In [1]:

```
# EDA Analysis on DATASET
# Perform EDA on Haberman dataset
```

In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np


#how to add columns name if there is no colums name in dataset and how to read csv file
data_clm = ['age','Operation_Year','axil_nodes','Surv_status']
ds = pd.read_csv("haberman.csv",header=None,names = data_clm)
ds.head()
```

Out[1]:

	age	Operation_Year	axil_nodes	Surv_status
0	30	64	1	1
1	30	62	3	1
2	30	65	0	1
3	31	59	2	1
4	31	65	4	1

In [2]:

```
#data-points and features
ds.shape
ds['Surv_status'].value_counts()
```

Out[2]:

```
1    225
2     81
Name: Surv_status, dtype: int64
```

In [3]:

```
#column name of data set
ds.columns
```

Out[3]:

```
Index(['age', 'Operation_Year', 'axil_nodes', 'Surv_status'], dtype='object')
```

In [4]:

```
# data points for each class
ds['Surv_status'].value_counts()
# its a balanced dataset
```

Out[4]:

```
1    225
2     81
Name: Surv_status, dtype: int64
```

In [5]:

```
ds['survival'] = ds['Surv_status'].map({1:"yes",2:"no"})
del ds['Surv_status']
```

```
del ds['Surv_status']
ds.head()
```

Out[5]:

	age	Operation_Year	axil_nodes	survival
0	30	64	1	yes
1	30	62	3	yes
2	30	65	0	yes
3	31	59	2	yes
4	31	65	4	yes

In [6]:

```
ds.min()
```

Out[6]:

```
age               30
Operation_Year    58
axil_nodes         0
survival          no
dtype: object
```

In [71]:

```
survive = ds[ds['survival']=='yes']
not_survive = ds[ds['survival']=='no']
not_survive.head()
```

Out[71]:

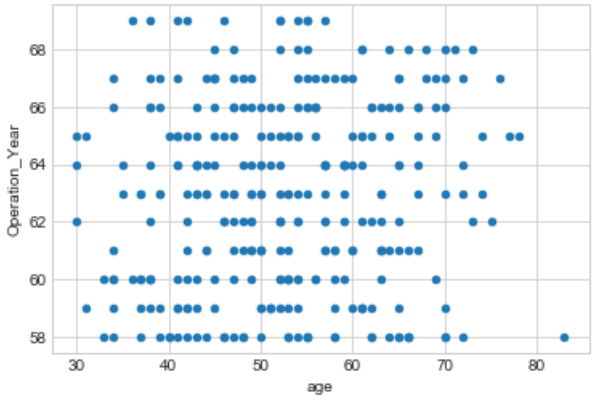
	age	Operation_Year	axil_nodes	survival
7	34	59	0	no
8	34	66	9	no
24	38	69	21	no
34	39	66	0	no
43	41	60	23	no

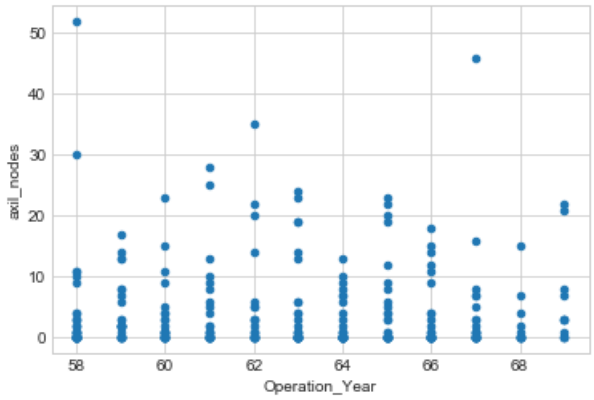
In [75]:

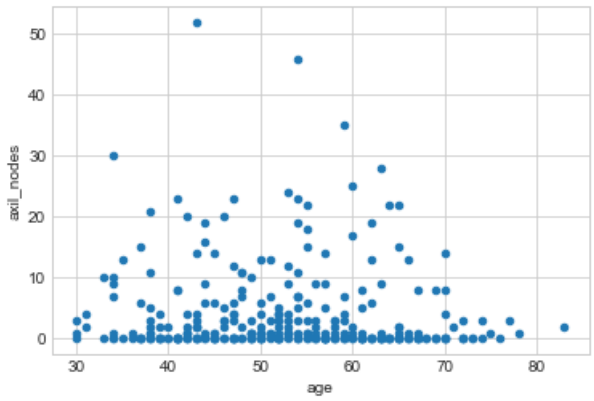
```
survival_age_lessthan_45 = ds[(ds['survival'] == 'yes') & (ds['age']<45)]
#survival_age_lessthan_45.count()
#survival_age_lessthan_45.head()
```

In [76]:

```
#2-D scatter plot
ds.plot(kind='scatter',x='age',y='Operation_Year');
plt.show()

ds.plot(kind='scatter',x='Operation_Year',y='axil_nodes');
plt.show()

ds.plot(kind='scatter',x='age',y='axil_nodes')
plt.show()


#Observations for 2d-scatter plot
#IN this we are not abe to reactify which point belongs to which class and also data are highly ov
erlapping each other
```

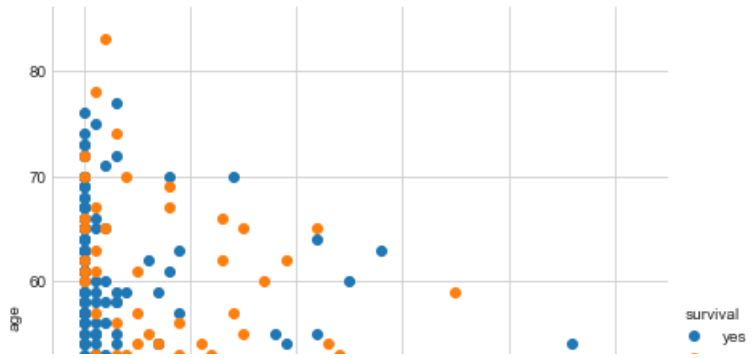


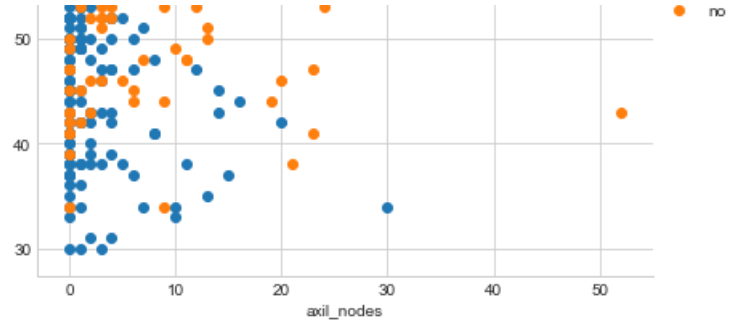




In [69]:

```
sns.set_style("whitegrid");
sns.FacetGrid(ds, hue="survival", size=6) \
   .map(plt.scatter, "axil_nodes", "age") \
   .add_legend();
plt.show();

#observations
#in this we are abe to rectify the class where it lies but here also we are not able to  saperate
the both classes just because
# data are overlapped but one thing we can observe is that max data points from both the classes l
ies between the axil_node of
# 0 - 5
#and if we talk about age this do not make sense peoperly
```



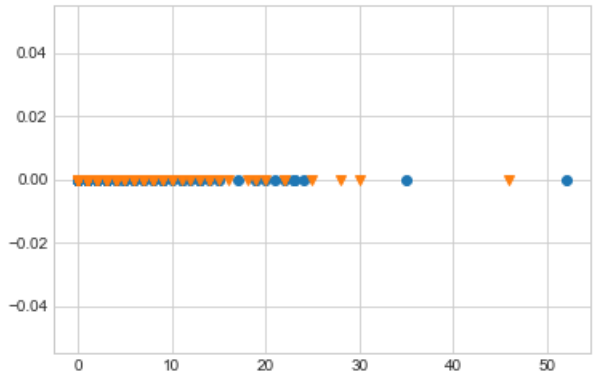


In [53]:

```
plt.close();
sns.set_style("whitegrid");
sns.pairplot(ds, hue="survival", size=3);
plt.show()

#observations
#here in 2d pair plot we are able to rectify the proper cluster or the proper visualization of the
data based on its class
#just because in this in each pair plot data are messhed up or getting over lapped with each able
so that we are not able to
#give the peoper conclusion that in which range or between which point we will get the perticular
class
```

In [79]:

```
#Histogram
import numpy as np

plt.plot(not_survive['axil_nodes'],np.zeros_like(not_survive['axil_nodes']),'o')
plt.plot(survive['axil_nodes'],np.zeros_like(survive['axil_nodes']),'v')

plt.show()
```



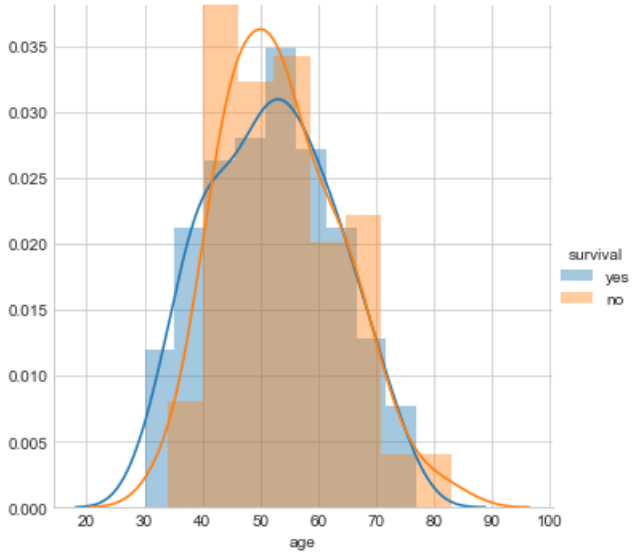
In [ ]:

```
# observation of above histogram
# by seeing above histogram plot it is hard to observe data just because data are higly overlapped
```

In [45]:

```
sns.FacetGrid(ds, hue="survival", size=5) \
   .map(sns.distplot, "age") \
   .add_legend();
plt.show();
```

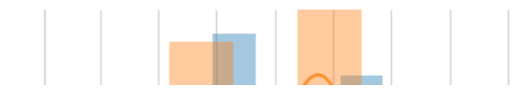
```
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

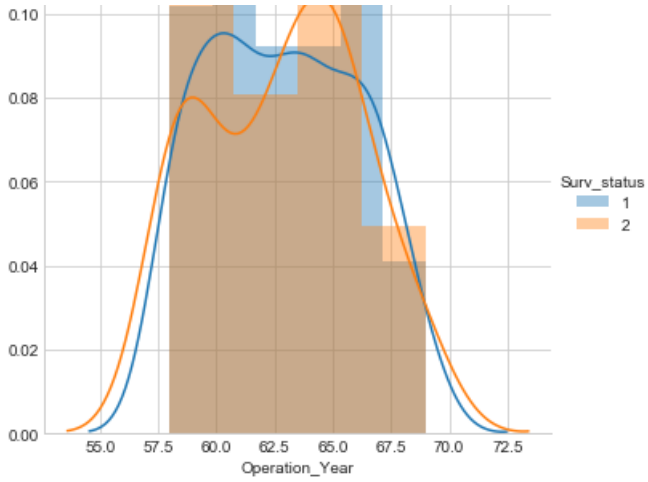


In [78]:

```
sns.FacetGrid(ds, hue="Surv_status", size=5) \
   .map(sns.distplot, "Operation_Year") \
   .add_legend();
plt.show();
```

```
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

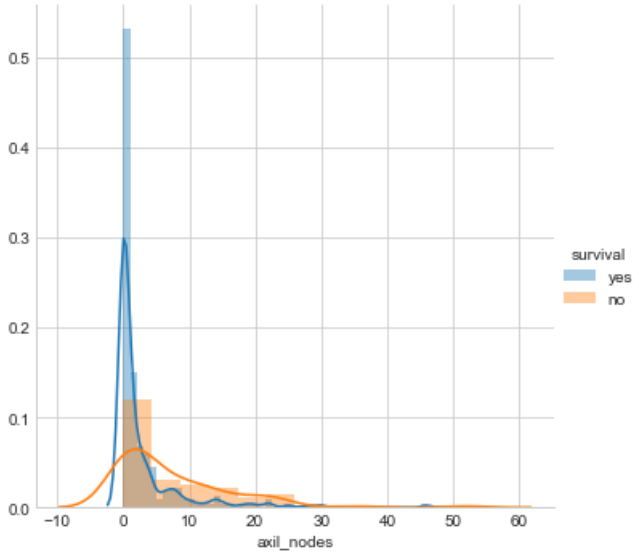




In [46]:

```
sns.FacetGrid(ds, hue="survival", size=5) \
   .map(sns.distplot, "axil_nodes") \
   .add_legend();
plt.show();
```

```
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Users\Nishant\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarning: The
'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



In [ ]:

```
#observation based on histohram
# by seeing histograms one this which we can observe is :
# 1. in age histogram the survival rate is more i.e (age < 35) approxmately and age between 35 - 8
5 death rate
# 2. in Operation year histogram  it is not clear just because data are highly overlaped
# 3. in axil_node histogram we see that in between 0-2 we are getting value for survival
```

In [107]:

```
#observation
#here we see that axil_node is less than 5 then there is more chance of survival
# survival = ds[(ds['survival'] == 'yes') & (ds['axil_nodes']<5)]
# survival.count()
survive = ds[ds['survival']=='yes']
not_survive = ds[ds['survival']=='no']
survive.head()
```

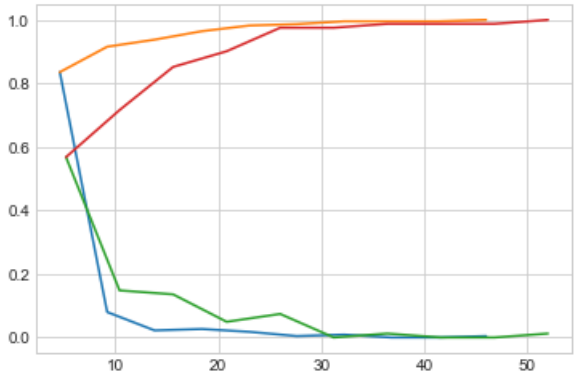
Out[107]:

	age	Operation_Year	axil_nodes	survival
0	30	64	1	yes
1	30	62	3	yes
2	30	65	0	yes
3	31	59	2	yes
4	31	65	4	yes

In [119]:

```
#Cumulative Distribution Function (CDF)

counts,bin_edges = np.histogram(survive['axil_nodes'],bins=10,density = True)

pdf = counts/(sum(counts))
print(pdf);
print(bin_edges);
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf);
plt.plot(bin_edges[1:],cdf);

counts,bin_edges = np.histogram(not_survive['axil_nodes'],bins=10,density = True)

pdf = counts/(sum(counts))
print(pdf);
print(bin_edges);
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf);
plt.plot(bin_edges[1:],cdf);


plt.show();
```

```
[0.83555556 0.08       0.02222222 0.02666667 0.01777778 0.00444444
 0.00888889 0.         0.         0.00444444]
[ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.8 41.4 46. ]
[0.56790123 0.14814815 0.13580247 0.04938272 0.07407407 0.
 0.01234568 0.         0.         0.01234568]
[ 0.   5.2 10.4 15.6 20.8 26.  31.2 36.4 41.6 46.8 52. ]
```



In [120]:

```
#Observations
#here we see that 82% of survival rate if axil_node is less than 8
#and here we also see that approx 96% of axil_node which isless than 25 with not survive rate
```

In [92]:

```
#mean ,Variance and std-dev
print('Mean');
print(np.mean(survive['axil_nodes']))
print(np.mean(not_survive['axil_nodes']))

print('\nstd');
```

```
print(np.std(survive['axil_nodes']))
print(np.std(not_survive['axil_nodes']))
```

```
Mean
2.7911111111111113
7.45679012345679

std
5.857258449412131
9.128776076761632
```

In [94]:

```
print('Median')
print(np.median(not_survive['axil_nodes']))
print(np.median(survive['axil_nodes']))

print('\n Quantiles');
print(np.percentile(survive['axil_nodes'],np.arange(0,100,25)))
print(np.percentile(survive['Operation_Year'],np.arange(0,100,25)))
print(np.percentile(survive['age'],np.arange(0,100,25)))

print('\n 90th Percentile')
print(np.percentile(survive['axil_nodes'],90))
print(np.percentile(survive['Operation_Year'],90))
print(np.percentile(survive['age'],90))
```

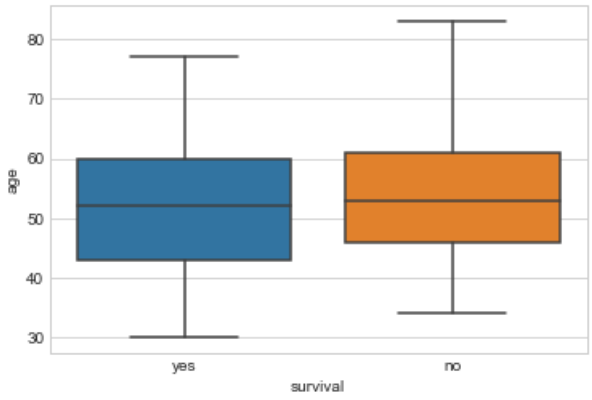
```
Median
4.0
0.0

 Quantiles
[0. 0. 0. 3.]
[58. 60. 63. 66.]
[30. 43. 52. 60.]

 90th Percentile
8.0
67.0
67.0
```

In [160]:

```
#observation
#here we see that if no of axil_nodes is 3 then 75% of survival
#here we see that 75% is survival rate above 5 years whos age is below 60

#here we see that if the no of axil_nodes less than 8 than 90% of survival rate is about 90%
```

In [97]:

```
#Box-plotes
sns.boxplot(x='survival',y='age',data=ds)
plt.show()
```



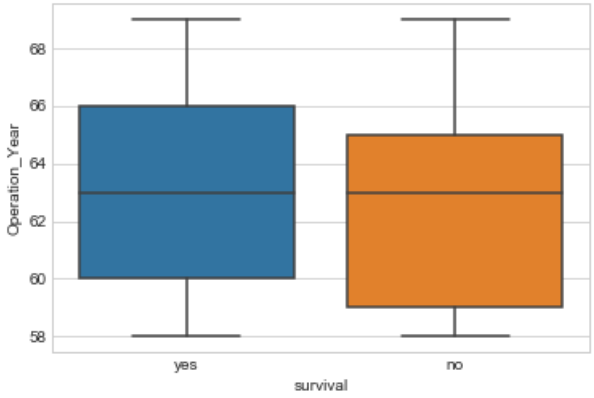
In [168]:

In [100]:

```
#observations
#here we see that 25% of survival  with age between 30 to 42
#here we can see that there is only age between 30 to 34 which has survival rate
```

In [121]:

```
sns.boxplot(x='survival',y='Operation_Year',data=ds)
plt.show()
```

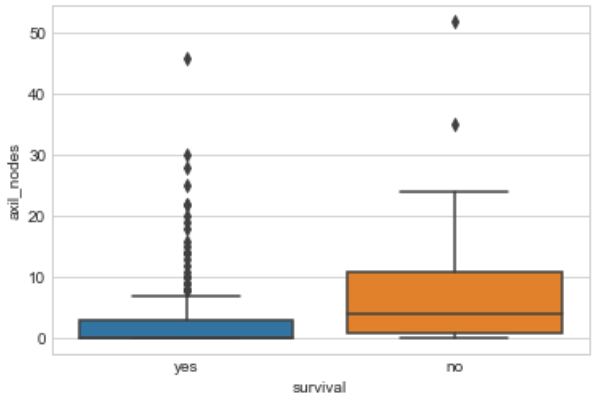


In [ ]:

```
#observation
#here it is difficult to prodict any conclusion just because data are highly overlapped
```

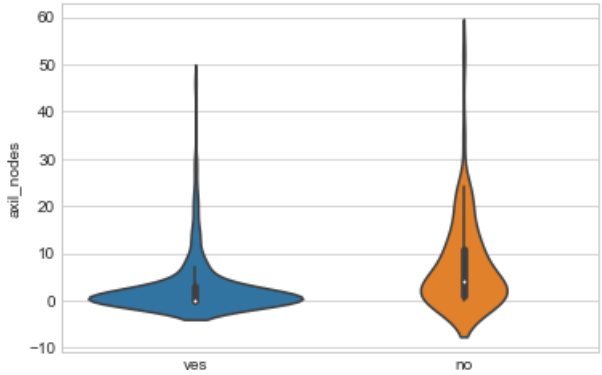
In [101]:

```
sns.boxplot(x='survival',y='axil_nodes',data=ds)
plt.show()
```



In [102]:

```
#violin plots

sns.violinplot(x="survival",y="axil_nodes",data=ds,size=8)
plt.show()
```



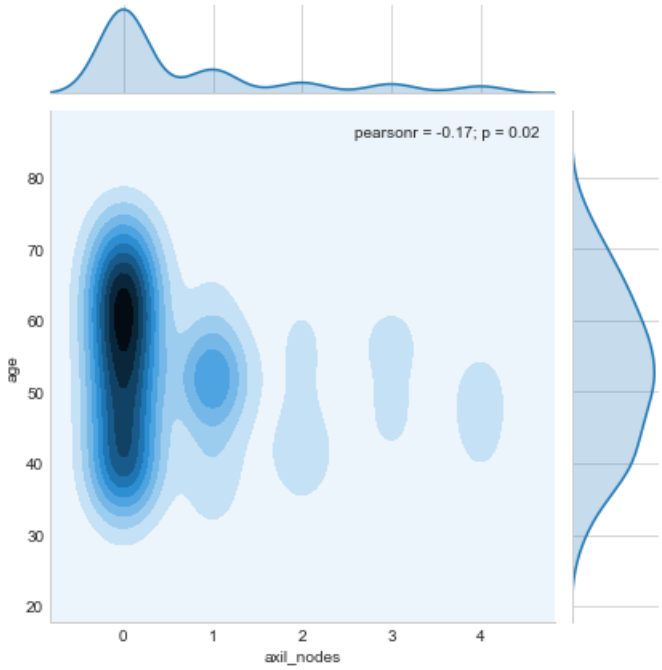
survival

In [103]:

```
#overall observations
#here we see that if no of axil_nodes is less than 3 then 75% of survival
#we see that 25% of survival with age between 30 to 42

#reason for ploting
#the violin plot shows the full distribution of the data,where statistics such as mean/median and
interquartile ranges
#in this we are able to see where
```

In [117]:

```
#2D Density plot, contors-plot
sns.jointplot(x="axil_nodes", y="age", data=survival, kind="kde");
plt.show();
```



In [ ]:

```
#above plot show  that in between age 58 to 68 most of the points lies with axil_node 0-2
#and apart from that points are spread 50-40 and remaining between 30-40
# here
```

In [ ]:

```
#Observation
#In this data set it is difficult to prodict/visualise the accurately just because
#the points from both the class and for the ever features overlaping each other so it is difficult
to visualise apart from that
#here are some conslusion from this dataset what i have observe
#from my observaion i this some home userful feature is axil_node and age which helps to visualise
some of the data so that
#we can say whether patient survive or not

#here we see that 82% of survival rate if axil_node is less than 8
#and here we also see that approx 96% of axil_node which isless than 25 with not survive rate

#here we see that 25% of survival  with age between 30 to 42
#here we can see that there is only age between 30 to 34 which has survival rate
```