Code:

```
import csv
import numpy as np
with open("Restaurent.csv") as f:
    dataset=csv.reader(f)
    restuarent=list(dataset)
print("Restaurent dataset \n",restuarent[0])

def calculate_entropy_Info_Class2(feature,index):
    yy=0
    yn=0
    ny=0
    nn=0
    for i in restuarent[1:]:
        feature=i[index]
        willwait=i[10]
        if feature=="yes" and willwait=="yes":
            yy+=1
        elif feature=="yes" and willwait=="no":
            yn+=1
        elif feature=="no" and willwait=="yes":
            ny+=1
        elif feature=="no" and willwait=="no":
            nn+=1
    return [(yy,yn),(ny,nn)]
def calculate_entropy_Info_Class3(feature,index):
    Sy=0
    Sn=0
    Fy=0
    Fn=0
    Ny=0
    Nn=0
    feature2=["some","full","none"]
    feature3=["$$$","$$","$"]
    if feature=="Pat":
        feature1=feature2
    elif feature=="Price":
        feature1=feature3
    for i in restuarent[1:]:
        feature=i[index]
        willwait=i[10]
        if feature==feature1[0] and willwait=="yes":
            Sy+=1
        elif feature==feature1[0] and willwait=="no":
            Sn+=1
        elif feature==feature1[1] and willwait=="yes":
            Fy+=1
        elif feature==feature1[1] and willwait=="no":
            Fn+=1
        elif feature==feature1[2] and willwait=="yes":
            Ny+=1
        elif feature==feature1[2] and willwait=="no":
            Nn+=1
    return [(Sy,Sn),(Fy,Fn),(Ny,Nn)]
def calculate_entropy_Info_Class4(feature, index):
    Sy = 0
    Sn = 0
    Fy = 0
    Fn = 0
    Ny = 0
    Nn = 0
    Iy = 0
    In = 0
    feature2 = ["french", "thai", "burger", "italian"]
    feature3 = ["0-10", "10-30", "30-60", ">60"]
    if feature == "Type":
        feature1 = feature2
    elif feature == "Est":
        feature1 = feature3
    for i in restuarent[1:]:
        feature = i[index]
        willwait = i[10]
        if feature == feature1[0] and willwait == "yes":
            Sy += 1
        elif feature == feature1[0] and willwait == "no":
            Sn += 1
        elif feature == feature1[1] and willwait == "yes":
            Fy += 1
        elif feature == feature1[1] and willwait == "no":
            Fn += 1
        elif feature == feature1[2] and willwait == "yes":
            Ny += 1
        elif feature == feature1[2] and willwait == "no":
            Nn += 1
        elif feature == feature1[3] and willwait == "yes":
            Iy += 1
        elif feature == feature1[3] and willwait == "no":
            In += 1
    return [(Sy, Sn), (Fy, Fn), (Ny, Nn), (Iy, In)]
# class 2 calculation of entropy
Alt_Info=calculate_entropy_Info_Class2("Alt",0)
Bar_Info=calculate_entropy_Info_Class2("Bar",1)
Fri_Info=calculate_entropy_Info_Class2("Fri",2)
Hun_Info=calculate_entropy_Info_Class2("Hun",3)
Rain_Info=calculate_entropy_Info_Class2("rain",6)
Res_Info=calculate_entropy_Info_Class2("Res",7)
#class 3 calculation of entropy
Pat_Info=calculate_entropy_Info_Class3("Pat",4)
Price_Info=calculate_entropy_Info_Class3("Price",5)
#class 4 calculation of entropy
Type_Info=calculate_entropy_Info_Class4("Type",8)
Est_Info=calculate_entropy_Info_Class4("Est",9)
print(Alt_Info,Bar_Info,Hun_Info,Rain_Info,Res_Info)
print(Type_Info,Est_Info)
print(Pat_Info,Price_Info)
#step2 calculate entropy of classes
def calculateEntropyInstances(Info):
    entropy=[]
    for I in Info:
        total=I[0]+I[1]
        if I[0] != 0:
            phase1= (I[0]/total) *np.log(I[0]/total)
        else:
            phase1=0
        if I[1] != 0:
            phase2= (I[1]/total)*np.log(I[1]/total)
```

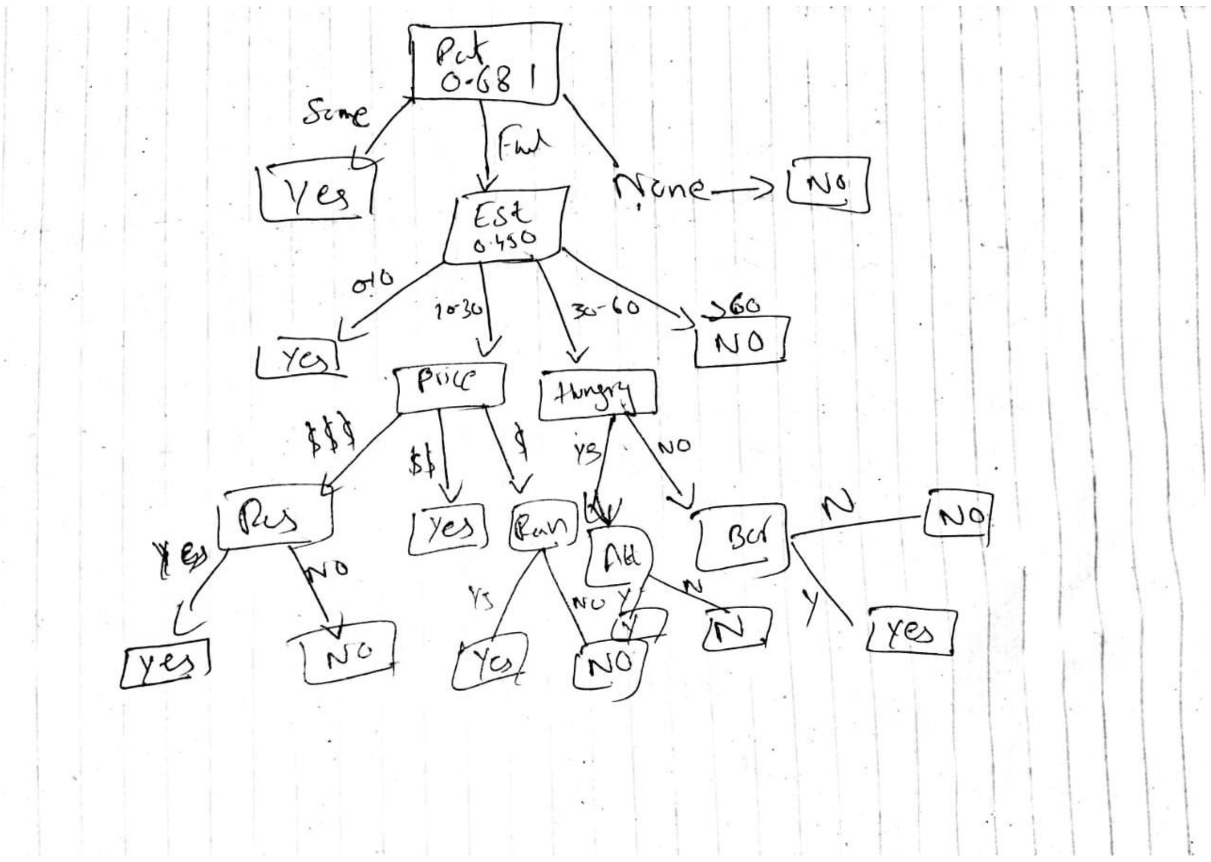
```python
        else:
            phase2=0
        total= -phase1-phase2
        entropy.append(total)
    return entropy

Alt_E=calculateEntropyInstances(Alt_Info)
Bar_E=calculateEntropyInstances(Bar_Info)
Hun_E=calculateEntropyInstances(Hun_Info)
Rain_E=calculateEntropyInstances(Rain_Info)
Res_E=calculateEntropyInstances(Res_Info)
Pat_E=calculateEntropyInstances(Pat_Info)
Price_E=calculateEntropyInstances(Price_Info)
Type_E=calculateEntropyInstances(Type_Info)
Est_E=calculateEntropyInstances(Est_Info)
print("Class wise  Entropy
Values:\n",Alt_E,Bar_E,Hun_E,Rain_E,Res_E,Pat_E,Price_E,Type_E,Est_E)

#step3 calculate entropy of features
def calculateEntropyFeature(Info,Alt_E):
    entropy=0
    def Total(Info):
        total=0
        for i in Info:
            total+=i[0]+i[1]
        return total
    for i in Info:
        sum=i[0]+i[1]
        ind=Info.index(i)
        v1=sum/Total(Info)
        v2=Alt_E[ind]
        value=v1*v2
        print(value)
        entropy += value
    return entropy

Alt_TE=calculateEntropyFeature(Alt_Info,Alt_E)
Bar_TE=calculateEntropyFeature(Bar_Info,Bar_E)
Hun_TE=calculateEntropyFeature(Hun_Info,Hun_E)
Rain_TE=calculateEntropyFeature(Rain_Info,Rain_E)
Res_TE=calculateEntropyFeature(Res_Info,Res_E)
Pat_TE=calculateEntropyFeature(Pat_Info,Pat_E)
Price_TE=calculateEntropyFeature(Price_Info,Price_E)
Type_TE=calculateEntropyFeature(Type_Info,Type_E)
Est_TE=calculateEntropyFeature(Est_Info,Est_E)
print("Colum wise Entropy
Values:\n",Alt_TE,Bar_TE,Hun_TE,Rain_TE,Res_TE,Pat_TE,Price_TE,Type_TE,Est_TE)
#step 4 Computing Info gain
def InfoGain(Alt_TE):
    Info_gain=1-Alt_TE
    return Info_gain

Alt_IG=InfoGain(Alt_TE)
Bar_IG=InfoGain(Bar_TE)
Hun_IG=InfoGain(Hun_TE)
Rain_IG=InfoGain(Rain_TE)
Res_IG=InfoGain(Res_TE)
Pat_IG=InfoGain(Pat_TE)
Price_IG=InfoGain(Price_TE)
```

```
Type_IG=InfoGain(Type_TE)
Est_IG=InfoGain(Est_TE)
print("Info Gain
Values:\n",Alt_IG,Bar_IG,Hun_IG,Rain_IG,Res_IG,Pat_IG,Price_IG,Type_IG,Est_IG)
```

Output:

```
Colum wise Entropy  Values:

 0.6931471805599453 0.6931471805599453 0.5574916031489784 0.6787845889957987
0.6787845889957987 0.3182570841474064 0.5574916031489784 0.6931471805599453
0.5493061443340548

Info Gain Values:

Alt: 0.3068528194400547

 Bar: 0.3068528194400547

 Hun: 0.4425083968510216

 rain: 0.32121541100420126

 Res: 0.32121541100420126

 Pat: 0.6817429158525936

 Price: 0.4425083968510216

 Type: 0.3068528194400547

 Est: 0.4506938556659452
```

Graph:



Pat 0.681
Some
Yes
Full
None
NO
Est 0.450
0-10
Yes
10-30
30-60
>60
NO
Price
Hungry
$$$
$$
$
YS
NO
Res
Yes
Ran
Alt
Bar
N
NO
Yes
NO
yes
NO
YS
NO
Yes
NO
Y
N
N
Y
yes