In [6]:
```python
import pandas as pd

ratings = [['kiranmai',5,5,2,1,1,2],['archana',4,5,3,2,2,3],['anitha',4,4,4,3,3,4],['varshitha',2,2,4,5,5,4],['poojitha',1,2,3,4,
titles = ['user','RRR','Kgf','Kanchana','Ganga','f2','jathiratnalu']
movies = pd.DataFrame(ratings,columns=titles)
movies
```

Out[6]:

	user	RRR	Kgf	Kanchana	Ganga	f2	jathiratnalu
0	kiranmai	5	5	2	1	1	2
1	archana	4	5	3	2	2	3
2	anitha	4	4	4	3	3	4
3	varshitha	2	2	4	5	5	4
4	poojitha	1	2	3	4	4	3
5	manasa	2	1	5	5	5	5

In [7]:
```python
from sklearn import cluster

data = movies.drop('user',axis=1)
k_means = cluster.KMeans(n_clusters=2, max_iter=50, random_state=1)
k_means.fit(data)
labels = k_means.labels_
pd.DataFrame(labels, index=movies.user, columns=['Cluster ID'])
```

Out[7]:

	Cluster ID
user	
kiranmai	1
archana	1
anitha	1
varshitha	0
poojitha	0
manasa	0

In [9]:
```python
centroids = k_means.cluster_centers_
pd.DataFrame(centroids,columns=data.columns)
```

Out[9]:

	RRR	Kgf	Kanchana	Ganga	f2	jathiratnalu
0	1.666667	1.666667	4.0	4.666667	4.666667	4.0
1	4.333333	4.666667	3.0	2.000000	2.000000	3.0

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

testData = np.array([[4,5,1,2,2,1],[3,2,4,4,4,4],[2,3,4,1,1,4],[3,2,3,3,3,3],[5,4,1,4,4,1]])
labels = k_means.predict(testData)
labels = labels.reshape(-1,1)
usernames = np.array(['chinnu','shannu','sanju','bunny','chintu']).reshape(-1,1)
cols = movies.columns.tolist()
cols.append('Cluster ID')
newusers = pd.DataFrame(np.concatenate((usernames, testData, labels), axis=1),columns=cols)
newusers
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but KMeans wa
s fitted with feature names
  warnings.warn(
```

Out[15]:

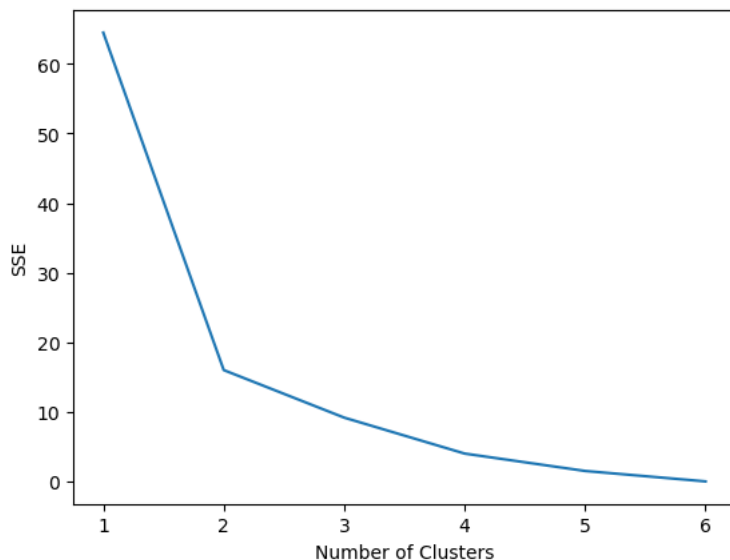|	user	RRR	Kgf	Kanchana	Ganga	f2	jathiratnalu	Cluster ID
0	chinnu	4	5	1	2	2	1	4
1	shannu	3	2	4	4	4	4	0
2	sanju	2	3	4	1	1	4	1
3	bunny	3	2	3	3	3	3	3
4	chintu	5	4	1	4	4	1	1

In [14]:
```python
import matplotlib.pyplot as plt
%matplotlib inline

numClusters = [1,2,3,4,5,6]
SSE = []
for k in numClusters:
    k_means = cluster.KMeans(n_clusters=k)
    k_means.fit(data)
    SSE.append(k_means.inertia_)

plt.plot(numClusters, SSE)
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
```

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\cluster_kmeans.py:1036: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(

Out[14]: Text(0, 0.5, 'SSE')

In [ ]: