NAME : PAREENITA A.SHIRSATH PRN : 221101062 B.E.A.I.&.D.S. ROLL.NO : 57

**DEEP LEARNING EXPERIMENT NO : 2**

CODE:

import os  
os.environ['TF\_CPP\_MIN\_LOG\_LEVEL'] = '2' *# Hide INFO and DEBUG logs; show only warnings and errors*import numpy as np  
import tensorflow as tf  
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Dense, Input  
  
*# 1. Define the XOR input and output data*X = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])  
y = np.array([[0], [1], [1], [0]])  
  
*# 2. Build the neural network model*model = Sequential([  
 Input(shape=(2,)), *# Explicit Input layer with shape matching your input* Dense(2, activation='relu'), *# Hidden layer with 2 neurons and relu activation* Dense(1, activation='sigmoid') *# Output layer with 1 neuron and sigmoid activation*])  
  
*# 3. Compile the model*model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])  
  
*# 4. Train the model*model.fit(X, y, epochs=5000, verbose=0)  
  
*# 5. Evaluate the model*loss, accuracy = model.evaluate(X, y, verbose=0)  
print(f"Loss: {loss}, Accuracy: {accuracy}")  
  
*# 6. Make predictions*predictions = model.predict(X)  
print("Predictions:")  
print(predictions.round())

OUTPUT:

