เริ่มต้นด้วยการ import

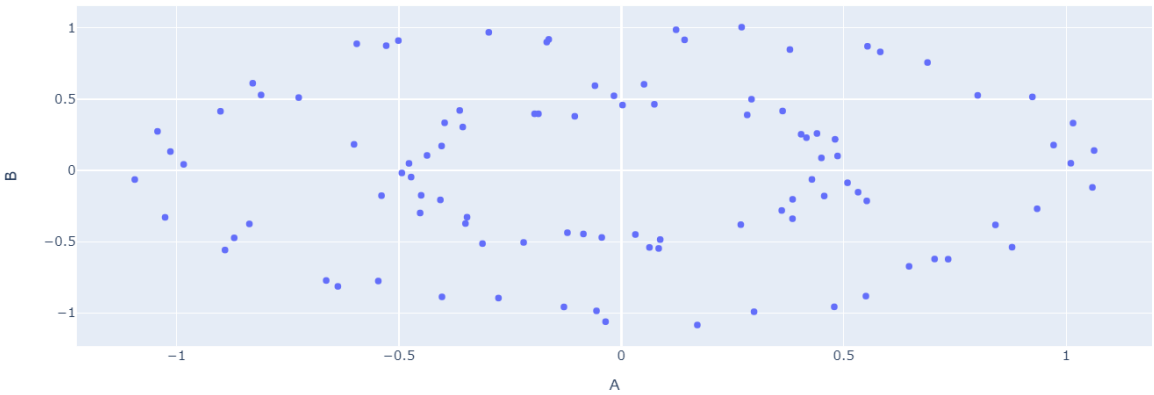
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import cluster
import pandas as pd
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.spatial import distance_matrix
```

```
data = pd.read_csv("data/data2Dset4.csv",header=None)
data.columns = ["A","B"]
data.head()
```

	A	B
0	-0.016537	0.523408
1	0.971732	0.177843
2	-0.403983	0.171210
3	-0.406715	-0.207174
4	-0.055831	-0.985840

2) Plot จุดข้อมูล data1

```
import plotly.express as px
fig = px.scatter(data, x="A", y="B")
fig.show()
```



3) Data2Dset4 เริ่มด้วยการเขียนโปรแกรม plot จุดข้อมูลโดยใช้วิธี kmeans

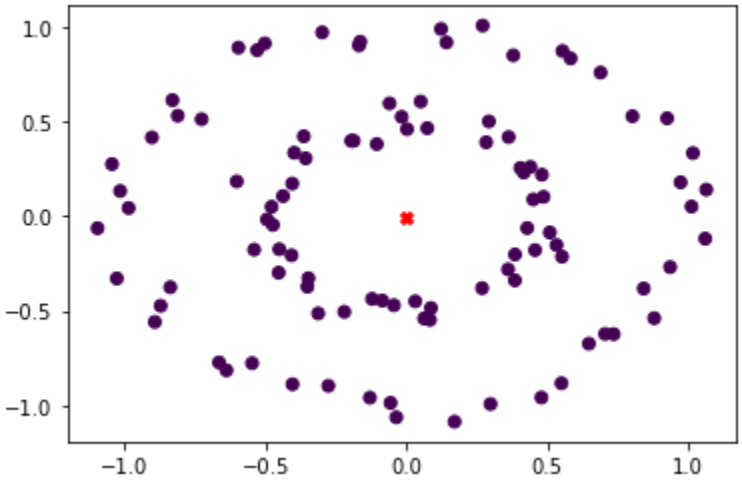
K=1

```
model_kmeans = cluster.KMeans(n_clusters=1, max_iter=50, random_state=1)
model_kmeans.fit(data)
data1['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-0.0020635  -0.00248395  0.        ]]
```

```
plt.scatter(data['A'],data['B'], c=data1['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()
```

ผลลัพธ์ K=1



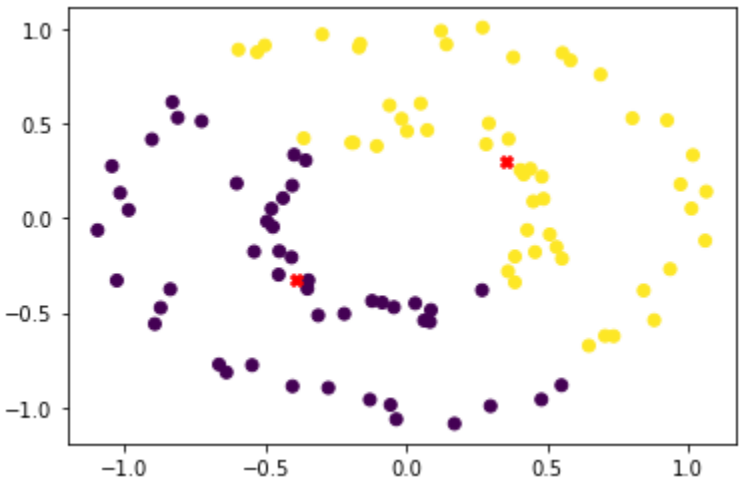
K=2

```
model_kmeans = cluster.KMeans(n_clusters=2, max_iter=50, random_state=1)
model_kmeans.fit(data)
data1['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-0.38961084 -0.3252557   0.        ]
 [ 0.35567251  0.2954592   0.        ]]
```

```
plt.scatter(data['A'],data['B'], c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()
```

ผลลัพธ์ K=2



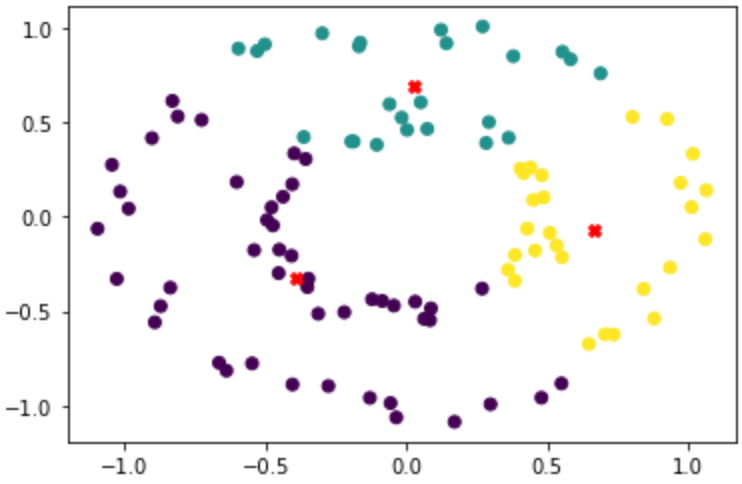
K=3

```
model_kmeans = cluster.KMeans(n_clusters=3, max_iter=50, random_state=1)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-3.89610837e-01 -3.25255700e-01  4.44089210e-16]
 [ 2.50569961e-02  6.88911462e-01  1.00000000e+00]
 [ 6.61797986e-01 -6.88484582e-02  1.00000000e+00]]
```

```
plt.scatter(data['A'],data['B'], c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()
```

ผลลัพธ์ K=3



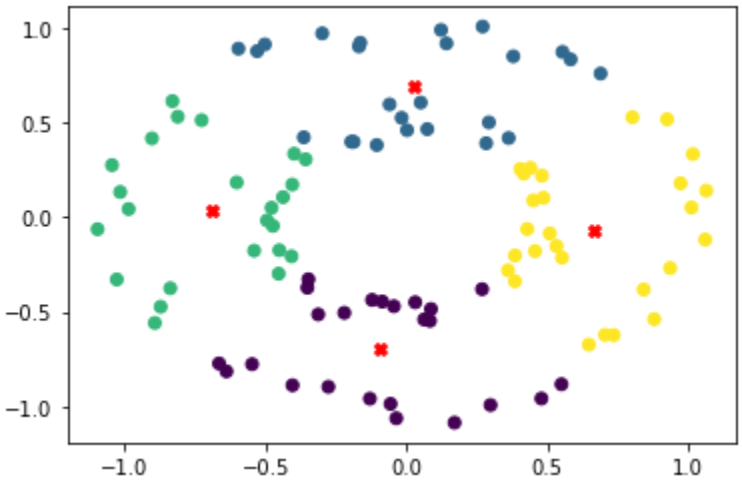
K=4

```
model_kmeans = cluster.KMeans(n_clusters=4, max_iter=50, random_state=1)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-9.13911135e-02 -6.89520810e-01  4.44089210e-16]
 [ 6.61797986e-01 -6.88484582e-02  2.00000000e+00]
 [ 2.50569961e-02  6.88911462e-01  1.00000000e+00]
 [-6.87830561e-01  3.90094112e-02  4.44089210e-16]]
```

```
plt.scatter(data['A'],data['B'], c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()
```

ผลลัพธ์K=4



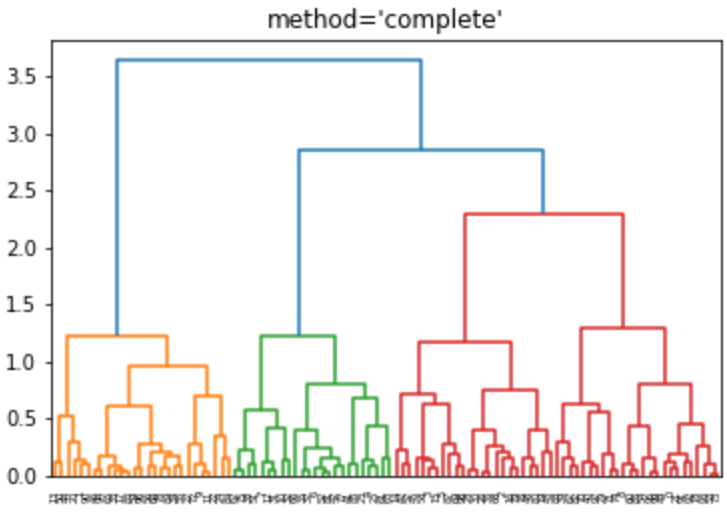
จากผลลัพธ์สรุปได้ว่า K=4 ดีที่สุดสำหรับ Data2Dset2 เพราะข้อมูลจะดูละเอียดอ่อนมากขึ้นถ้าเทียบกับKอื่นๆ ตัวข้อมูลจะถูกแบ่งเป็น 4กลุ่มใหญ่ๆอย่างเห็นได้ชัด

4) เขียนโปแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี Hierarchical Clustering

4.1) ให้เลือกใช้method ที่ต่างกัน 3 แบบ แสดง dendrogram ที่ได้แต่ละแบบ

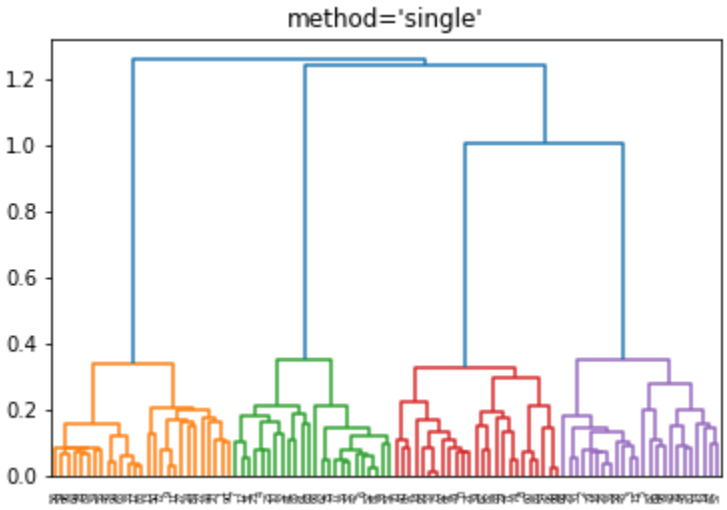
Complete:

```
linkage_data = linkage(data, method='complete' , metric='euclidean')
dendrogram(linkage_data)
plt.title("method='complete'")
plt.show()
```



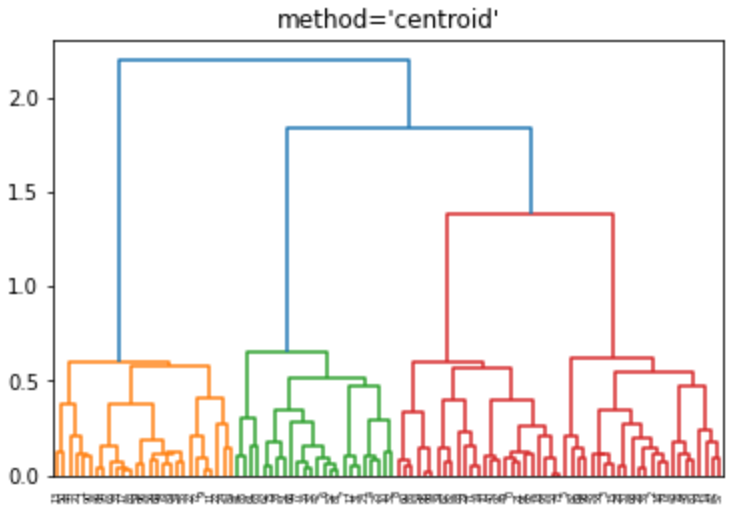
Single:

```
linkage_data = linkage(data, method='single' , metric='euclidean')
dendrogram(linkage_data)
plt.title("method='single'")
plt.show()
```



Centroid:

```
linkage_data = linkage(data, method='centroid' , metric='euclidean')
dendrogram(linkage_data)
plt.title("method='centroid'")
plt.show()
```



4.2) เลือก cut-off โดยกำหนด criterion='distance' และให้นักศึกษาเลือกระบุค่า t ที่คิดว่า เหมาะสม สำหรับแต่ละ dendrogram ที่ได้ในข้อ 4.1)

Complete: ค่า t ที่เหมาะที่สุดคือ 1.5

```
cluster_id = fcluster(linkage_data,t=1.5,criterion='distance')
plt.scatter(data["A"],data["B"],c=cluster_id)
plt.show()
```

Single: ค่า t ที่เหมาะที่สุดคือ 1

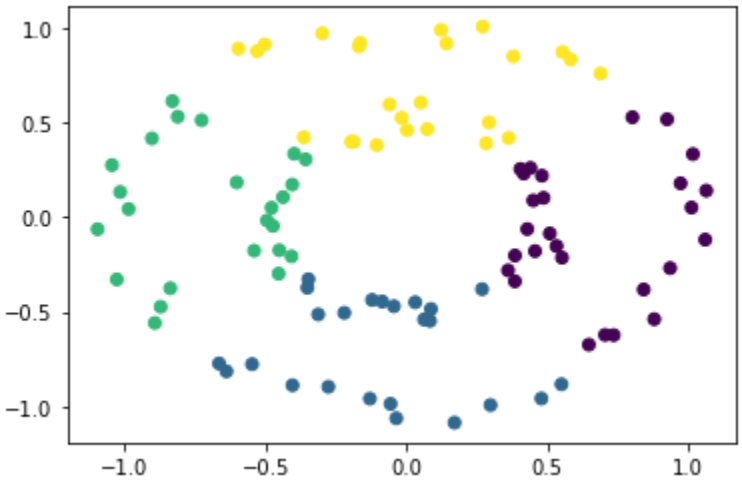
```
cluster_id = fcluster(linkage_data,t=1,criterion='distance')
plt.scatter(data["A"],data["B"],c=cluster_id)
plt.show()
```

Centroid: ค่า t ที่เหมาะที่สุดคือ 1

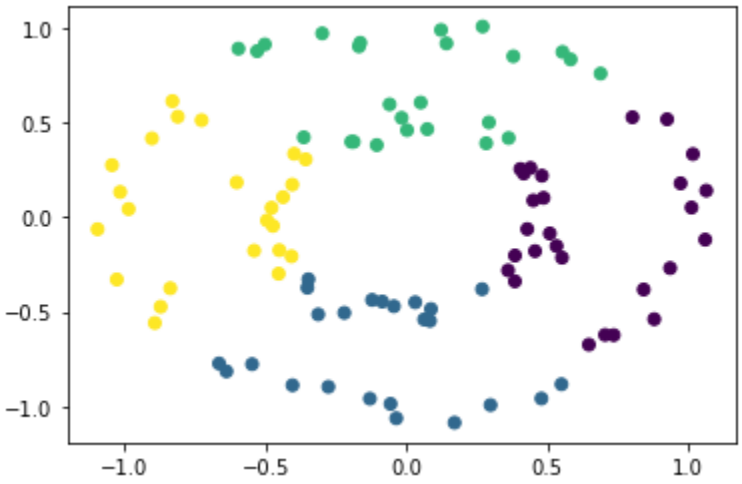
```
cluster_id = fcluster(linkage_data,t=1,criterion='distance')
plt.scatter(data["A"],data["B"],c=cluster_id)
plt.show()
```

4.3) Plot ผลการจัดกลุ่ม ที่ได้แต่ละแบบในข้อ 4.2)

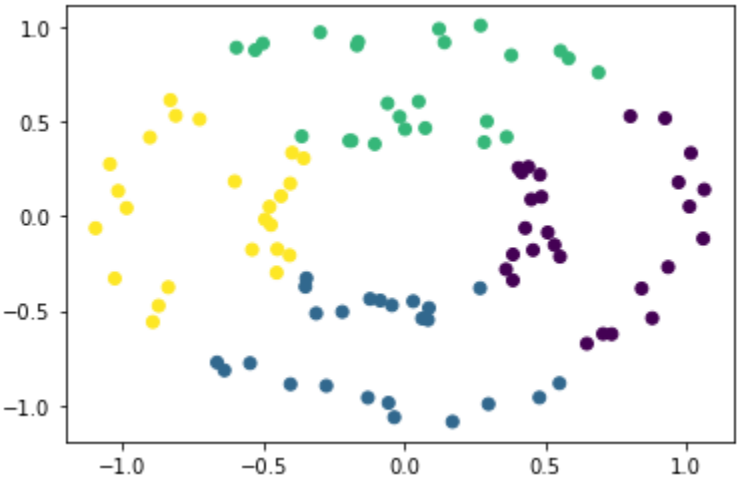
Complete:



Single:



Centroid:



6. เขียนบรรยายสรุปผลการทดลอง แสดงความคิดเห็น วิธีใด เหมาะกับ ชุดข้อมูลแบบไหน แต่ละวิธีมีข้อดี/ ข้อเสีย อย่างไร

- Data2DSet2 จากการทดลองพบว่า วิธี Hierarchical Clusterings เพราะจะมีความละเอียดมากกว่าและจากข้อมูลเราไม่สามารถแยกข้อมูลด้วยตาเปล่าได้ การใช้ kmeans จึงค่อนข้างยาก