*eg1 - Notepad

File Edit Format View Help

```
import matplotlib.pyplot
#following represents A class
matplotlib.pyplot.plot(10,20,"ro")
matplotlib.pyplot.plot(20,25,"ro")
matplotlib.pyplot.plot(15,20,"ro")
matplotlib.pyplot.plot(30,45,"ro")
matplotlib.pyplot.plot(10,10,"ro")
matplotlib.pyplot.plot(50,25,"ro")
matplotlib.pyplot.plot(30,55,"ro")
matplotlib.pyplot.plot(40,60,"ro")


#following represents B class
matplotlib.pyplot.plot(100,120,"go")
matplotlib.pyplot.plot(120,125,"go")
matplotlib.pyplot.plot(115,120,"go")
matplotlib.pyplot.plot(130,145,"go")
matplotlib.pyplot.plot(110,110,"go")
matplotlib.pyplot.plot(150,125,"go")
matplotlib.pyplot.plot(130,155,"go")
matplotlib.pyplot.plot(140,160,"go")

queryX=80
```

Ln 1, Col 1 100% Windows (CRLF) UTF-8

*eg1 - Notepad

File Edit Format View Help

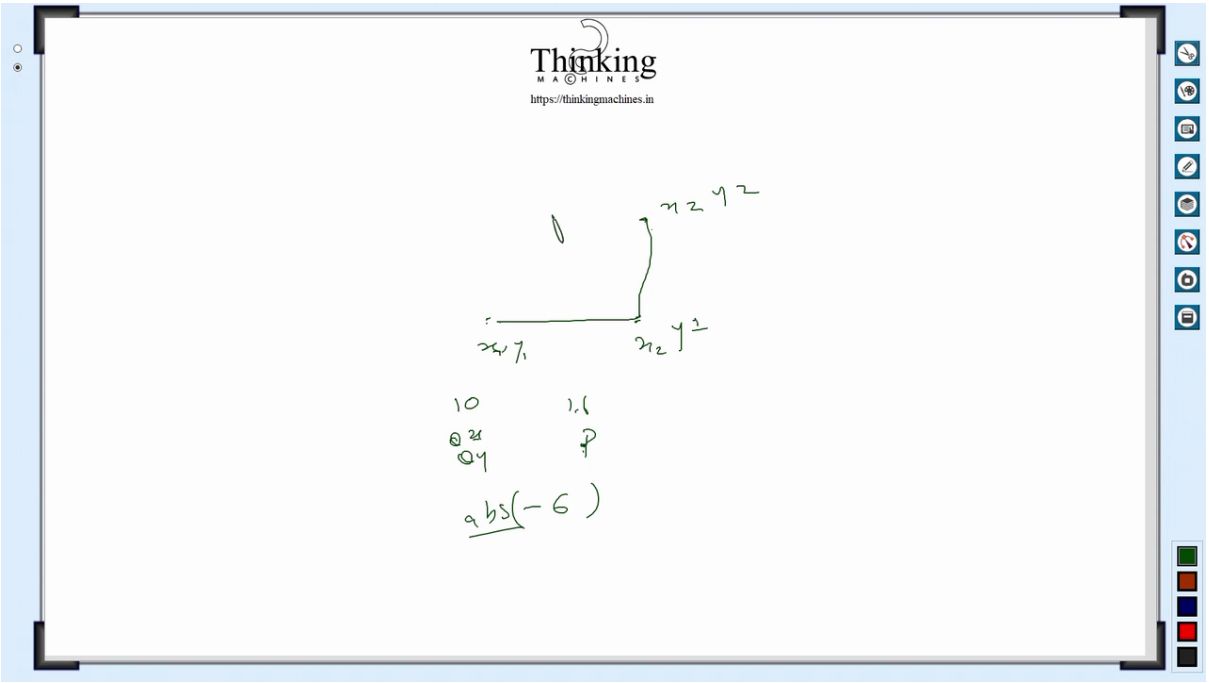
```
matplotlib.pyplot.plot(10,10,"ro")
matplotlib.pyplot.plot(50,25,"ro")
matplotlib.pyplot.plot(30,55,"ro")
matplotlib.pyplot.plot(40,60,"ro")


#following represents B class
matplotlib.pyplot.plot(100,120,"go")
matplotlib.pyplot.plot(120,125,"go")
matplotlib.pyplot.plot(115,120,"go")
matplotlib.pyplot.plot(130,145,"go")
matplotlib.pyplot.plot(110,110,"go")
matplotlib.pyplot.plot(150,125,"go")
matplotlib.pyplot.plot(130,155,"go")
matplotlib.pyplot.plot(140,160,"go")

queryX=80
queryY=100
matplotlib.pyplot.plot(queryX,queryY,color='b',marker='o',markersize=20)

matplotlib.pyplot.show()
```

Ln 29, Col 1 100% Windows (CRLF) UTF-8



*eg1 - Notepad

File Edit Format View Help

```
import statistics
import matplotlib.pyplot
import numpy
import math
def getKey(r):
    return r[1]

#following represents A class
#matplotlib.pyplot.plot(10,20,"ro")
#matplotlib.pyplot.plot(20,25,"ro")
#matplotlib.pyplot.plot(15,20,"ro")
#matplotlib.pyplot.plot(30,45,"ro")
#matplotlib.pyplot.plot(10,10,"ro")
#matplotlib.pyplot.plot(50,25,"ro")
#matplotlib.pyplot.plot(30,55,"ro")
#matplotlib.pyplot.plot(40,60,"ro")
a=numpy.array([[10,20],[20,25],[15,20],[30,45],[10,10],[50,25],[30,55],[4
print(a)
print("*"*50)
print(a.T)
print("*"*50)
x,y=a.T
print(x)
```

Ln 7, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
import statistics
import matplotlib.pyplot
import numpy
import math
def getKey(r):
    return r[1]

#following represents A class
#matplotlib.pyplot.plot(10,20,"ro")
#matplotlib.pyplot.plot(20,25,"ro")
#matplotlib.pyplot.plot(15,20,"ro")
#matplotlib.pyplot.plot(30,45,"ro")
#matplotlib.pyplot.plot(10,10,"ro")
#matplotlib.pyplot.plot(50,25,"ro")
#matplotlib.pyplot.plot(30,55,"ro")
#matplotlib.pyplot.plot(40,60,"ro")
a=numpy.array([[10,20],[20,25],[15,20],[30,45],[10,10],[50,25],[30,55],[4
print(a)
print("*"*50)
print(a.T)
print("*"*50)
x,y=a.T
print(x)
```

Ln 1, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
[40,60]])
print(a)
print("*"*50)
print(a.T)
print("*"*50)
x,y=a.T
print(x)
print("*"*50)
print(y)
print("*"*50)
matplotlib.pyplot.scatter(x,y,color='r')
#following represents B class
#matplotlib.pyplot.plot(100,120,"go")
#matplotlib.pyplot.plot(120,125,"go")
#matplotlib.pyplot.plot(115,120,"go")
#matplotlib.pyplot.plot(130,145,"go")
#matplotlib.pyplot.plot(110,110,"go")
#matplotlib.pyplot.plot(150,125,"go")
#matplotlib.pyplot.plot(130,155,"go")
#matplotlib.pyplot.plot(140,160,"go")
b=numpy.array([[100,120],[120,125],[115,120],[130,145],[110,110],
[150,125],[130,155],[140,160]])
x,y=b.T
matplotlib.pyplot.scatter(x,y,color='g')
```

Ln 21, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
matplotlib.pyplot.scatter(x,y,color='r')
#following represents B class
#matplotlib.pyplot.plot(100,120,"go")
#matplotlib.pyplot.plot(120,125,"go")
#matplotlib.pyplot.plot(115,120,"go")
#matplotlib.pyplot.plot(130,145,"go")
#matplotlib.pyplot.plot(110,110,"go")
#matplotlib.pyplot.plot(150,125,"go")
#matplotlib.pyplot.plot(130,155,"go")
#matplotlib.pyplot.plot(140,160,"go")
b=numpy.array([[100,120],[120,125],[115,120],[130,145],[110,110],
[150,125],[130,155],[140,160]])
x,y=b.T
matplotlib.pyplot.scatter(x,y,color='g')
matplotlib.pyplot.grid(True)
queryX=80
queryY=100
matplotlib.pyplot.plot(queryX,queryY,color='b',marker='o',markersize=20)
#KNN Algorithm starts
distancesList=[]
for p in a:
    xd=abs(p[0]-queryX)
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
```

Ln 48, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
b=numpy.array([[100,120],[120,125],[115,120],[130,145],[110,110],
[150,125],[130,155],[140,160]])
x,y=b.T
matplotlib.pyplot.scatter(x,y,color='g')
matplotlib.pyplot.grid(True)
queryX=80
queryY=100
matplotlib.pyplot.plot(queryX,queryY,color='b',marker='o',markersize=20)
#KNN Algorithm starts
distancesList=[]
for p in a:
    xd=abs(p[0]-queryX)
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
    distancesList.append(('A',d))

for p in b:
    xd=abs(p[0]-queryX)
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
    distancesList.append(('B',d))

print("*"*50)
print(distancesList)
```

Ln 58, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
    distancesList.append(('A',d))

for p in b:
    xd=abs(p[0]-queryX)
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
    distancesList.append(('B',d))

print("*"*50)
print(distancesList)
#distancesList.sort(key=getKey)
distancesList.sort(key=lambda r: r[1])
print("*"*50)
print(distancesList)
kFactor=int(math.sqrt(len(a)+len(b)))
if kFactor%2==0: kFactor+=1
print('k factor : ',kFactor)
kElementsFromTop=distancesList[:5]
print(kElementsFromTop)
classes=list(x[0] for x in kElementsFromTop)
print(classes)
identifiedClass=statistics.mode(classes)
```

Ln 70, Col 1 100% Windows (CRLF) UTF-8

eg1 - Notepad

File Edit Format View Help

```
    yd=abs(p[1]-queryY)
    d=math.sqrt(xd**2 + yd**2)
    distancesList.append(('B',d))

print("*"*50)
print(distancesList)
#distancesList.sort(key=getKey)
distancesList.sort(key=lambda r: r[1])
print("*"*50)
print(distancesList)
kFactor=int(math.sqrt(len(a)+len(b)))
if kFactor%2==0: kFactor+=1
print('k factor : ',kFactor)
kElementsFromTop=distancesList[:5]
print(kElementsFromTop)
classes=list(x[0] for x in kElementsFromTop)
print(classes)
identifiedClass=statistics.mode(classes)
print(f"({queryX},{queryY}) belongs to {identifiedClass}")

matplotlib.pyplot.show()
```

Ln 76, Col 1 100% Windows (CRLF) UTF-8