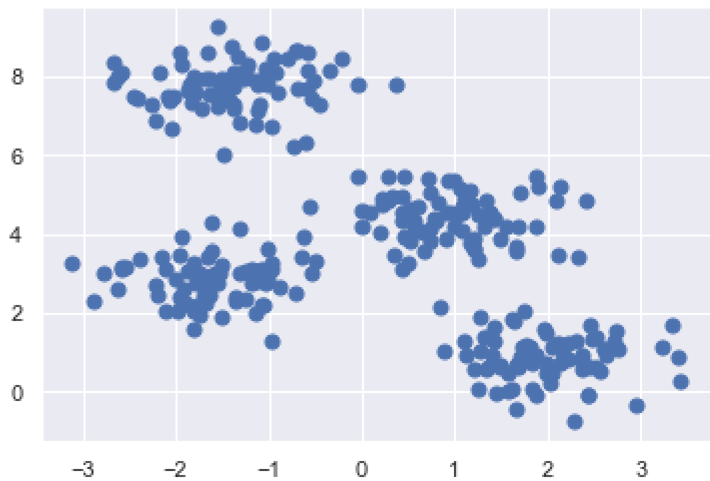
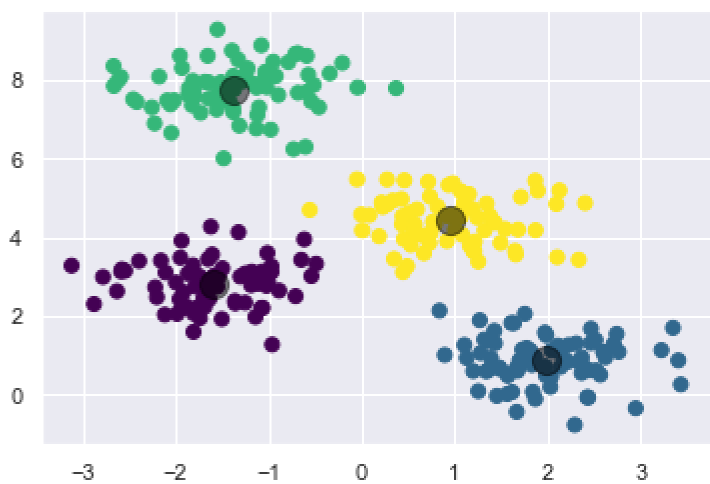
In [2]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns; sns.set()  # for plot styling
import numpy as np
```

In [9]:
```
from sklearn.datasets import make_blobs
X, y_true = make_blobs(n_samples=300, centers=4,
                       cluster_std=0.60, random_state=0)
plt.scatter(X[:, 0], X[:, 1], s=50);
```



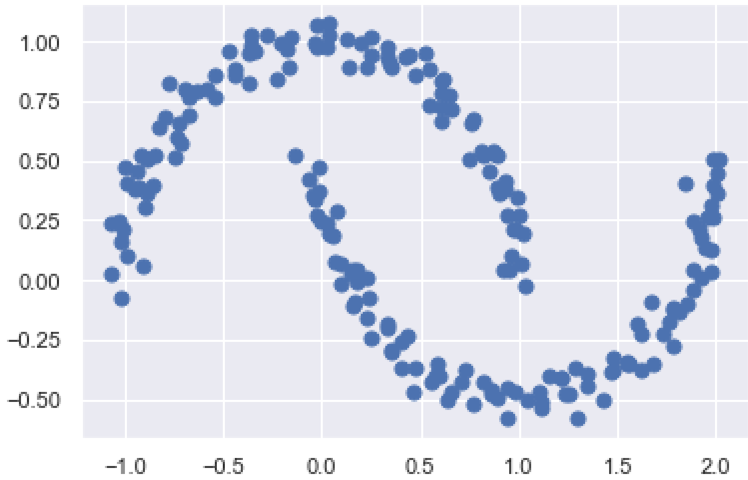
In [11]:
```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4)
kmeans.fit(X)
y_kmeans = kmeans.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, s=50, cmap='viridis')

centers = kmeans.cluster_centers_
plt.scatter(centers[:, 0], centers[:, 1], c='black', s=200, alpha=0.5);
```

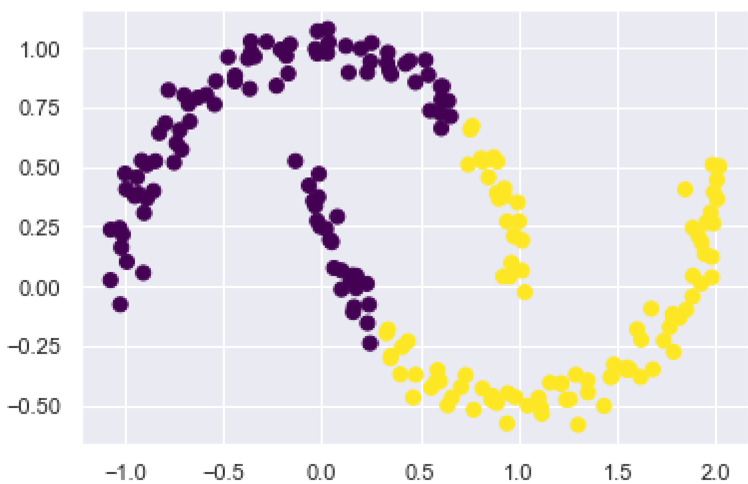


In [12]:
```
from sklearn.datasets import make_moons
X, y = make_moons(200, noise=.05, random_state=0)
```

In [15]:
```
plt.scatter(X[:, 0], X[:, 1], s=50);
```



In [16]:
```
labels = KMeans(2, random_state=0).fit_predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels,
            s=50, cmap='viridis');
```



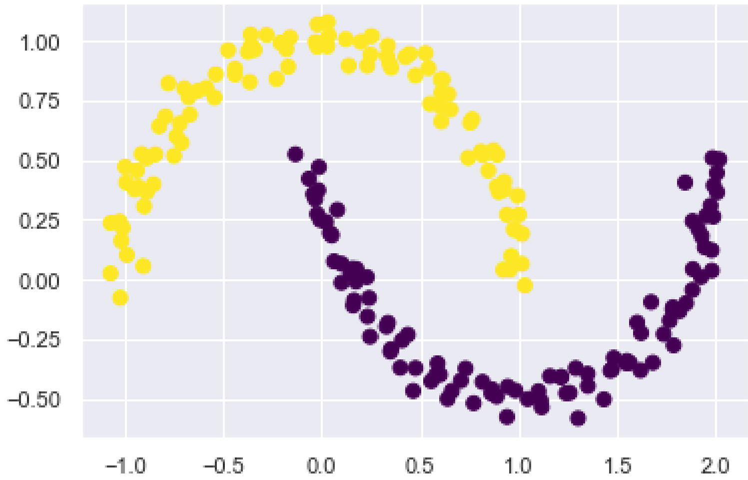
In [18]:
```
#This situation is reminiscent of the discussion in In-Depth: Support Vector Machines,
#where we used a kernel transformation to project the data into a higher dimension wher
#We might imagine using the same trick to allow k-means to discover non-linear boundari
```

In [19]:
```
#SpectralClustering estimator is an expanded version on K-means.
#It uses the graph of nearest neighbors to compute a higher-dimensional representation
#and then assigns labels using a k-means algorithm
```

In [21]:
```
from sklearn.cluster import SpectralClustering
model = SpectralClustering(n_clusters=2, affinity='nearest_neighbors',
                           assign_labels='kmeans')
labels = model.fit_predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels,
            s=50, cmap='viridis');
```

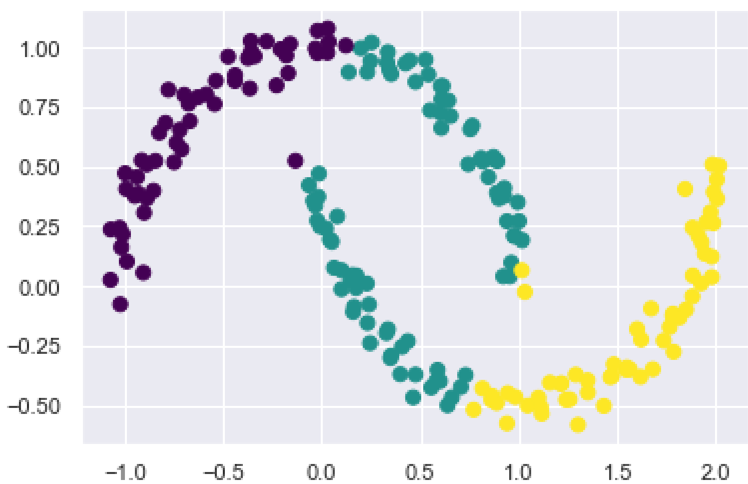
```
C:\Users\tabkh\anaconda3\lib\site-packages\sklearn\manifold\_spectral_embedding.py:245:
```

```
UserWarning: Graph is not fully connected, spectral embedding may not work as expected.
  warnings.warn("Graph is not fully connected, spectral embedding"
```



In [25]:

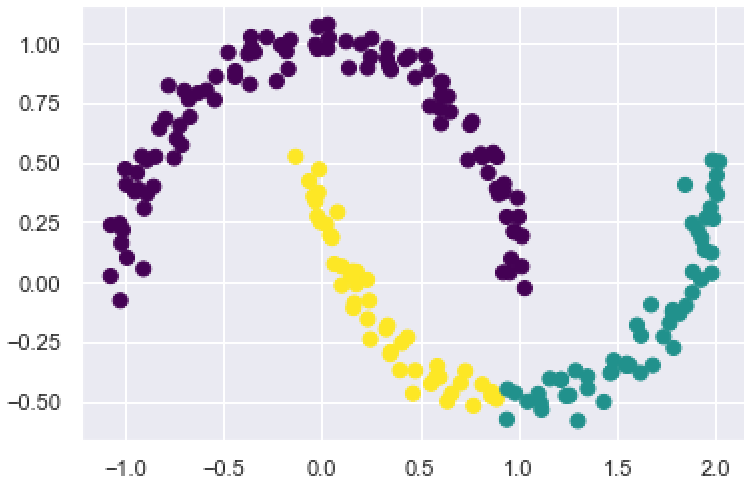
```
labels = KMeans(3, random_state=0).fit_predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels,
            s=50, cmap='viridis');
```



In [26]:

```
model = SpectralClustering(n_clusters=3, affinity='nearest_neighbors',
                           assign_labels='kmeans')
labels = model.fit_predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels,
            s=50, cmap='viridis');
```

```
C:\Users\tabkh\anaconda3\lib\site-packages\sklearn\manifold\_spectral_embedding.py:245:
UserWarning: Graph is not fully connected, spectral embedding may not work as expected.
  warnings.warn("Graph is not fully connected, spectral embedding"
```




```
In [ ]:
```