**Implementation of XOR in Deep learning using python**

**Code:**

import numpy as np

from keras.models import Sequential

from keras.layers import Dense

from keras.optimizers import SGD

#input data

X = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])

y = np.array([[0], [1], [1], [0]])

#neural network model

model = Sequential()

model.add(Dense(2, input\_dim=2, activation='sigmoid')) # First hidden layer with 2 units and sigmoid activation

model.add(Dense(1, activation='sigmoid')) # Output layer with 1 unit and sigmoid activation

# Compile the model

model.compile(loss='mean\_squared\_error', optimizer=SGD(lr=0.1))

# Train the model

model.fit(X, y, epochs=10000, verbose=0)

# Evaluate the model

predictions = model.predict(X).round().astype(int)

# Print the results

print(predictions)

**Output:**

