Breast Cancer Prediction Project:

Data Collection And Analysis:

The dataset is loaded, explored, and missing values are checked.

Target variable 'diagnosis' is encoded using Label Encoder

1. **Loading the data set from the given csv file.**

   ```
   cancer_dataset = pd.read_csv("/content/data.csv"
   ```

2. **Display the first 5 rows of the dataset.**

   ```
   cancer_dataset.head()
   ```

3. **Print the shape (number of rows and columns) of the dataset.**

   ```
   cancer_dataset.shape
   ```

4. **Display statistical measures of the data.**

   ```
   cancer_dataset.describe()
   ```

5. **Check for missing values in the dataset.**

   ```
   cancer_dataset.isnull().sum()
   ```

Target variable Encoding:

```
label_encode = LabelEncoder()

labels = label_encode.fit_transform(cancer_dataset.diagnosis)

cancer_dataset["diagnosis"] = labels

cancer_dataset["diagnosis"].value_counts()
```

Encode the target variable 'diagnosis' using Label Encoder.

Display the count of each class in the 'diagnosis' column.

Data Standardization:

Used StandardScaler to standardize the training and test data.

```
scaler = StandardScaler()

scaler.fit(X_train)

X_train_standardized = scaler.transform(X_train)

X_test_standardized = scaler.transform(X_test)
```

Feature Selection:

No explicit feature selection is mentioned; the entire feature set is used.

Model Implementation:

SVM with a linear kernel is used for breast cancer prediction.

```
classifier = svm.SVC(kernel='linear')

classifier.fit(X_train, Y_train)
```

The model is trained and evaluated on both training and test sets.

```
training_data_accuracy = accuracy_score(classifier.predict(X_train), Y_train)

test_data_accuracy = accuracy_score(classifier.predict(X_test), Y_test)

print('Training Data Accuracy:', training_data_accuracy)

print('Test Data Accuracy:', test_data_accuracy)
```