```python
def main():
    #Get Training Data
    input = np.array([ [0, 0], [0, 1], [1, 0], [1, 1] ])
    output = np.array([ 0, 1, 1, 0 ])

    #Create Model Structure
    model = Sequential()

    model.add(Dense(2))
    model.add(Dense(32, activation= "relu"))
    model.add(Dense(64, activation= "relu"))
    model.add(Dense(1, activation= "sigmoid"))

    model.compile(optimizer="sgd", loss="binary_crossentropy", metrics=
      ["accuracy"])

    #Train model with the data
    model.fit(input, output, batch_size=2, epochs=1_000, verbose=1)

    #Make Prediction
    print("[%i, %i] = %i" % (0,0, Predict(model, 0,0)))
    print("[%i, %i] = %i" % (0,1, Predict(model, 0,1)))
    print("[%i, %i] = %i" % (1,0, Predict(model, 1,0)))
    print("[%i, %i] = %i" % (1,1, Predict(model, 1,1)))

def Predict(model: Sequential, p: int, q: int) -> int:
    prediction = model.predict(np.array([[p, q]]))
    predictedValue = prediction[0][0]
    roundedPredictedValue = int(round(predictedValue))

    return roundedPredictedValue
```