CODE:

Heriarical Clustering:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

dataset = pd.read_csv('Mall_Customers.csv')
X = dataset.iloc[:, [3, 4]].values

#using dendogram for finding optimal number of clusterings
import scipy.cluster.hierarchy as sch
dendrogram = sch.dendrogram(sch.linkage(X, method='ward'))
plt.title('Dendrogram')
plt.xlabel('Customers')
plt.ylabel('Eucladian Distances')
plt.show()

#training the cluster Using HC
from sklearn.cluster import AgglomerativeClustering
hc = AgglomerativeClustering(n_clusters = 3, affinity='euclidean', linkage='ward')
y_hc = hc.fit_predict(X)

plt.scatter(X[y_hc == 0, 0], X[y_hc == 0, 1], s = 100, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_hc == 1, 0], X[y_hc == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')
plt.scatter(X[y_hc == 2, 0], X[y_hc == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```

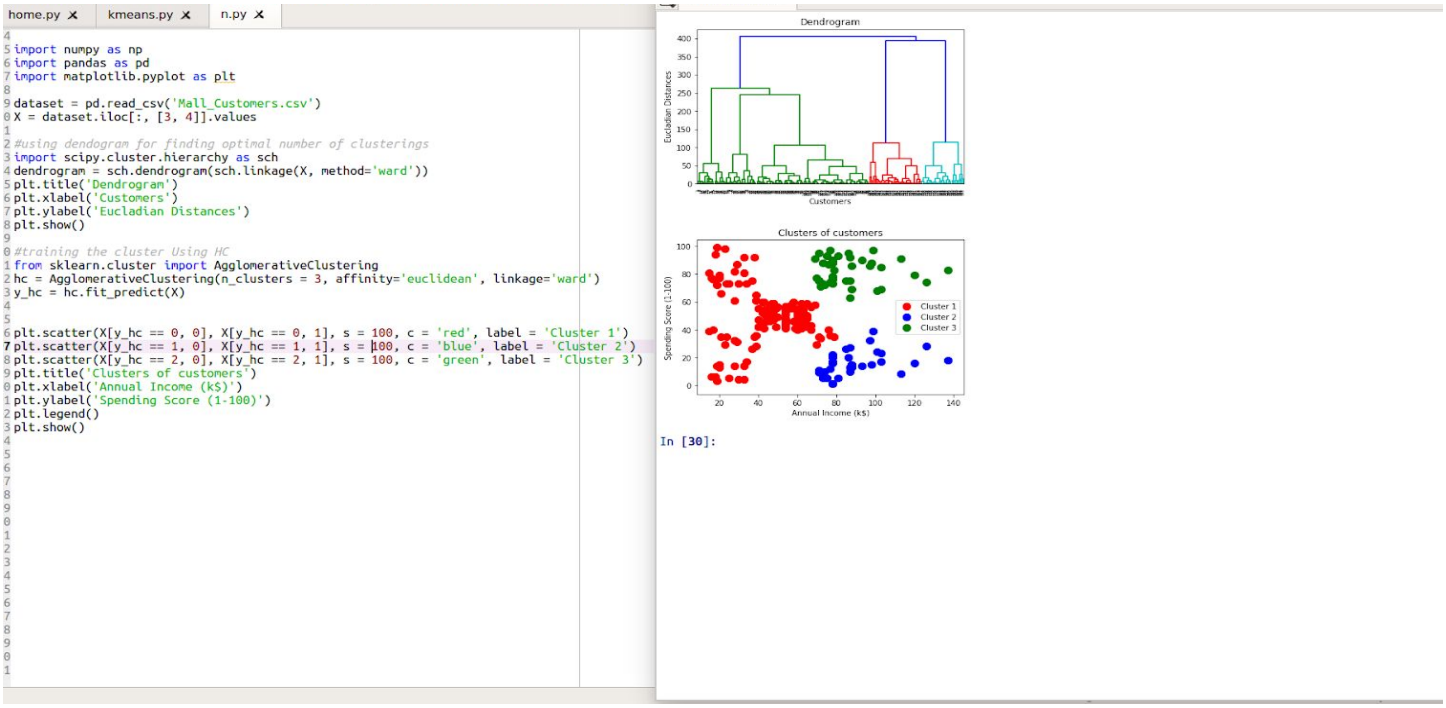
KMeans Clustering:

```
# Importing the libraries
from sklearn import datasets
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

#Dataset
iris = datasets.load_iris()

# Importing the dataset
x = iris.data[:, :2]
y = iris.data[:, :2]
```

```
# Using the elbow method to find the optimal number of clusters
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4)
y_kmeans = kmeans.fit_predict(x)

# print(y_kmeans)
kmeans.cluster_centers_

# Fitting K-Means to the dataset
plt.scatter(x[:,0], x[:,1], c=y_kmeans, cmap='gist_rainbow')
plt.xlabel('Sepal Length')
plt.ylabel('Sepal Width')
```

OUTPUT:

HIERARCHICAL CLUSTERING



K-MEANS CLUSTERING:

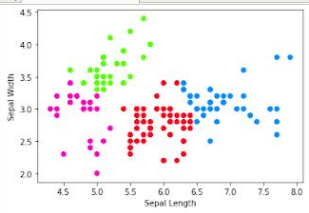
```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Tue Sep 29 16:11:43 2020

@author: srushti
"""

# K-Means Clustering

# Importing the libraries
from sklearn import datasets
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

#Dataset
iris = datasets.load_iris()

# Importing the dataset
x = iris.data[:, :2]
# print(x)
y = iris.data[:, :2]
# print(y)

# Using the elbow method to find the optimal number of clusters
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4)
y_kmeans = kmeans.fit_predict(x)

# print(y_kmeans)
kmeans.cluster_centers_


# Fitting K-Means to the dataset
plt.scatter(x[:,0], x[:,1], c=y_kmeans, cmap='gist_rainbow')
plt.xlabel('Sepal Length')
plt.ylabel('Sepal Width')
```

Name	Size	Type	Date Modified
n.py	3 KB	py File	27/09/20 9:48 PM
natural_language_processing.py	1 KB	py File	23/11/16 2:22 AM
natural_language_processing.R	1 KB	R File	23/11/16 2:24 AM
Restaurant_Reviews.tsv	59 KB	tsv File	15/11/16 8:08 AM

Help | Variable explorer | File explorer

IPython console

Console 1/A




```
In [28]:
```

IPython console | History log

Permissions: RW End-of-lines: LF Encoding: UTF-8 Line: 31 Column: 18 Memory: 79 %