

```

# Train & Test Scores of Random Forest
print("Accuracy (Train) score of Random Forest
",rf.score(X_train,Y_train)*100)
print("Accuracy (Test) score of Random Forest ", rf.score(X_test,Y_test)*100)
print("Accuracy score of Random Forest ", accuracy_score(Y_test,rf_pred)*100)

```

Step 9: Building a Predictive System

```

input_data =
(197.07600,206.89600,192.05500,0.00289,0.00001,0.00166,0.00168,0.00498,0.0109
8,0.09700,0.00563,0.00680,0.00802,0.01689,0.00339,26.77500,0.422229,0.741367,
-7.348300,0.177551,1.743867,0.085569)

# changing input data to a numpy array
input_data_as_numpy_array = np.asarray(input_data)

# reshape the numpy array
input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

prediction = sv.predict(input_data_reshaped)
print(prediction)

if (prediction[0] == 0):
    print("The Person does not have Parkinsons Disease")

else:
    print("The Person has Parkinsons")

```

Step 10: Saving the trained model

```

import pickle
filename = 'parkinsons_model.sav'
pickle.dump(sv, open(filename, 'wb'))
# Loading the saved model
loaded_model = pickle.load(open('parkinsons_model.sav', 'rb'))
for column in X.columns:
    print(column)

```