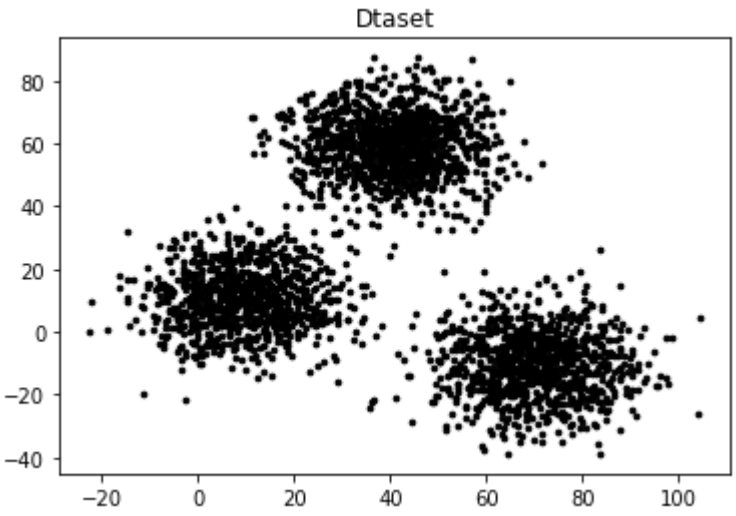
In [1]:
```
import pandas as pd
import numpy as np
from sklearn.mixture import GaussianMixture
from sklearn.cluster import KMeans
import sklearn.metrics as sm
import matplotlib.pyplot as plt
```

In [2]:
```
data=pd.read_csv("prog8-Kmeans_EM.csv")
f1=data['V1'].values
f2=data['V2'].values
x=np.array(list(zip(f1,f2)))
plt.scatter(f1,f2, c='black',s=7)
plt.title("Dtaset")
```

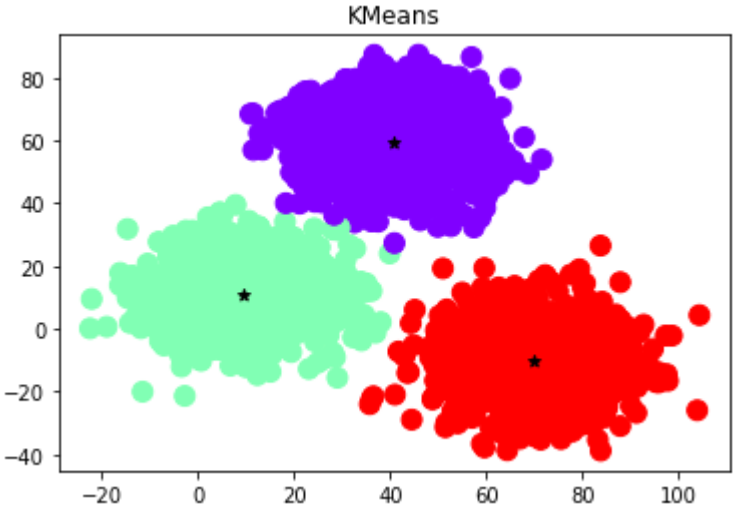
Out[2]:
```
Text(0.5, 1.0, 'Dtaset')
```



In [5]:
```
k=3
kmeans=KMeans(n_clusters=k)
kmeans=kmeans.fit(x)
labels=kmeans.predict(x)
centroids=kmeans.cluster_centers_
print("CENTROIDS")
print(centroids)
plt.title("KMeans")
plt.scatter(x[:,0],x[:,1],c=labels, cmap='rainbow',s=100)
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],marker='*',
```

```
CENTROIDS
[[ 40.68362784  59.71589274]
 [  9.4780459   10.686052  ]
 [ 69.92418447 -10.11964119]]
```

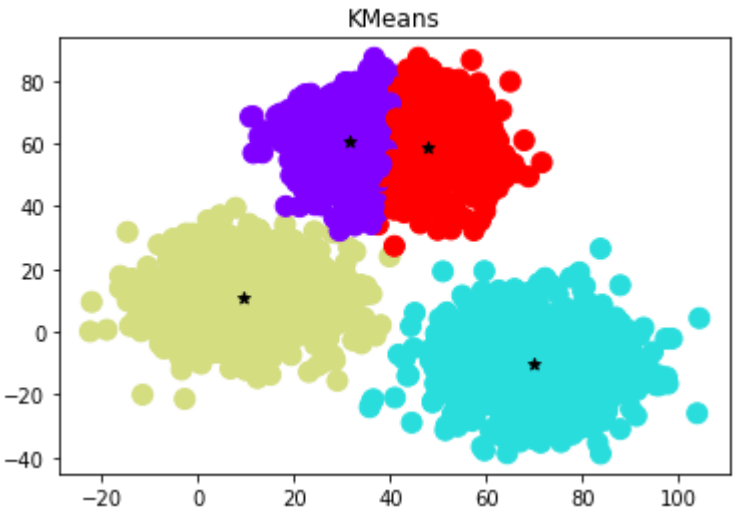
Out[5]: <matplotlib.collections.PathCollection at 0x20bef1da1c8>



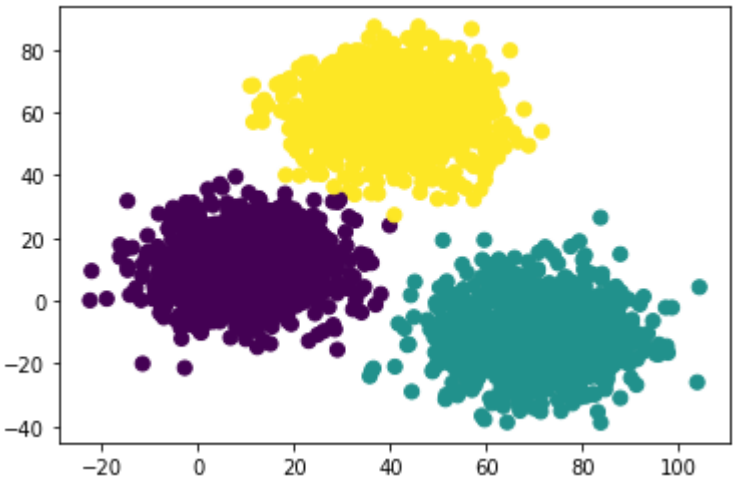
In [6]:
```
k=4
kmeans=KMeans(n_clusters=k)
kmeans=kmeans.fit(x)
labels=kmeans.predict(x)
centroids=kmeans.cluster_centers_
print("CENTROIDS")
print(centroids)
plt.title("KMeans")
plt.scatter(x[:,0],x[:,1],c=labels, cmap='rainbow',s=100)
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],marker='*',
```

```
CENTROIDS
[[ 31.44293118  60.57619674]
 [ 69.92418447 -10.11964119]
 [  9.45577774  10.66209744]
 [ 47.77461177  59.01178406]]
```

Out[6]: <matplotlib.collections.PathCollection at 0x20bf1563348>



In [11]:
```
gmm=GaussianMixture(n_components=3).fit(x)
labels=gmm.predict(x)
plt.scatter(x[:,0],x[:,1],c=labels,s=50,cmap='viridis')
plt.show()
print("EM predictions")
probs=gmm.predict_proba(x)
print(probs)
print("Kmeans:\n",gmm.means_)
print("Covariances:\n",gmm.covariances_)
```

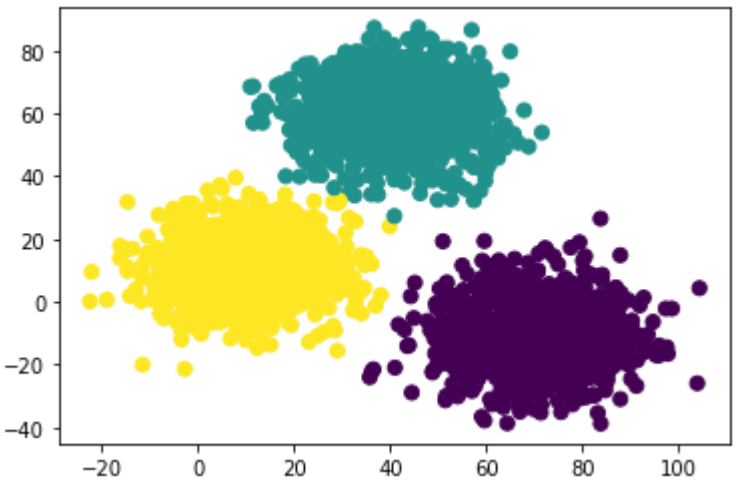



```
EM predictions
[[9.99999998e-01 2.47857809e-09 1.51560970e-12]
 [9.99994655e-01 3.50898583e-07 4.99388810e-06]
 [1.00000000e+00 1.79826264e-10 3.06364433e-10]
 ...
 [3.31608645e-08 9.99999967e-01 5.88532728e-13]
 [3.40814080e-15 1.00000000e+00 1.18181069e-16]
 [7.83207859e-10 9.99999999e-01 1.17363874e-17]]
Kmeans:
 [[  9.46516999  10.71326558]
 [ 69.89436808 -10.11243325]
 [ 40.68585643  59.70939105]]
Covariences:
 [[[103.72485985   3.83148201]
  [  3.83148201  96.80338694]]

 [[110.42521599  -0.60038245]
  [ -0.60038245 106.47434137]]

 [[103.11540751  -1.82258688]
  [ -1.82258688  94.29478126]]]
```

In [13]:
```
gmm=GaussianMixture(n_components=3).fit(x)
labels=gmm.predict(x)
plt.scatter(x[:,0],x[:,1],c=labels,s=50,cmap='viridis')
plt.show()
print("EM predictions")
probs=gmm.predict_proba(x)
print(probs)
print("Kmeans:\n",gmm.means_)
print("Covariances:\n",gmm.covariances_)
```




```
EM predictions
[[2.47857809e-09 1.51560970e-12 9.99999998e-01]
 [3.50898583e-07 4.99388810e-06 9.99994655e-01]
 [1.79826264e-10 3.06364433e-10 1.00000000e+00]
 ...
 [9.99999967e-01 5.88532728e-13 3.31608645e-08]
 [1.00000000e+00 1.18181069e-16 3.40814080e-15]
 [9.99999999e-01 1.17363874e-17 7.83207859e-10]]
Kmeans:
 [[ 69.89436808 -10.11243325]
 [ 40.68585643  59.70939105]
 [  9.46516999  10.71326558]]
Covariances:
 [[[110.42521599  -0.60038245]
  [ -0.60038245 106.47434137]]

 [[103.11540751  -1.82258688]
  [ -1.82258688  94.29478126]]

 [[103.72485985   3.83148201]
  [  3.83148201  96.80338694]]]
```

In [ ]: