In [1]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [4]:

```python
from sklearn.datasets import load_breast_cancer

cancer=load_breast_cancer()

df=pd.DataFrame(cancer['data'],columns=cancer['feature_names'])

df.head()
```

Out[4]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst radius	worst texture	worst perimeter	worst area
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	...	25.38	17.33	184.60	2019.0
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	...	24.99	23.41	158.80	1956.0
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	...	23.57	25.53	152.50	1709.0
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	...	14.91	26.50	98.87	567.7
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	...	22.54	16.67	152.20	1575.0

5 rows × 30 columns

In [5]:

```python
from sklearn.preprocessing import StandardScaler

scaler=StandardScaler()

scaler.fit(df)
scaled_data=scaler.transform(df)
```

In [6]:

```python
scaled_data
```

Out[6]:

```
array([[ 1.09706398, -2.07333501,  1.26993369, ...,  2.29607613,
         2.75062224,  1.93701461],
       [ 1.82982061, -0.35363241,  1.68595471, ...,  1.0870843 ,
        -0.24388967,  0.28118999],
       [ 1.57988811,  0.45618695,  1.56650313, ...,  1.95500035,
         1.152255  ,  0.20139121],
       ...,
       [ 0.70228425,  2.0455738 ,  0.67267578, ...,  0.41406869,
        -1.10454895, -0.31840916],
       [ 1.83834103,  2.33645719,  1.98252415, ...,  2.28998549,
         1.91908301,  2.21963528],
       [-1.80840125,  1.22179204, -1.81438851, ..., -1.74506282,
        -0.04813821, -0.75120669]])
```

In [18]:

```python
from sklearn.decomposition import PCA

pca=PCA(n_components=2)
pca.fit(scaled_data)
x_pca=pca.transform(scaled_data)
x_pca.shape
```

Out[18]:

```
(569, 2)
```

In [19]:

```python
x_pca
```

Out[19]:

```
array([[ 9.19283683,  1.94858307],
       [ 2.3878018 , -3.76817174],
       [ 5.73389628, -1.0751738 ],
       ...,
       [ 1.25617928, -1.90229671],
       [10.37479406,  1.67201011],
       [-5.4752433 , -0.67063679]])
```

In [20]:

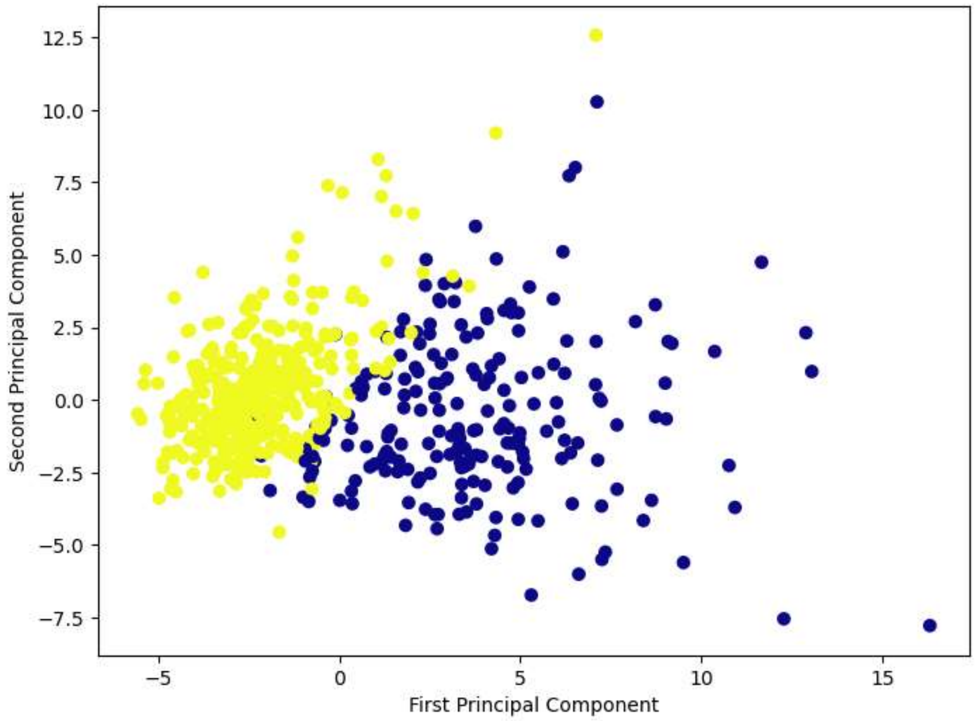
```python
plt.figure(figsize=(8,6))

plt.scatter(x_pca[:,0],x_pca[:,1],c=cancer['target'],cmap='plasma')

plt.xlabel('First Principal Component')
plt.ylabel('Second Principal Component')
```

Out[20]:

```
Text(0, 0.5, 'Second Principal Component')
```



In [21]:

```python
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
names=['sepal-length','sepal-width','petal-length','petal-width','Class']
dataset=pd.read_csv(url,names=names)
```

In [22]:

```python
dataset
```

Out[22]:

	sepal-length	sepal-width	petal-length	petal-width	Class
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [23]:

```python
X=dataset.iloc[:,0:4].values
y=dataset.iloc[:,4].values
```

In [24]:

```python
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=1)
```

In [25]:

```python
sc=StandardScaler()
X_train=sc.fit_transform(X_train)
X_test=sc.transform(X_test)
```

In [29]:

```python
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA

lda = LDA(n_components=1)
X_train = lda.fit_transform(X_train, y_train)
X_test = lda.transform(X_test)
```

In [30]:
```
lda
```

Out[30]:
```
LinearDiscriminantAnalysis(n_components=1)
```

In [31]:
```
X_train
```

Out[31]:
```
array([[-2.1560854 ],
       [-6.11662239],
       [-0.90912129],
       [-6.26379402],
       [-6.44115272],
       [ 8.56164613],
       [-2.01994399],
       [ 8.31277554],
       [-2.13442116],
       [-6.07270723],
       [-5.24089883],
       [ 8.07646049],
       [-4.56862701],
       [-5.558475  ],
       [-1.40925096],
       [-8.69494915],
       [ 8.44562193],
       [ 8.31096247],
       [ 7.0599544 ],
       [ 1.87341062]
```

In [32]:
```
X_test
```

Out[32]:
```
array([[10.35675286],
       [ 0.78309218],
       [-1.00007779],
       [ 8.98976393],
       [-4.91813822],
       [-2.25040201],
       [-4.42164596],
       [ 6.92657595],
       [ 6.90332131],
       [-5.84849778],
       [-1.90327928],
       [ 7.95404718],
       [-5.89372605],
       [-1.49622857],
       [-2.38691789],
       [ 7.29721934],
       [-1.021476  ],
       [-2.77071396],
       [ 7.838289  ],
       [ 8.33103761]
```

In [33]:
```
from sklearn.ensemble import RandomForestClassifier

classifier=RandomForestClassifier(max_depth=2,random_state=0)
classifier.fit(X_train,y_train)
y_pred=classifier.predict(X_test)
```

In [34]:
```
y_pred
```

Out[34]:
```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-setosa', 'Iris-virginica', 'Iris-versicolor',
       'Iris-setosa', 'Iris-virginica', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-setosa', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-setosa', 'Iris-setosa', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-virginica', 'Iris-versicolor', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

In [35]:
```
from sklearn.metrics import accuracy_score

accuracy_score(y_test,y_pred)
```

Out[35]:
```
1.0
```

In [ ]:
```

```