import random

import matplotlib

from matplotlib import pyplot as plt

import operator

random.seed(0) # make x0 and y0 become fixed values

agents = []

x0 = random.randint(0, 99)

y0 = random.randint(0, 99)

agents.append([x0,y0])

plt.scatter(agents[0][0], [1][0], color='black')

plt.scatter(agents[0][0], [0][0], color='yellow' )

plt.show()

print(max(agents, key=operator.itemgetter(0)))

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

agents = []

# create a scatter containing 10 random dots

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

plt.scatter(agents[i][0], agents[i][1], color='black')

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

agents = []

# create a scatter containing 10 random dots

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

plt.scatter(agents[i][0], agents[i][1], color='black')

# highlight the max dot in x-axis with red

r = max(agents, key=operator.itemgetter(0))

print(r) # not necessary

plt.scatter(r[0], r[1], color='red')

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

agents = []

# create a scatter containing 10 random dots

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

plt.scatter(agents[i][0], agents[i][1], color='black')

# highlight the max dot in x-axis with red

r = max(agents, key=operator.itemgetter(0))

print(r) # not necessary

plt.scatter(r[0], r[1], color='red')

# highlight the min dot in x-axis with blue

b = min(agents, key=operator.itemgetter(0))

print(b)

plt.scatter(b[0], b[1], color='blue')

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

agents = []

# create a scatter containing 10 random dots

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

plt.scatter(agents[i][0], agents[i][1], color='black')

# highlight the max dot in x-axis with red

r = max(agents, key=operator.itemgetter(0))

print(r) # not necessary

plt.scatter(r[0], r[1], color='red')

# highlight the min dot in x-axis with blue

b = min(agents, key=operator.itemgetter(0))

print(b)

plt.scatter(b[0], b[1], color='blue')

# highlight the max dot in y-axis with yellow

y = max(agents, key=operator.itemgetter(1))

print(y)

plt.scatter(y[0], y[1], color='yellow')

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

agents = []

# create a scatter containing 10 random dots

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

plt.scatter(agents[i][0], agents[i][1], color='black')

# highlight the max dot in x-axis with red

r = max(agents, key=operator.itemgetter(0))

print(r) # not necessary

plt.scatter(r[0], r[1], color='red')

# highlight the min dot in x-axis with blue

b = min(agents, key=operator.itemgetter(0))

print(b)

plt.scatter(b[0], b[1], color='blue')

# highlight the max dot in y-axis with yellow

y = max(agents, key=operator.itemgetter(1))

print(y)

plt.scatter(y[0], y[1], color='yellow')

# highlight the min dot in y-axis with green

g = min(agents, key=operator.itemgetter(1))

print(g)

plt.scatter(g[0], g[1], color='green')

图表, 散点图

描述已自动生成

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

# make x0 and y0 become fixed values

random.seed(0)

# create 10 agents to replace the single agent

n\_agents = 10

n\_ite = 5

agents = []

# initialise the agents list

for i in range(n\_agents):

agents.append([random.randint(0, 99), random.randint(0, 99)])

print(agents)

for j in range(n\_ite):

# move agents

for i in range(n\_agents):

rn = random.random()

print("rn", rn)

if rn < 40:

agents[i][0] = agents[i][0] + 5

else:

agents[i][0] = agents[i][0] - 5

print("x0", agents[i][0])

rn = random.random()

if rn < 40:

agents[i][1] = agents[i][1] + 5

else:

agents[i][1] = agents[i][1] - 5

print("y0", agents[i][1])

print(agents)

# plot scatter

for i in range(n\_agents):

plt.scatter(agents[i][0], agents[i][1], color='black')

# highlight the max dot in x-axis with red

r = max(agents, key=operator.itemgetter(0))

print(r) # not necessary

plt.scatter(r[0], r[1], color='red')

# highlight the min dot in x-axis with blue

b = min(agents, key=operator.itemgetter(0))

print(b)

plt.scatter(b[0], b[1], color='blue')

# highlight the max dot in y-axis with yellow

y = max(agents, key=operator.itemgetter(1))

print(y)

plt.scatter(y[0], y[1], color='yellow')

# highlight the min dot in y-axis with green

g = min(agents, key=operator.itemgetter(1))

print(g)

plt.scatter(g[0], g[1], color='green')

图表, 散点图

描述已自动生成