

In []:

```
1 Program-7:
2 Write a program to construct a Bayesian network considering medical data.
3 Use this model to demonstrate the diagnosis of heart patients using
4 standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
```

In [1]:

```
1 import numpy as np
2 import pandas as pd
3 import csv
4 from pgmpy.estimators import MaximumLikelihoodEstimator
5 from pgmpy.models import BayesianModel
6 from pgmpy.inference import VariableElimination
```

In [2]:

```
1 heartDisease = pd.read_csv('heart.csv')
2 heartDisease = heartDisease.replace('?', np.nan)
3
4 print('Sample instances from the dataset are given below')
5 print(heartDisease)
```

Sample instances from the dataset are given below

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	\
0	63	1	1	145	233	1	2	150	0	2.3	
1	67	1	4	160	286	0	2	108	1	1.5	
2	67	1	4	120	229	0	2	129	1	2.6	
3	37	1	3	130	250	0	0	187	0	3.5	
4	41	0	2	130	204	0	2	172	0	1.4	
..	
298	45	1	1	110	264	0	0	132	0	1.2	
299	68	1	4	144	193	1	0	141	0	3.4	
300	57	1	4	130	131	0	0	115	1	1.2	
301	57	0	2	130	236	0	2	174	0	0.0	
302	38	1	3	138	175	0	0	173	0	0.0	

	slope	ca	thal	heartdisease
0	3	0	6	0
1	2	3	3	2
2	2	2	7	1
3	3	0	3	0
4	1	0	3	0
..
298	2	0	7	1
299	2	2	7	2
300	2	1	7	3
301	2	1	3	1
302	1	NaN	3	0

[303 rows x 14 columns]

In [3]:



```
1 print('\n Attributes and datatypes')
2 print(heartDisease.dtypes)
```

```
Attributes and datatypes
age                int64
sex                int64
cp                int64
trestbps          int64
chol              int64
fbs              int64
restecg          int64
thalach          int64
exang            int64
oldpeak         float64
slope            int64
ca              object
thal            object
heartdisease     int64
dtype: object
```

In [4]:



```
1 model= BayesianModel([('age','heartdisease'),('sex','heartdisease'),('exang','heartdisease'),
2                        ('heartdisease','restecg'),('heartdisease','chol')])
3 print('\nLearning CPD using Maximum likelihood estimators')
4 model.fit(heartDisease,estimator=MaximumLikelihoodEstimator)
```

Learning CPD using Maximum likelihood estimators

In [5]:



```
1 print('\n Inferencing with Bayesian Network:')
2 HeartDiseasetest_infer = VariableElimination(model)
```

Inferencing with Bayesian Network:


```
1 print('\n 1. Probability of HeartDisease given evidence= restecg')
2 q1=HeartDiseasetest_infer.query(variables=['heartdisease'],evidence={'restecg':1})
3 print(q1)
```

1. Probability of HeartDisease given evidence= restecg

```
Finding Elimination Order: : 100%|██████████  
██████████ | 5/5 [00:00<00:00, 2500.18it/s]  
Eliminating: chol: 100%|██████████  
██████████ | 5/5 [00:00<00:00, 102.03it/s]
```

heartdisease	phi(heartdisease)
heartdisease(0)	0.1012
heartdisease(1)	0.0000
heartdisease(2)	0.2392
heartdisease(3)	0.2015
heartdisease(4)	0.4581


```
1 print('\n 2. Probability of HeartDisease given evidence= cp ')
2 q2=HeartDiseasetest_infer.query(variables=['heartdisease'],evidence={'cp':2})
3 print(q2)
```

2. Probability of HeartDisease given evidence= cp

[illegible]

heartdisease	phi(heartdisease)
heartdisease(0)	0.3610
heartdisease(1)	0.2159
heartdisease(2)	0.1373
heartdisease(3)	0.1537
heartdisease(4)	0.1321