Linear regression by using Deep Neural network: Implement Boston housing price prediction problem by Linear regression using Deep Neural network. Use the Boston House price prediction dataset.

```
import numpy as np
from sklearn.datasets import fetch_openml
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.optimizers import Adam

# Load the Boston Housing dataset
boston = fetch_openml(name='boston', version=1, as_frame=True)

# Split the dataset into features and target
X = boston.data
y = boston.target

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Scale the data for better convergence
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Define the model
model = Sequential()
model.add(Dense(1, input_shape=(X_train.shape[1],)))

# Compile the model
model.compile(optimizer=Adam(learning_rate=0.001), loss='mean_squared_error')

# Train the model
model.fit(X_train, y_train, batch_size=32, epochs=100, verbose=1)

# Evaluate the model
loss = model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', loss)

# Make predictions
predictions = model.predict(X_test)

# Print some predictions and actual values
for i in range(10):
    print('Predicted:', predictions[i][0], 'Actual:', y_test.iloc[i])
```