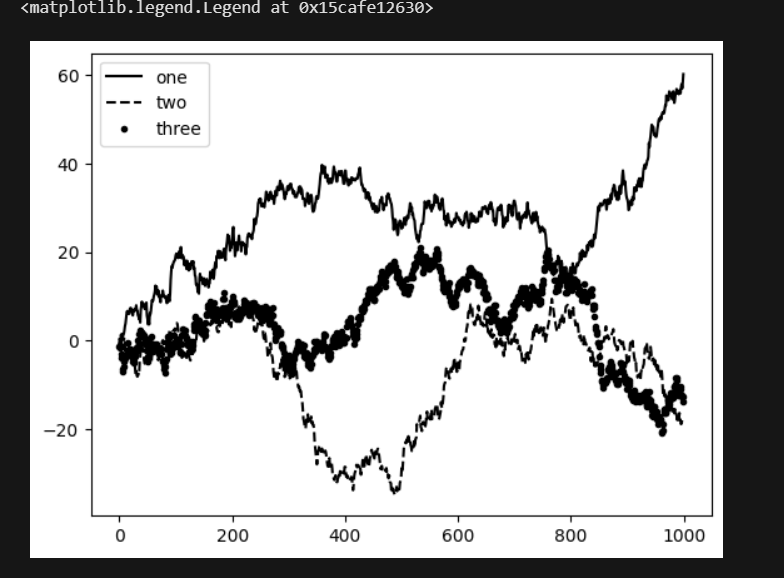


fig = plt.figure()

ax = fig.add\_subplot(1, 1, 1)

rect = plt.Rectangle((0.2, 0.75), 0.4, 0.15, *color*='k', *alpha*=0.3)

circ = plt.Circle((0.7, 0.2), 0.15, *color*='b', *alpha*=0.3)

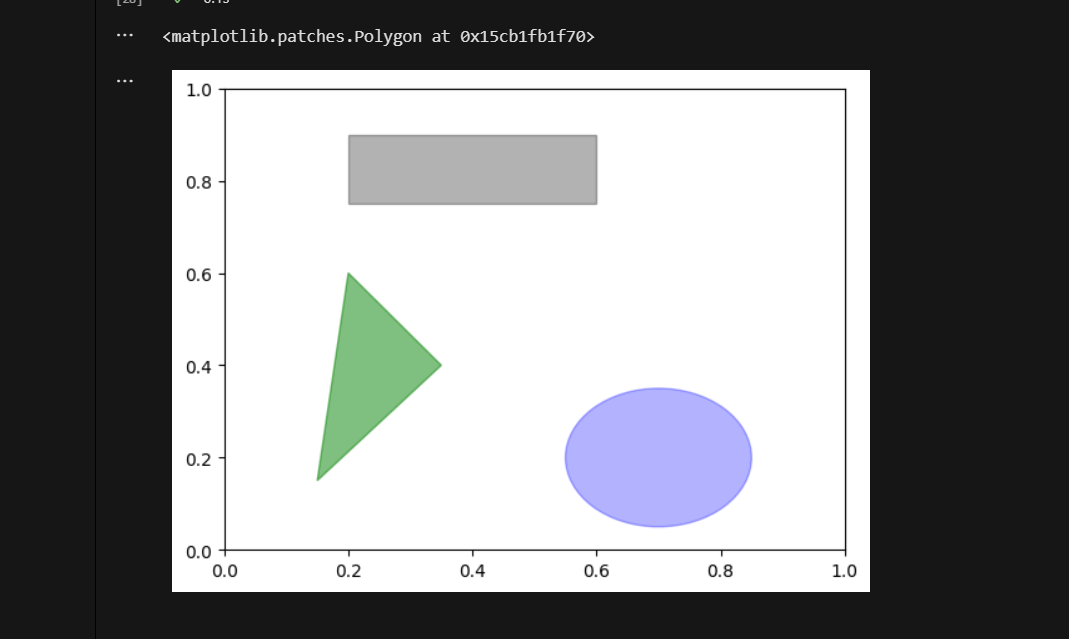
pgon = plt.Polygon([[0.15, 0.15], [0.35, 0.4], [0.2, 0.6]],

*color*='g', *alpha*=0.5)

ax.add\_patch(rect)

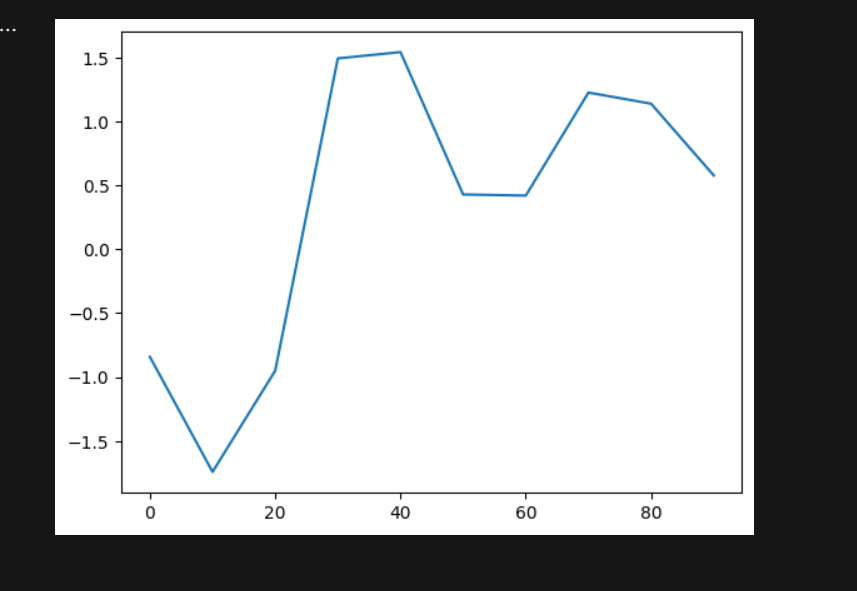
ax.add\_patch(circ)

ax.add\_patch(pgon)



s = pd.Series(np.random.randn(10).cumsum(), *index*=np.arange(0, 100, 10))

s.plot()

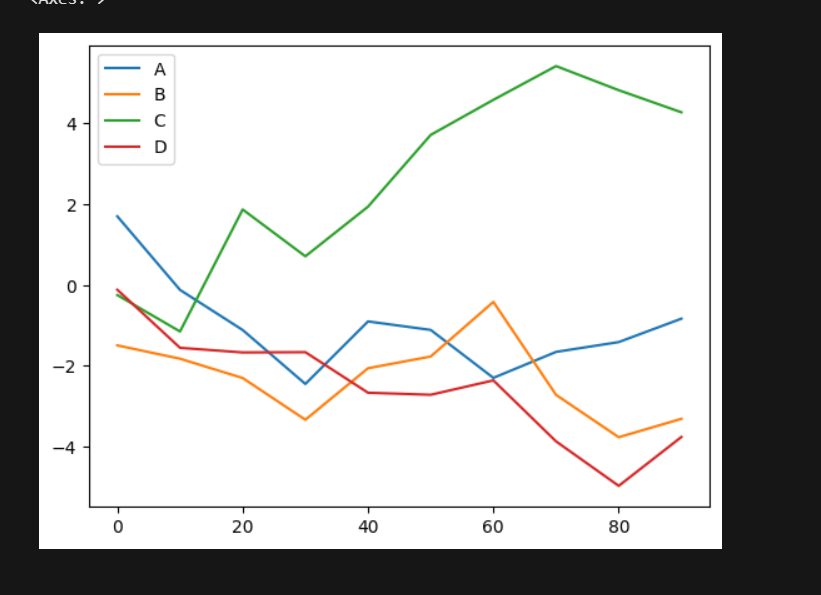


df = pd.DataFrame(np.random.randn(10, 4).cumsum(0),

*columns*=['A', 'B', 'C', 'D'],

*index*=np.arange(0, 100, 10))

df.plot()

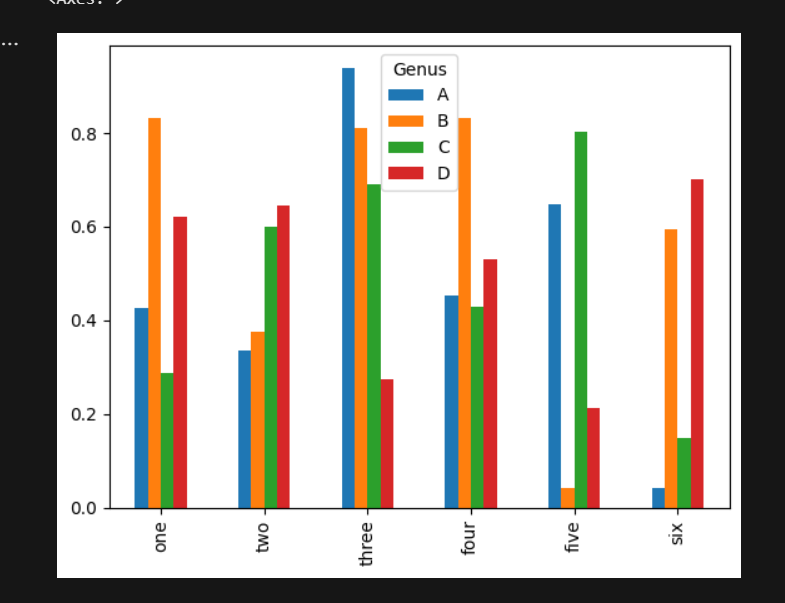


df = pd.DataFrame(np.random.rand(6, 4),

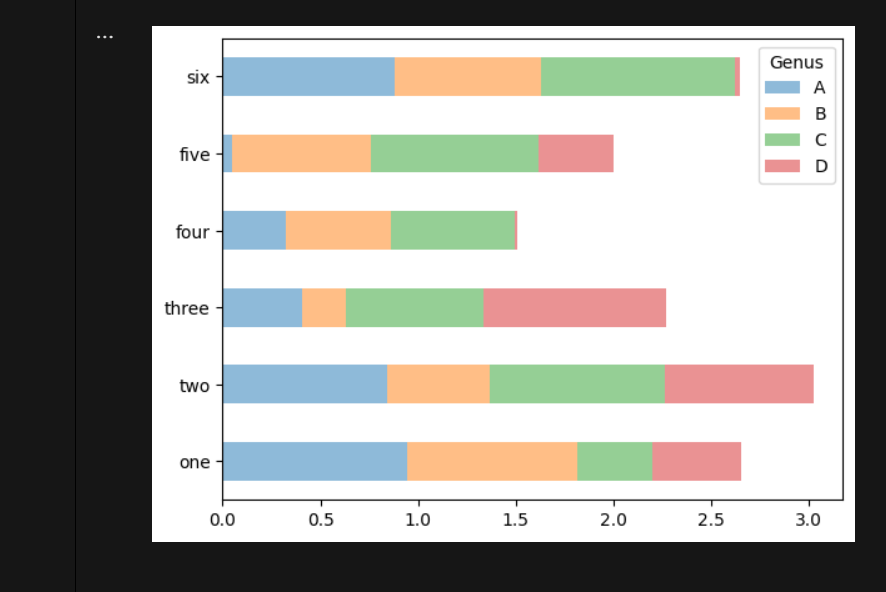
*index*=['one', 'two', 'three', 'four', 'five', 'six'],

*columns*=pd.Index(['A', 'B', 'C', 'D'], *name*='Genus'))

df.plot.bar()



df.plot.barh(*stacked*=True, *alpha*=0.5)



import seaborn as sns

comp1 = np.random.normal(0, 1, *size*=200)

comp2 = np.random.normal(10, 2, *size*=200)

values = pd.Series(np.concatenate([comp1, comp2]))

sns.displot(values, *bins*=100, *color*='k')

