Exploratory data analysis on Haberman dataset

1) Data Overview

The dataset contains information on the survival rates of patients who have been diagnosed with breast cancer.

Below are the various columns present in the dataset:

Age: Age of the patient
Year: Year in which the patient was diagnosed
Nodes: Number of axillary lymph nodes to which the cancer has spread
Status: Status column contains 2 values which tells us whether the patient has survived more than 5 years or not

1 - Patient survived 5 year or longer
2 - Patient died within 5 year

Objective: To find out if we can predict the survival chances of a patient based on the age, year and number of axillary lymph nodes

2) Basic Analysis

In [15]:
```
import pandas as pd # for file reading and dataframes
import matplotlib.pyplot as plot # for 2D plots
import seaborn as sns #for box, dist and violin plots
import warnings # to supress warnings

warnings.filterwarnings("ignore")

dtst=pd.read_csv('haberman.csv')
```

```
print(dtst.shape)
print(dtst.columns)
```

```
(306, 4)
Index(['age', 'year', 'nodes', 'status'], dtype='object')
```

In [16]:
```
print(dtst['status'].value_counts())
```

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

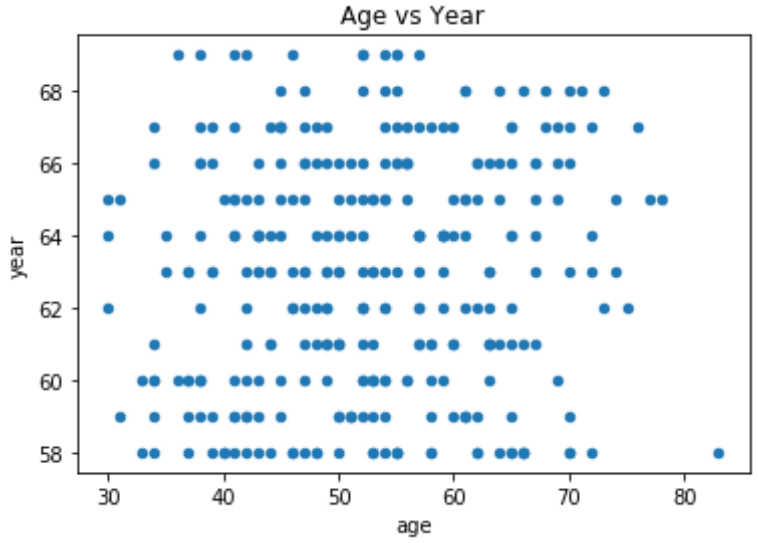
Observation:

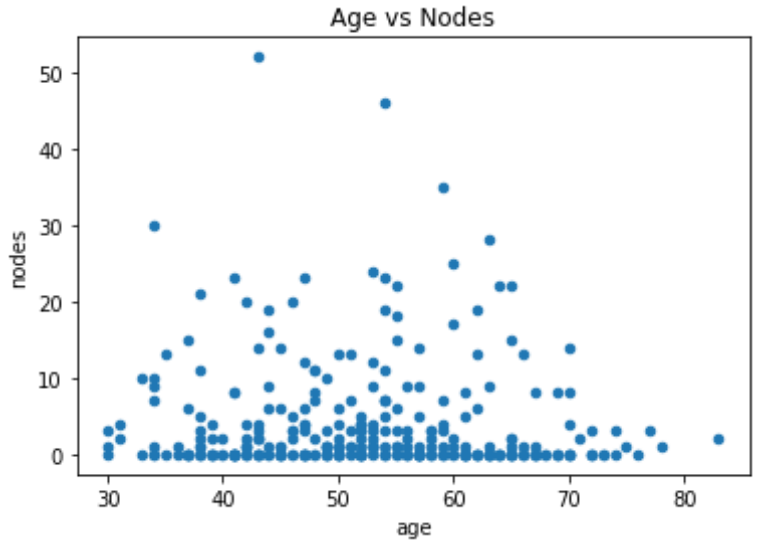
1. There are 306 rows and 4 columns in the dataset
2. The dataset is imbalanced as the data points in class 1 are almost 3 times the number in class 2

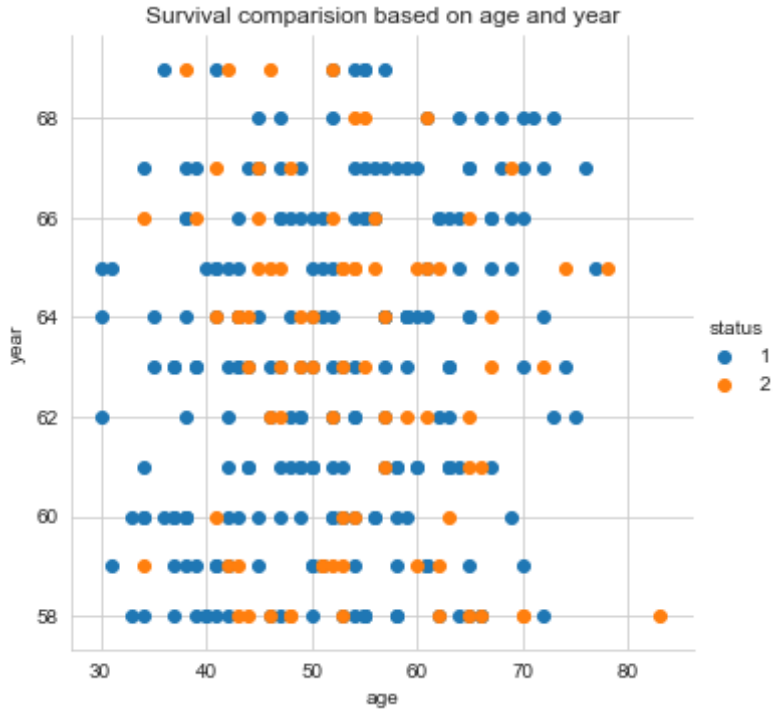
3) 2D Scatter Plots

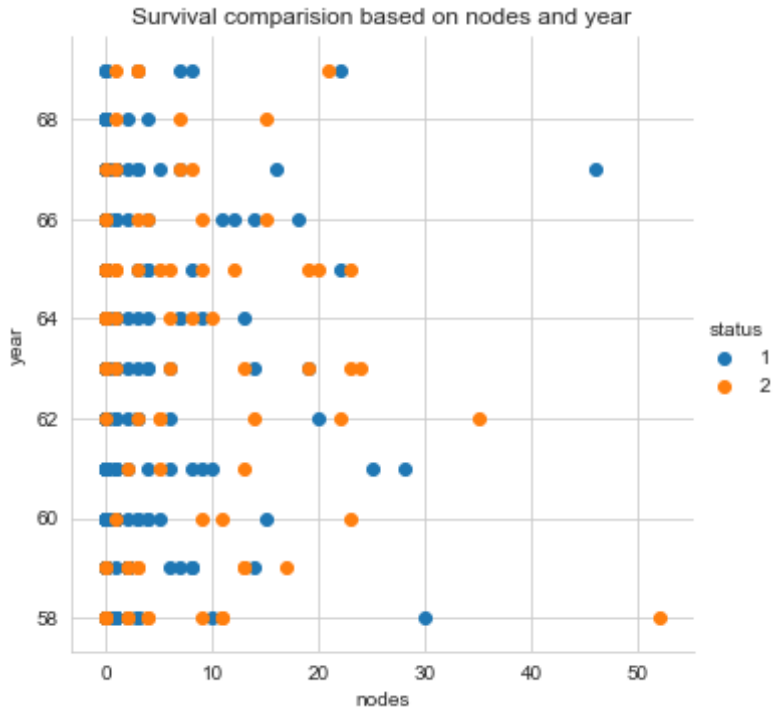
In [3]:
```
dtst.plot(kind='scatter',x='age',y='year');
plot.title('Age vs Year')
plot.show()

dtst.plot(kind='scatter',x='age',y='nodes');
plot.title('Age vs Nodes')
plot.show()

sns.set_style('whitegrid');
sns.FacetGrid(dtst,hue='status', height=5)\
    .map(plot.scatter, 'age','year').add_legend();
plot.title('Survival comparision based on age and year')
plot.show()

sns.set_style('whitegrid');
```

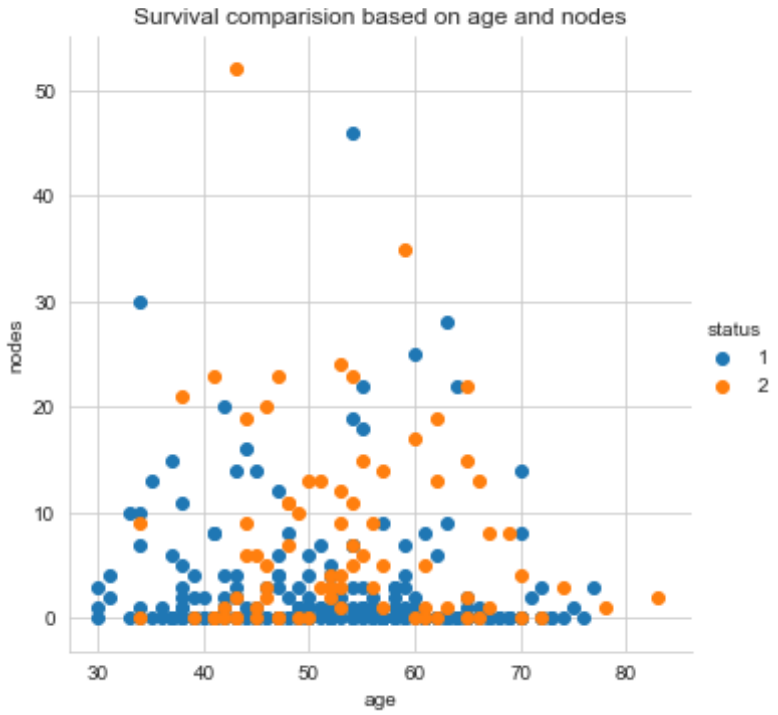
```
sns.FacetGrid(dtst,hue='status', height=5)\
    .map(plot.scatter, 'nodes','year').add_legend();
plot.title('Survival comparision based on nodes and year')
plot.show()

sns.set_style('whitegrid');
sns.FacetGrid(dtst,hue='status', height=5)\
    .map(plot.scatter, 'age','nodes').add_legend();
plot.title('Survival comparision based on age and nodes')
plot.show()
```









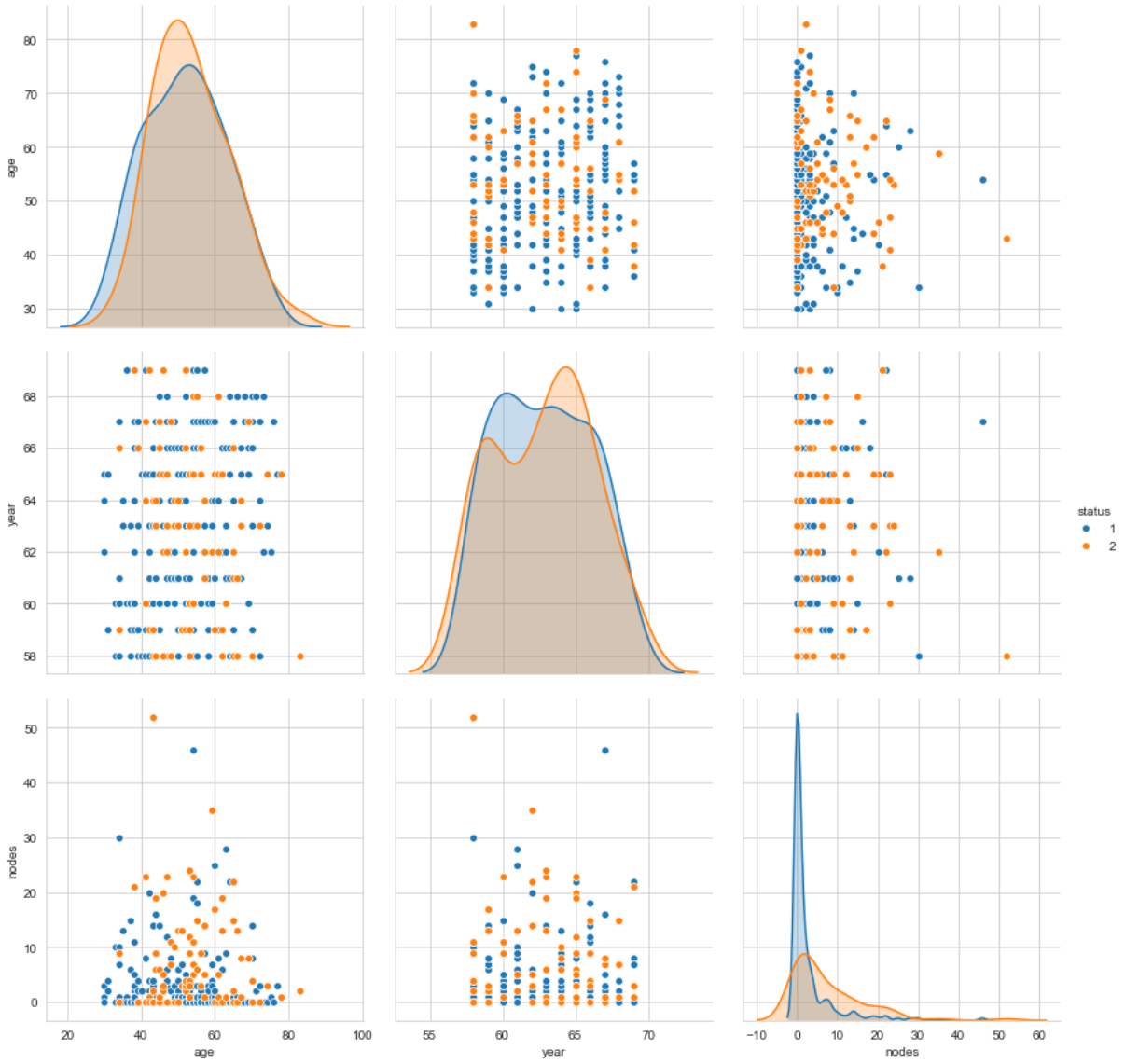


Observations from the above 2D scatter plots:

1. Most of the patients are aged between 30 and 80
2. The dataset contains patients info who were diagnosed between 1956 and 1970
3. Irrespective of the number of axillary nodes, patients below the age of 40 have higher chances of survival

4) Pair Plots

In [4]:
```
sns.pairplot(dtst,hue='status',vars=['age','year','nodes'],height=4)
plot.show()
```

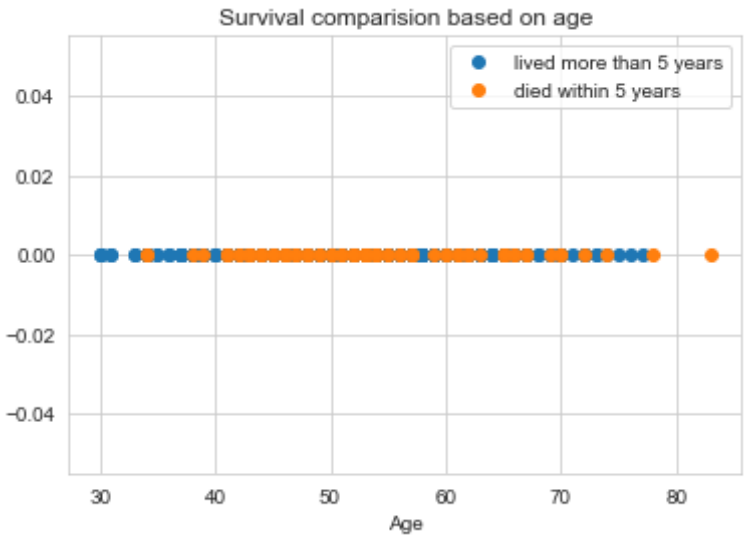


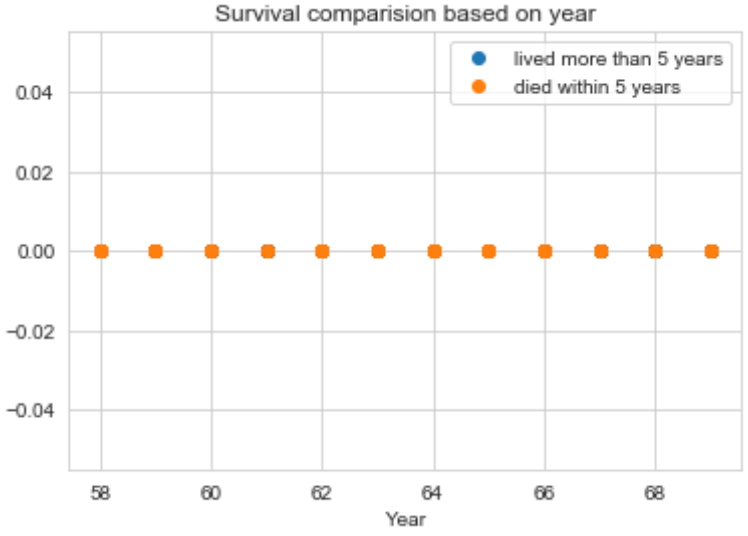
Observations from the above pair plots:

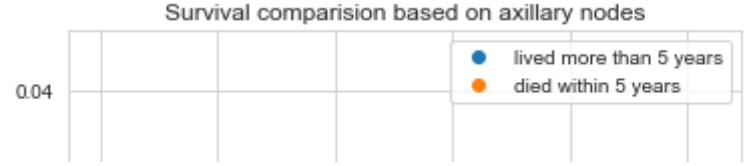
1. No major giveaway from the above pair plots as the data points are overlapping in all of the graphs

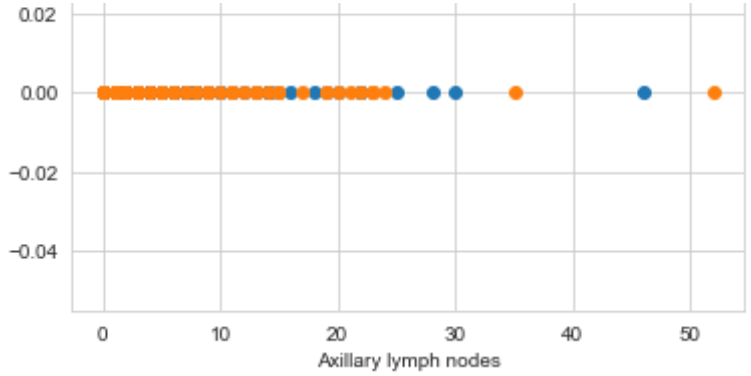
5) 1D Scatter Plots

In [5]:
```
import numpy as np

dtst_1=dtst[dtst.status==1] #patients who survived more than 5 years
dtst_2=dtst[dtst.status==2] #patients who died within 5 years

plot.plot(dtst_1['age'],np.zeros_like(dtst_1['age']),'o',label='lived m
ore than 5 years')
plot.plot(dtst_2['age'],np.zeros_like(dtst_2['age']),'o',label='died wi
thin 5 years')
plot.xlabel('Age')
plot.title('Survival comparision based on age')
plot.legend()
plot.show()

plot.plot(dtst_1['year'],np.zeros_like(dtst_1['year']),'o',label='lived
 more than 5 years')
plot.plot(dtst_2['year'],np.zeros_like(dtst_2['year']),'o',label='died
 within 5 years')
plot.xlabel('Year')
plot.title('Survival comparision based on year')
plot.legend()
plot.show()

plot.plot(dtst_1['nodes'],np.zeros_like(dtst_1['nodes']),'o',label='liv
ed more than 5 years')
plot.plot(dtst_2['nodes'],np.zeros_like(dtst_2['nodes']),'o',label='die
d within 5 years')
plot.xlabel('Axillary lymph nodes')
plot.title('Survival comparision based on axillary nodes')
plot.legend()
plot.show()
```







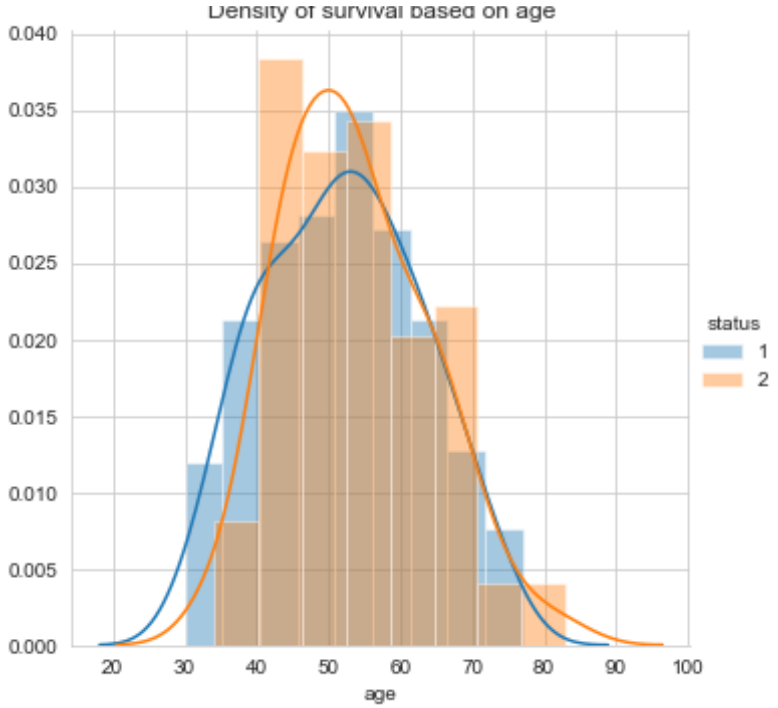


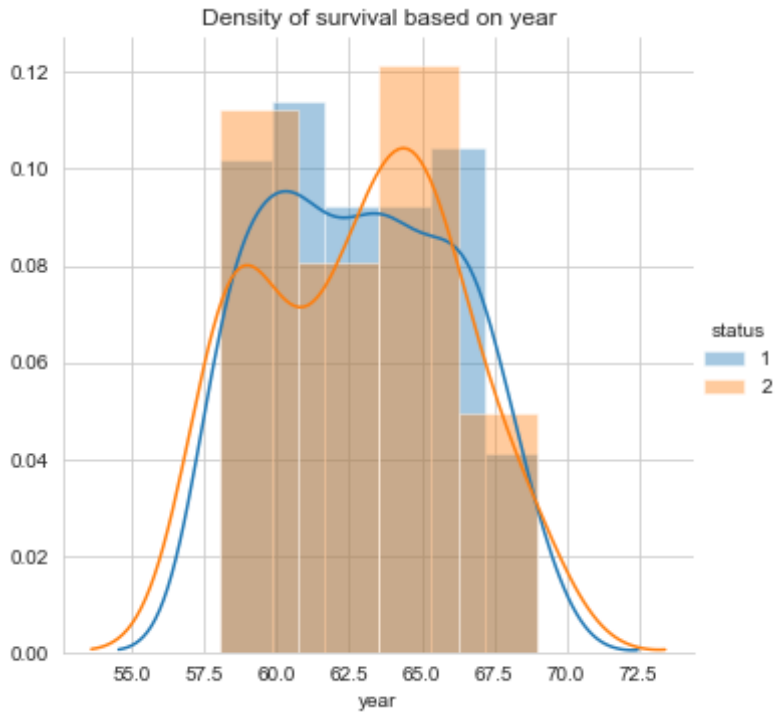
Observation from the above 1D scatter plots:

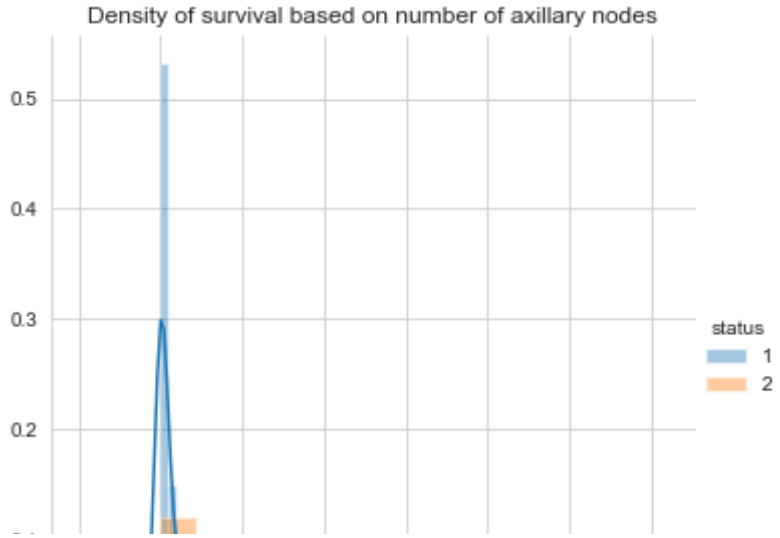
1. Most of the points overlap and there is no clear divison between the two classes

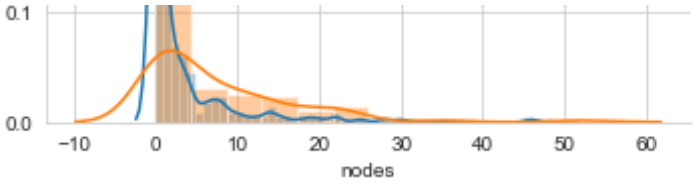
6) PDF - Histogram

In [6]:
```
sns.FacetGrid(dtst,hue='status',height=5)\
    .map(sns.distplot,'age').add_legend();
plot.title('Density of survival based on age')
plot.show()

sns.FacetGrid(dtst,hue='status',height=5)\
    .map(sns.distplot,'year').add_legend();
plot.title('Density of survival based on year')
plot.show()

sns.FacetGrid(dtst,hue='status',height=5)\
    .map(sns.distplot,'nodes').add_legend();
plot.title('Density of survival based on number of axillary nodes')
plot.show()
```









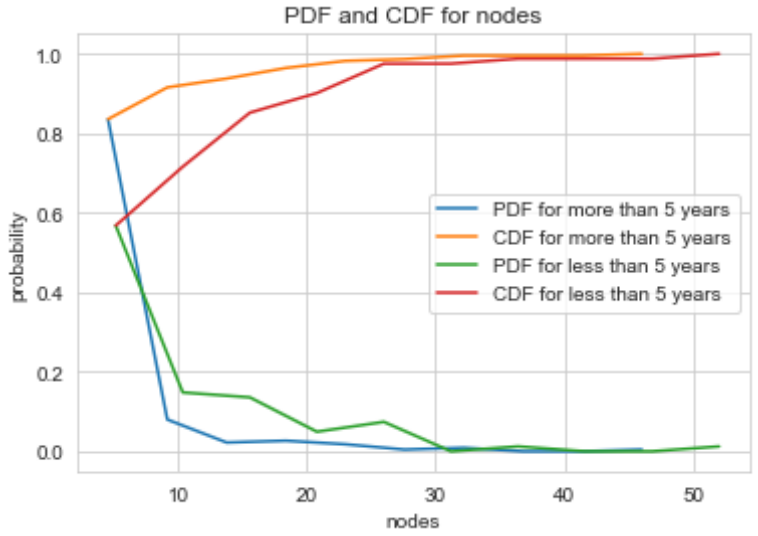
Observations from the above histograms:

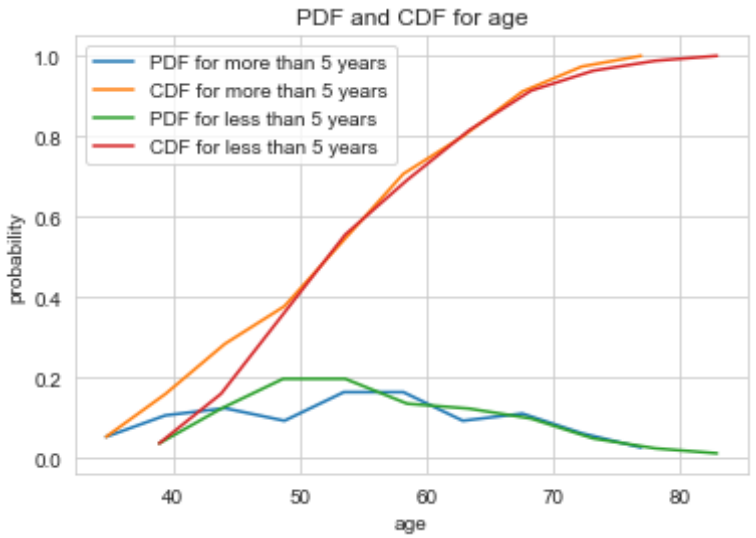
1. The third graph tells us that the survival rate is comparitively high when the number of axillary nodes is less than 3
2. The first 2 graphs have high overlap, due to which no inference can be made

7) PDF and CDF

In [7]:
```
#To calculate PDF and CDF for both the statuses

counts_1, edges_1 = np.histogram(dtst_1['nodes'], bins=10, density=True
)
counts_2, edges_2 = np.histogram(dtst_2['nodes'], bins=10, density=True
)

pdf_1,pdf_2 = counts_1/sum(counts_1), counts_2/sum(counts_2)
cdf_1, cdf_2=np.cumsum(pdf_1), np.cumsum(pdf_2)

plot.title('PDF and CDF for nodes')
plot.xlabel('nodes')
plot.ylabel('probability')
plot.plot(edges_1[1:],pdf_1,label='PDF for more than 5 years')
plot.plot(edges_1[1:],cdf_1,label='CDF for more than 5 years')
plot.plot(edges_2[1:],pdf_2,label='PDF for less than 5 years')
plot.plot(edges_2[1:],cdf_2,label='CDF for less than 5 years')
plot.legend()
plot.show()

counts_1, edges_1 = np.histogram(dtst_1['age'], bins=10, density=True)
```

```
counts_2, edges_2 = np.histogram(dtst_2['age'], bins=10, density=True)

pdf_1,pdf_2 = counts_1/sum(counts_1), counts_2/sum(counts_2)
cdf_1, cdf_2=np.cumsum(pdf_1), np.cumsum(pdf_2)

plot.title('PDF and CDF for age')
plot.xlabel('age')
plot.ylabel('probability')
plot.plot(edges_1[1:],pdf_1,label='PDF for more than 5 years')
plot.plot(edges_1[1:],cdf_1,label='CDF for more than 5 years')
plot.plot(edges_2[1:],pdf_2,label='PDF for less than 5 years')
plot.plot(edges_2[1:],cdf_2,label='CDF for less than 5 years')
plot.legend()
plot.show()
```





Observations from the above PDF and CDF graphs:

1. Contradicting observations from the above graphs.
2. 85% of the patients with less than 5 nodes survived more than 5 years.
3. 55% of the patients with less than 5 nodes, died within 5 years.

8) Mean, Variance, Standard deviation

In [8]:
```
print("Mean of nodes:")
print("without outlier:",np.mean(dtst_1["nodes"]))
print("with outlier:",np.mean(np.append(dtst_1["nodes"],100)))
print("without outlier:",np.mean(dtst_2["nodes"]))
print("with outlier:",np.mean(np.append(dtst_2["nodes"],100)))

print("\nStd dev of nodes:")
print(np.std(dtst_1["nodes"]))
print(np.std(dtst_2["nodes"]))
```

```
print("\nMean of age:")
print("without outlier:",np.mean(dtst_1["age"]))
print("with outlier:",np.mean(np.append(dtst_1["age"],200)))
print("without outlier:",np.mean(dtst_2["age"]))
print("with outlier:",np.mean(np.append(dtst_2["age"],200)))

print("\nStd dev of age:")
print(np.std(dtst_1["age"]))
print(np.std(dtst_2["age"]))
```

```
Mean of nodes:
without outlier: 2.7911111111111113
with outlier: 3.2212389380530975
without outlier: 7.45679012345679
with outlier: 8.585365853658537

Std dev of nodes:
5.857258449412131
9.128776076761632

Mean of age:
without outlier: 52.01777777777778
with outlier: 52.67256637168141
without outlier: 53.67901234567901
with outlier: 55.46341463414634

Std dev of age:
10.98765547510051
10.10418219303131
```

Observations from the above stats:

1. Although the means of nodes between the 2 classes are wide apart, their standard deviations tell us that there is large overlap between the 2 classes
2. The mean of the ages between the 2 classes are almost same and the standard deviation is same too. So, no inference can be made.

9) Median, Quantiles, Percentiles and IQR

In [9]:
```
print("\nMedian of nodes:")
print("without outlier:",np.median(dtst_1["nodes"]))
print("with outlier:",np.median(np.append((dtst_1["nodes"]),100)))
print("without outlier:",np.median(dtst_2["nodes"]))
print("with outlier:",np.median(np.append((dtst_2["nodes"]),100)))

print("\nQuantiles on nodes:")
print(np.percentile(dtst_1["nodes"],np.arange(0, 100, 20)))
print(np.percentile(dtst_2["nodes"],np.arange(0, 100, 20)))

print("\nPercentiles on nodes:")
print("83rd percentile for class 1:",np.percentile(dtst_1["nodes"],83))
 #used brute force approach to arrive at the value 83
print("50th percentile for class 2:",np.percentile(dtst_2["nodes"],50))
 #used brute force approach to arrive at the value 83

from statsmodels import robust
print ("\nMedian Absolute Deviation for nodes")
print(robust.mad(dtst_1["nodes"]))
print(robust.mad(dtst_2["nodes"]))

print("\nMedian of age:")
print(np.median(dtst_1["age"]))
print(np.median(dtst_2["age"]))

print("\nQuantiles on age:")
print(np.percentile(dtst_1["age"],np.arange(0, 100, 20)))
print(np.percentile(dtst_2["age"],np.arange(0, 100, 20)))

from statsmodels import robust
print ("\nMedian Absolute Deviation for age")
print(robust.mad(dtst_1["age"]))
print(robust.mad(dtst_2["age"]))
```

```
Median of nodes:
without outlier: 0.0
with outlier: 0.0
```

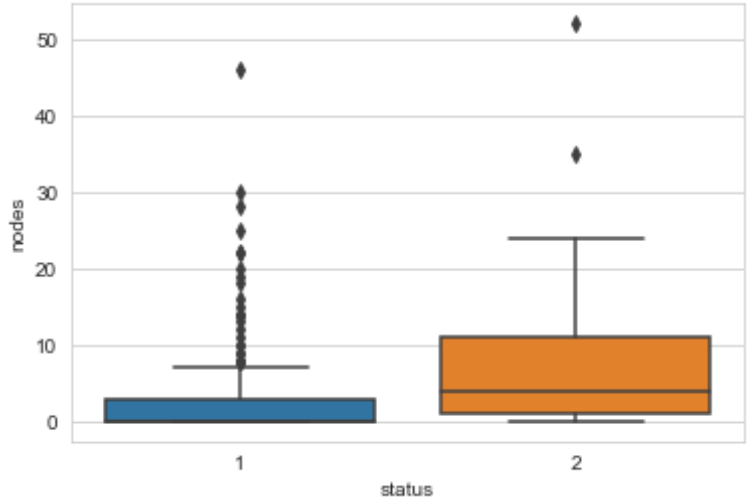
```
without outlier: 4.0
with outlier: 4.0

Quantiles on nodes:
[0. 0. 0. 1. 4.]
[ 0.  0.  3.  6. 13.]

Percentiles on nodes:
83rd percentile for class 1: 4.0
50th percentile for class 2: 4.0

Median Absolute Deviation for nodes
0.0
5.930408874022408

Median of age:
52.0
53.0

Quantiles on age:
[30.  41.  49.  55.  62.2]
[34. 45. 50. 54. 62.]

Median Absolute Deviation for age
13.343419966550417
11.860817748044816
```

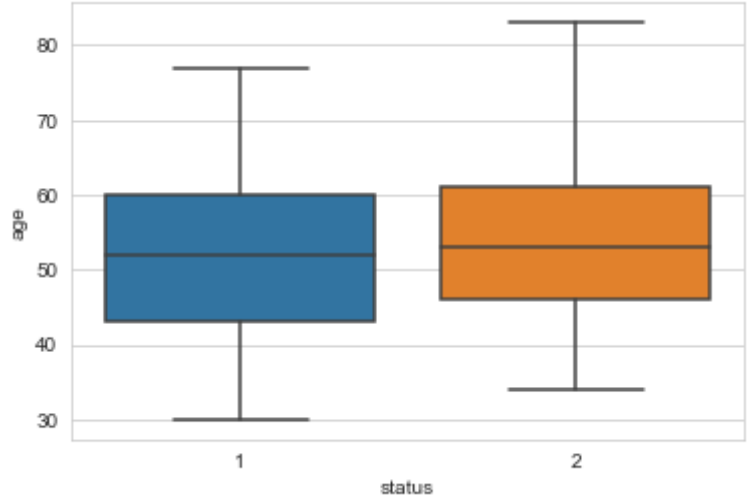
Observations from the above stats:

1. Unlike mean, the median is not changing with outlier values.
2. 83% of the patients with nodes less than 4 have survived more than 5 years after diagnosis
3. 50% of patients with nodes greater than 4 have not survived more than 5 years
4. Due to contradicting stats, no meaningful inference can be made

10) Box plots

In [10]:
```
sns.boxplot(x='status',y='nodes', data=dtst)
plot.show()

sns.boxplot(x='status',y='age', data=dtst)
plot.show()
```



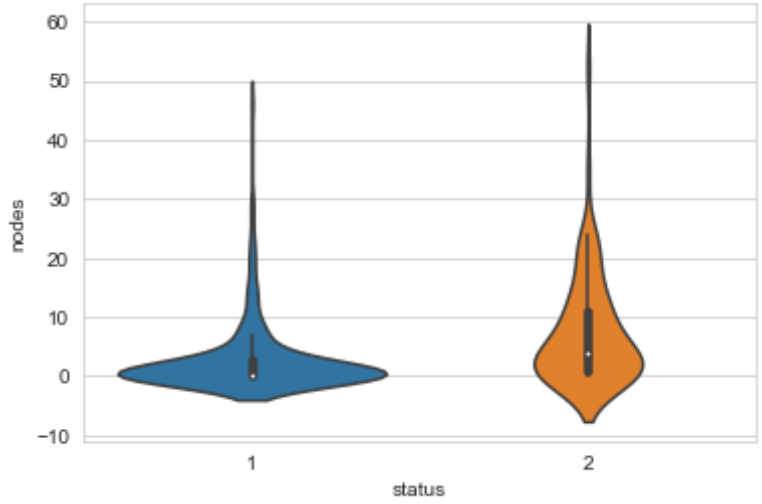


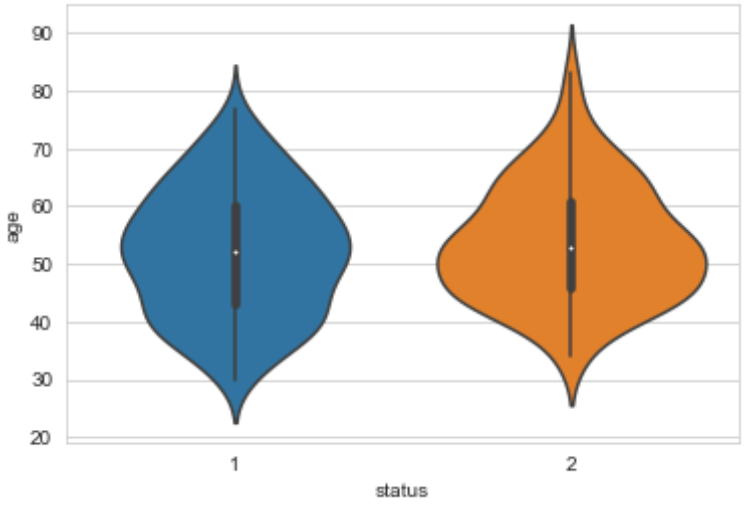
Observations:

1. As the 75 percentile value is less than 5, we can say that the lesser the number of nodes, greater the chances of survival.
2. Major overlap in the age graph. No inference can be made.

11) Violin plots

In [26]:
```
sns.violinplot(x="status", y="nodes", data=dtst, height=10)
plot.show()

sns.violinplot(x="status", y="age", data=dtst, height=10)
plot.show()
```



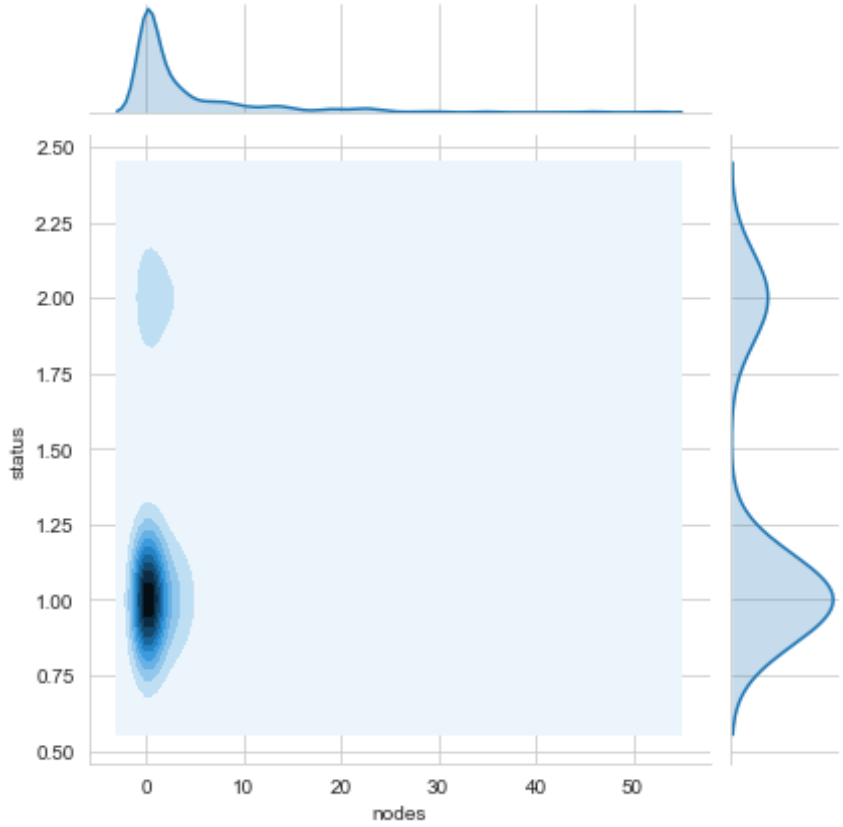


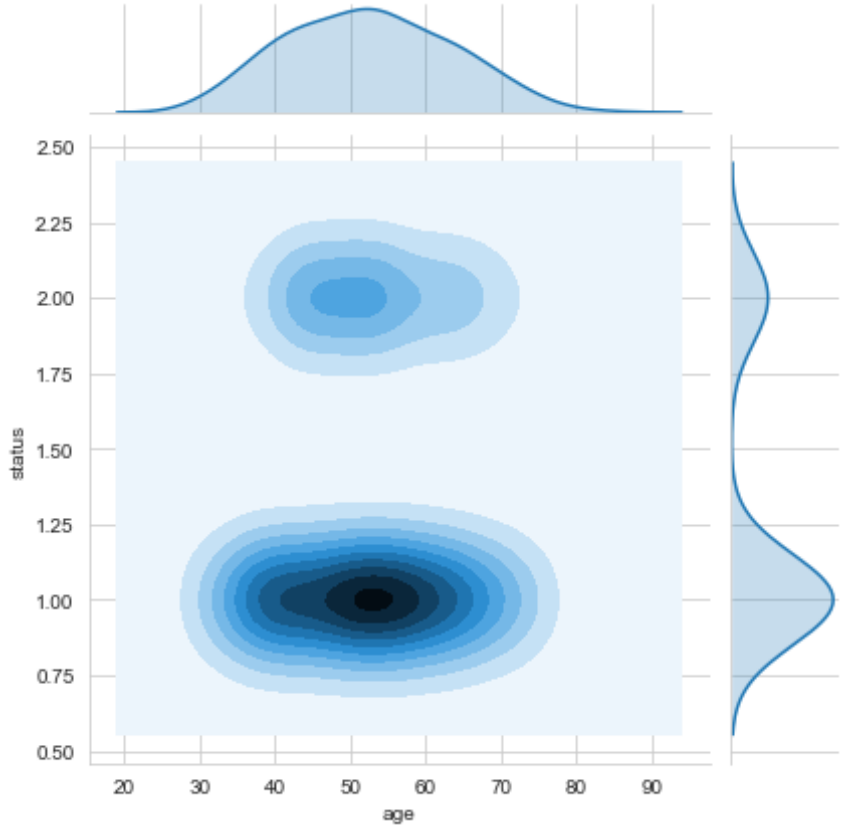
Observations:

1. As the PDF curve is wider in the 0-3 nodes area, we can say that the lesser the number of nodes, greater the chances of survival after 5 years of diagnosis.
2. Both the violin structures in the age graph are similar in size. No inference can be made.

12) Contour Density Plots

In [12]:
```
sns.jointplot(x="nodes", y="status", data=dtst, kind="kde");
plot.show();

sns.jointplot(x="age", y="status", data=dtst, kind="kde");
plot.show();
```





Observations:

1. Most of the data points for status 1 are concentrated in the 0-5 nodes area. So, the survival chances are greater if the number of nodes are less.
2. The chances of survival are higher if the age of the patient is between 45-60.

Conclusion:

1. The dataset is imbalanced as the number of data points are not equal across status types.

2. Although no proper threshold can be set to find survival chances, we can say that lesser the number of nodes, greater the chances of survival.
3. If the patient age falls between 45-60, there are greater chances of survival.