In [ ]:

```
#Implementing Agglomerative Clustering in python
# MCA-II                                        CA LAB-VII(A): LAB on Machine Learning
```

In [4]:

```
from sklearn import datasets
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

In [5]:

```
#loading the dataset and split into input and output.
iris = datasets.load_iris()
X=iris['data']
Y=iris.target
print(X.shape)
```

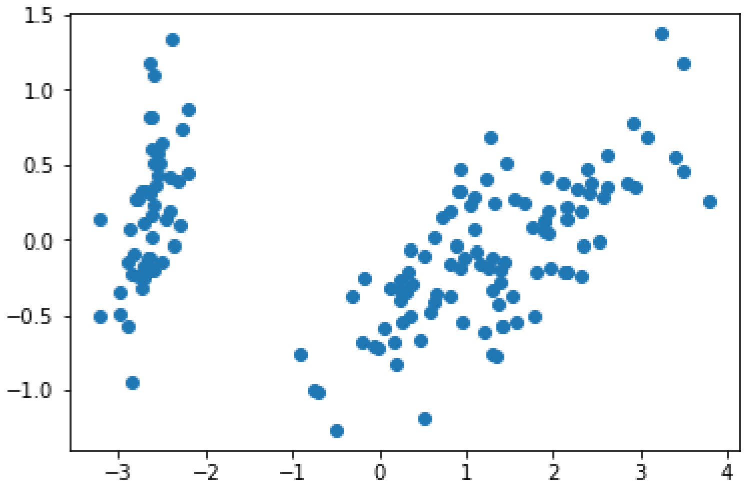
```
(150, 4)
```

In [6]:

```
#now perform Principal Component Analysis to reduce the features from four to two, f
X = PCA(n_components=2).fit_transform(X)
plt.scatter(X[:,0],X[:,1])
```

Out[6]:

```
<matplotlib.collections.PathCollection at 0x1b002764a60>
```



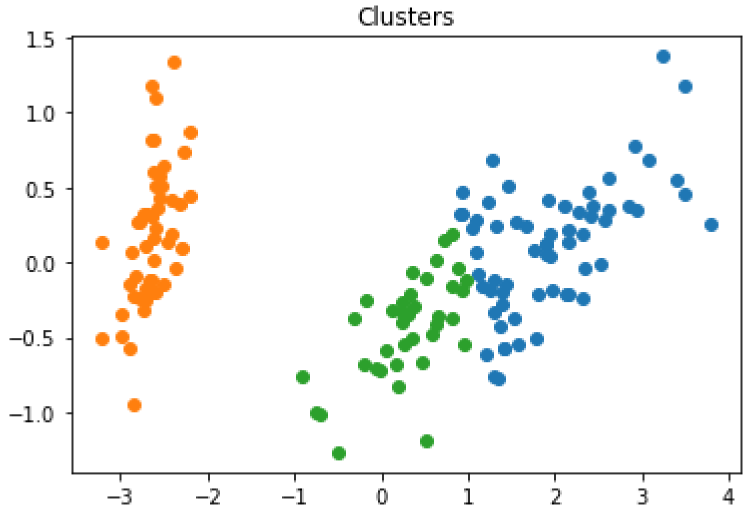
In [8]:

```
#The Agglomerative clustering module present inbuilt in sklearn is used for this pur
from sklearn.cluster import AgglomerativeClustering
classifier = AgglomerativeClustering(n_clusters = 3, affinity = 'euclidean', linkage
clusters = classifier.fit_predict(X)
```

In [9]:

```
#Visualize the results of the clustering algorithm.

plt.scatter(X[clusters == 0, 0], X[clusters == 0, 1], label = 'Type 1')
plt.scatter(X[clusters == 1, 0], X[clusters == 1, 1], label = 'Type 2')
plt.scatter(X[clusters == 2, 0], X[clusters == 2, 1], label = 'Type 3')
plt.title('Clusters')
plt.show()
```



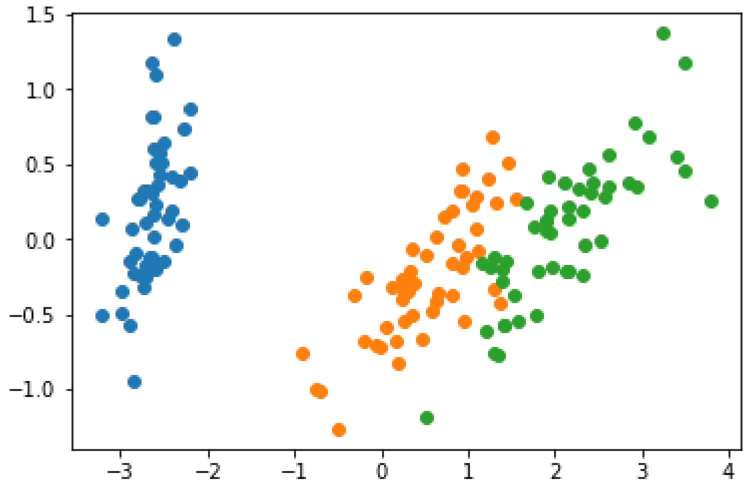
In [10]:

```
#Compare it with the actual target values of the data and can hence see that the grou

plt.scatter(X[Y == 0, 0], X[Y == 0, 1], label = 'Type 1')
plt.scatter(X[Y == 1, 0], X[Y == 1, 1], label = 'Type 2')
plt.scatter(X[Y == 2, 0], X[Y == 2, 1], label = 'Type 3')
```

Out[10]:

```
<matplotlib.collections.PathCollection at 0x1b002b8efd0>
```



In [ ]:

```

```