EX.NO :

DATE :

IMPLEMENTING ARTIFICIAL NEURAL NETWORKS FOR AN APPLICATION USING PYTHON - REGRESSION

AIM :

To implementing artificial neural networks for an application in Regression using python.

SOURCE CODE :

```
from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.datasets import make_regression
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
X, y = make_regression(n_samples=1000, noise=0.05, n_features=100)
X.shape, y.shape = ((1000, 100), (1000,))
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=True, random_state
=42)
clf = MLPRegressor(max_iter=1000)
clf.fit(X_train, y_train)
print(f"R2 Score for Training Data = {clf.score(X_train, y_train)}")
print(f"R2 Score for Test Data = {clf.score(X_test, y_test)}")
```

OUTPUT :

```
R2 Score for Training Data = 0.9999960757303987
R2 Score for Test Data = 0.9620311946670963
/usr/local/lib/python3.10/dist-packages/sklearn/neural_network/_multilayer_perceptron.py:690: ConvergenceWarning: Stochastic Optimizer: Maximum iterations (1000) reached and the optimization hasn't converged yet.
  warnings.warn(
```

RESULT :

Thus Program is Executed Successfully And Output is Verified.