# Ex No: 6 IMPLEMENT BAYESIAN NETWORKS

**Install Packages** pip install pgmpy pip install network

# Program

from pgmpy.models import BayesianNetwork from pgmpy.factors.discrete import TabularCPD import networkx as nx

import pylab as plt

# Defining Bayesian Structure

model = BayesianNetwork([('Guest', 'Host'), ('Price', 'Host')]) # Defining the CPDs:

cpd\_guest = TabularCPD('Guest', 3, [[0.33], [0.33], [0.33]])

cpd\_price = TabularCPD('Price', 3, [[0.33], [0.33], [0.33]])

cpd\_host = TabularCPD('Host', 3, [[0, 0, 0, 0, 0.5, 1, 0, 1, 0.5],

[0.5, 0, 1, 0, 0, 0, 1, 0, 0.5],

[0.5, 1, 0, 1, 0.5, 0, 0, 0, 0]],

evidence=['Guest', 'Price'], evidence\_card=[3, 3])

# Associating the CPDs with the network structure. model.add\_cpds(cpd\_guest, cpd\_price, cpd\_host) model.check\_model()

# Output:

True

# Program

# Infering the posterior probability

from pgmpy.inference import VariableElimination infer = VariableElimination(model)

posterior\_p = infer.query(['Host'], evidence={'Guest': 2, 'Price': 2}) print(posterior\_p)

# Output:

+ + +

| Host | phi(Host) |

+=====+==== ===+

| Host(0) | 0.5000 |

+ + +

| Host(1) | 0.5000 |

+ + +

| Host(2) | 0.0000 |

+ + +

# Program

nx.draw(model, with\_labels=True) plt.savefig('model.png')

plt.close()

# Output: