Clustering:

--> Example of Unsupervised Learning.

--> Clustering divides the data points into groups based on siilarity or distance measure.

--> Goal: Smaples with in the cluster are vary similar and samples in different clusters are different.

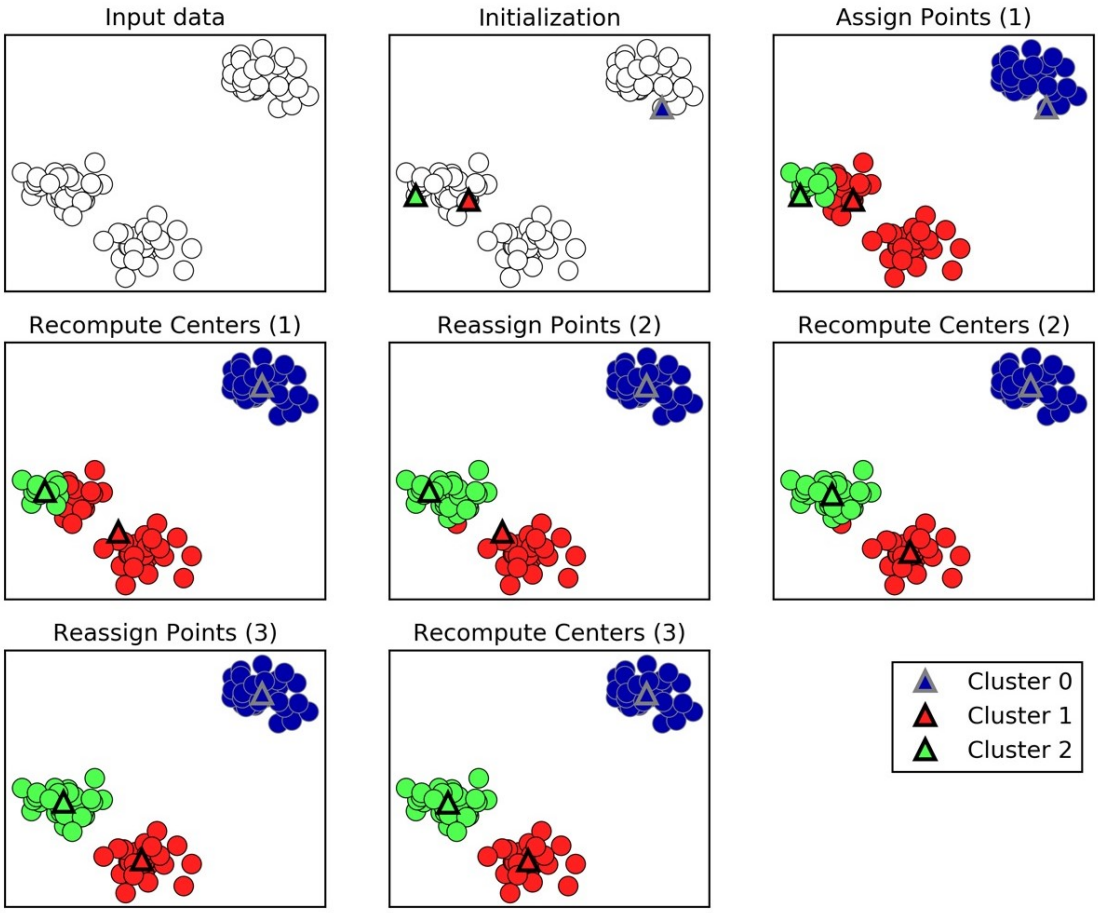
K-Means Algorithm

--> Inputs: X - Input samples and K - Number of clusters need to be created.

--> Outputs: Group Labels assigned to each input and Cluster centers.

--> Pocedure:

```
i.   Randomly picks centroids for K clusters
ii.  Each data point is assigned to one of the K clusters based on th
e distance from centroids.
iii. Recalculate each centroid using the mean of all the data points
assigned to that cluster.
iV.  Repeat the steps ii and iii until there is no further re-arrange
ment of cluster centers.
```



Step 1: Load iris dataset and trim it with only two features, petal length and petal width.

--> Import load_iris dataset from sklearn.datasets module.

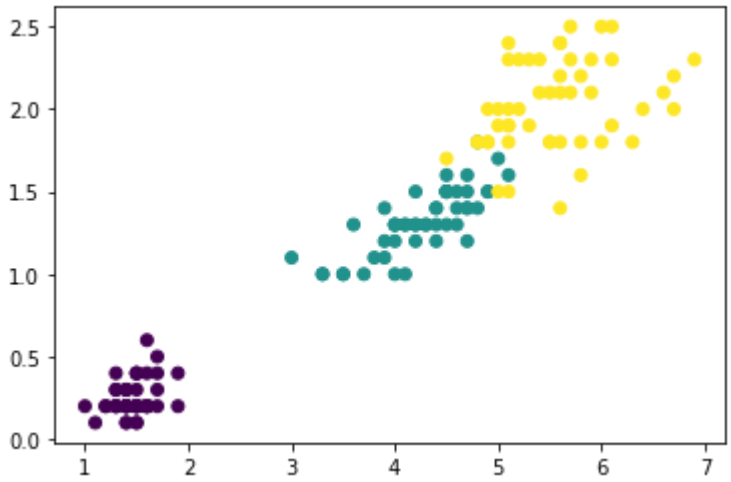
--> Visualize the input samples with different colors for the categories.

```
* Use matplotlib.pyplot.scatter(feature1, feature2, c=categorical_var
iable)
```

In [1]:
```
from sklearn.datasets import load_iris

dataset = load_iris()
print(dataset.feature_names)
print(dataset.target_names)
x = dataset.data[:, 2:4]
y = dataset.target
print(x.shape)
print(y.shape)
print(x[0:5])
print(y[0:5])
```

```
['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width
(cm)']
['setosa' 'versicolor' 'virginica']
(150, 2)
(150,)
[[1.4 0.2]
 [1.4 0.2]
 [1.3 0.2]
 [1.5 0.2]
 [1.4 0.2]]
[0 0 0 0 0]
```

In [2]:
```
import matplotlib.pyplot as plt

plt.scatter(x[:,0], x[:, 1], c=y)
plt.show()
```



Step 2: Fit the KMeans clustering model

--> Import the KMeans class from sklearn.cluster module.

--> Create a model object using n_clusters attribute.

--> Fit the model using fit(x) method.

--> Use labels_ attribute to get labels assigned to all samples.

--> Use cluster_centers_ attribute to get cluster centers.

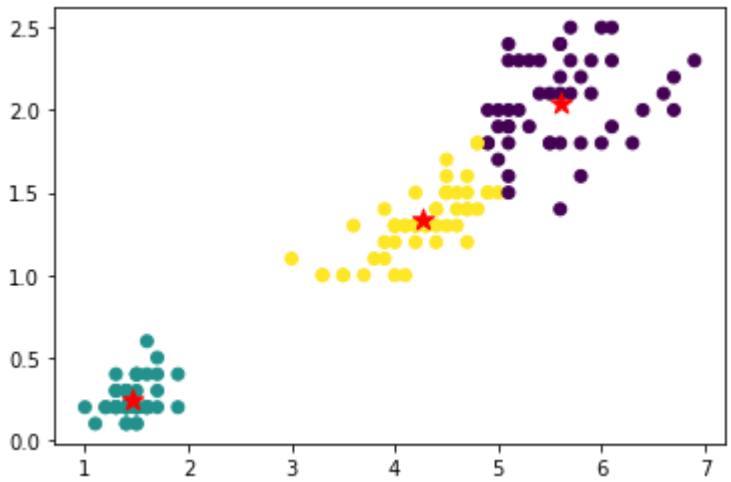
--> Use predict method to know the cluster label for new data.

In [11]:
```
from sklearn.cluster import KMeans

kc = KMeans(n_clusters=3)
kc.fit(x)

labels = kc.labels_
centers = kc.cluster_centers_

print("Sample Group Labels:")
print(labels)
print("Cluster Centers: ")
print(centers)
```

```
Sample Group Labels:
[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 2 0 0 0 0
 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0
 0 0]
Cluster Centers:
[[5.59583333 2.0375    ]
 [1.462      0.246     ]
 [4.26923077 1.34230769]]
```

Step 3: Visulaize the clusters with centers

In [12]:
```
plt.scatter(x[:,0], x[:, 1], c=labels)

plt.scatter(centers[:, 0], centers[:, 1], color = "red", marker = "*", s = 120
plt.show()
```



In [15]:
```
import numpy as np

print(kc.predict(np.array([2.4, 1.3]).reshape(1, -1)))
```

```
[1]
```

In [ ]:
```

```

In [ ]:
```

```