**Name –DHANA KORANGA**

**Roll no. -2001056**

**# Splitting the dataset into the Training set and Test set**

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)

**# Feature Scaling**

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X_train = sc.fit_transform(X_train)

X_test = sc.transform(X_test)

**# Fitting Decision Tree Classification to the Training set**

from sklearn.tree import DecisionTreeClassifier

classifier = DecisionTreeClassifier(criterion = 'entropy', random_state = 0)

classifier.fit(X_train, y_train)

**# Predicting the Test set results**

y_pred = classifier.predict(X_test)

from sklearn.metrics import confusion_matrix

from sklearn.metrics import accuracy_score

**#printing the accuracy of Decision tree**

accuracy_score(y_test, y_pred)

OUPUT –

`0.9895833333333334`

# Fitting SVM to the Training set

```python
from sklearn.svm import SVC

classifier = SVC(kernel = 'linear', random_state = 0)

classifier.fit(X_train, y_train)
```

# Predicting the Test set results

```python
y_pred = classifier.predict(X_test)
```

#Accuracy of SVM

```python
accuracy_score(y_test, y_pred)
```

OUTPUT – `0.9739583333333334`

#fitting knn model

```python
from sklearn.neighbors import KNeighborsClassifier

classifier=KNeighborsClassifier(n_neighbors=5,metric='minkowski',p=2)

classifier.fit(X_train,y_train)
```

# Predicting the Test set results

```python
y_pred = classifier.predict(X_test)
```

# Predicting the Test set results

```python
y_pred = classifier.predict(X_test)
```

# Making the Confusion Matrix

```python
from sklearn.metrics import confusion_matrix
```

```
cm = confusion_matrix(y_test, y_pred)
```

**#Accuracy of knn**

```
accuracy_score(y_test, y_pred)
```

OUTPUT -

`0.875`