In [2]:
```python
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
import warnings
```

In [3]:
```python
hbrmn = pd.read_csv("haberman.csv")    # loading data
print(hbrmn.shape)   # print the shape to understand the row and columns
```

```
(305, 4)
```

In [5]:
```python
hbrmn.columns=['age', 'year', 'nodes', 'status']
print(hbrmn.columns)                       # print the all features
```

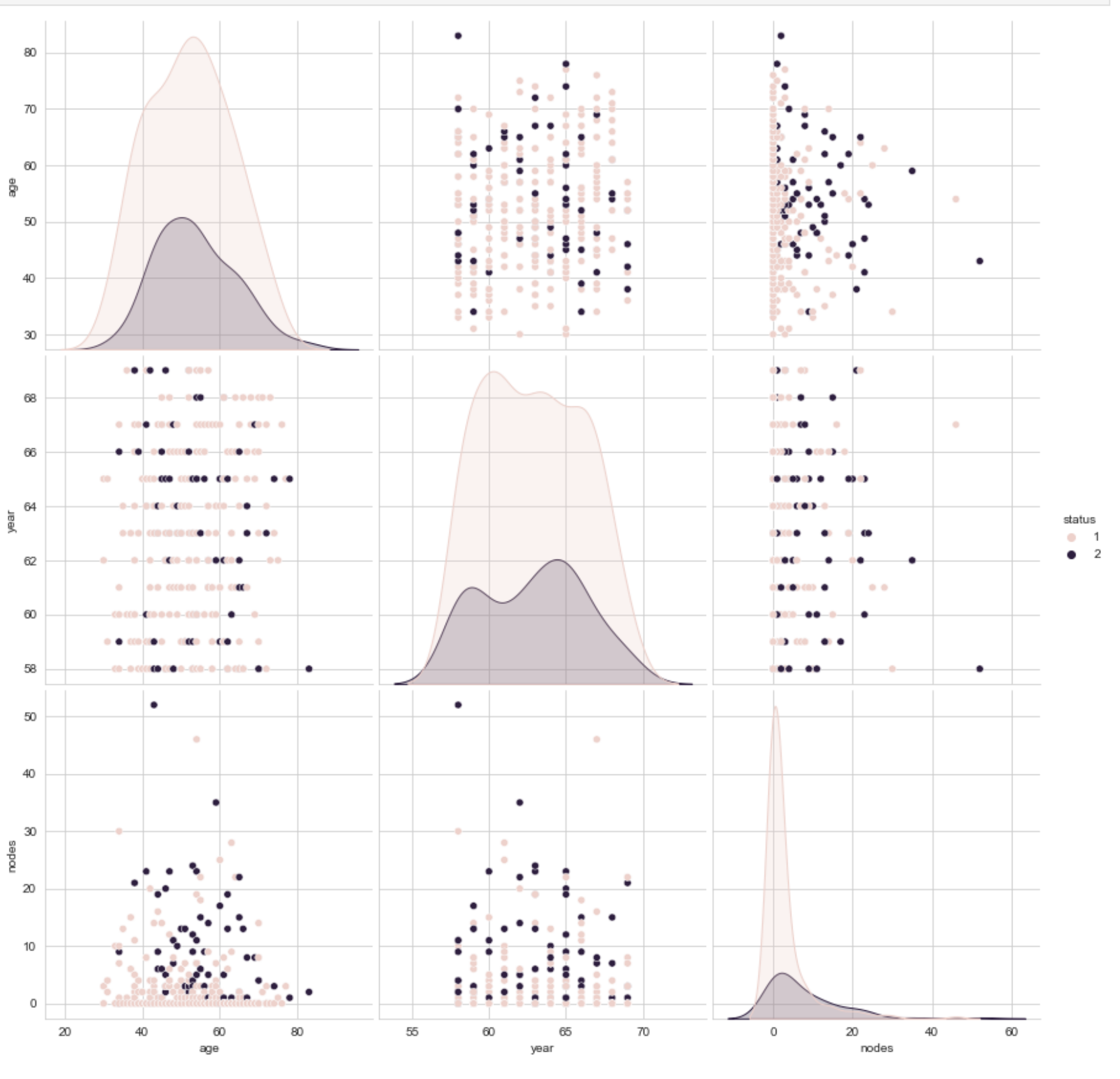
```
Index(['age', 'year', 'nodes', 'status'], dtype='object')
```

In [6]:
```python
hbrmn["status"].value_counts()         # count the no data in each class
```

Out[6]:
```
1    224
2     81
Name: status, dtype: int64
```

ploting the pairplot, it give graphical representation of each feature with respect to other one

In [7]:
```python
sns.set_style("whitegrid");
sns.pairplot(hbrmn, hue="status", height=4);
# plt.title("pairplot of hbrmn")
plt.show()
warnings.filterwarnings("ignore")
```



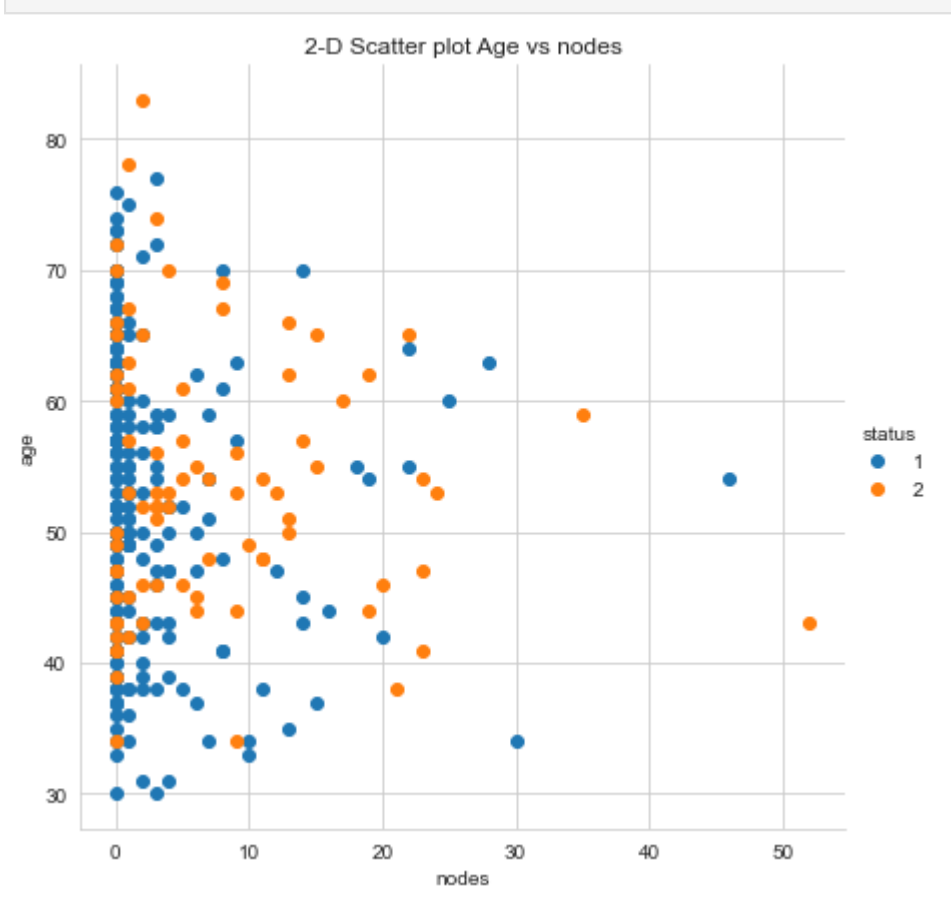
obervations - here we get total six graphs but we see go through above diagonal one

1 - age/year, age/nodes, year/nodes and as we see [age/node] is more clear than the other one

2 - in graph [age/year , year/nodes] we see more overlapping of the points

here we draw seperate graph for the [age/node]

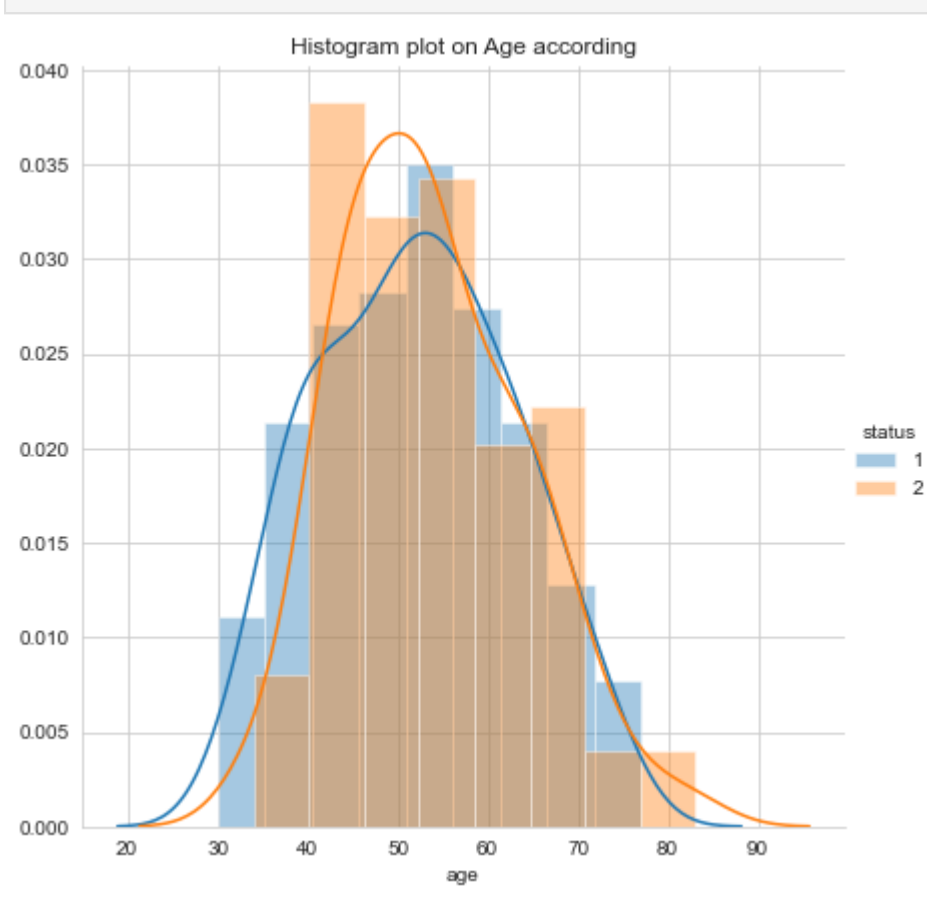
In [8]:
```python
sns.set_style("whitegrid");
sns.FacetGrid(hbrmn, hue="status", height=6) \
   .map(plt.scatter,"nodes", "age") \
   .add_legend();
plt.title("2-D Scatter plot Age vs nodes")
plt.show();
```



observation - here we clearly see understand the approx(range) of age and nodes according to the status

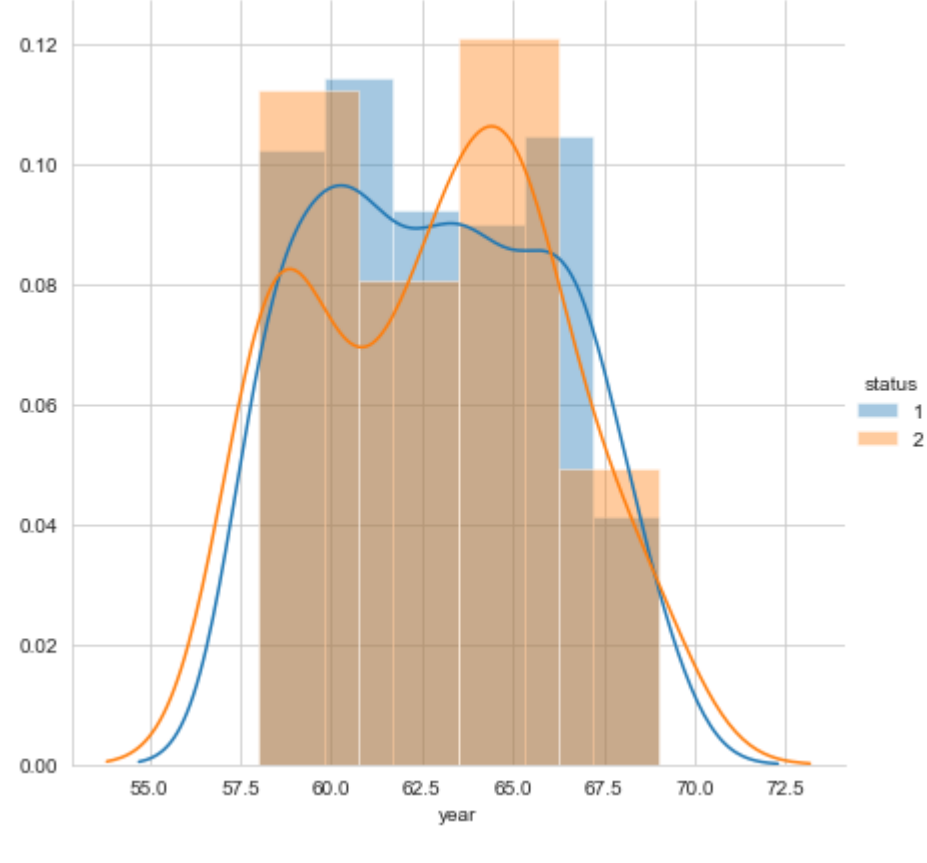
Histogram

In [9]:
```python
sns.FacetGrid(hbrmn, hue="status", height=6) \
   .map(sns.distplot, "age") \
   .add_legend();
plt.title("Histogram plot on Age according")
plt.show();
warnings.filterwarnings("ignore")
```



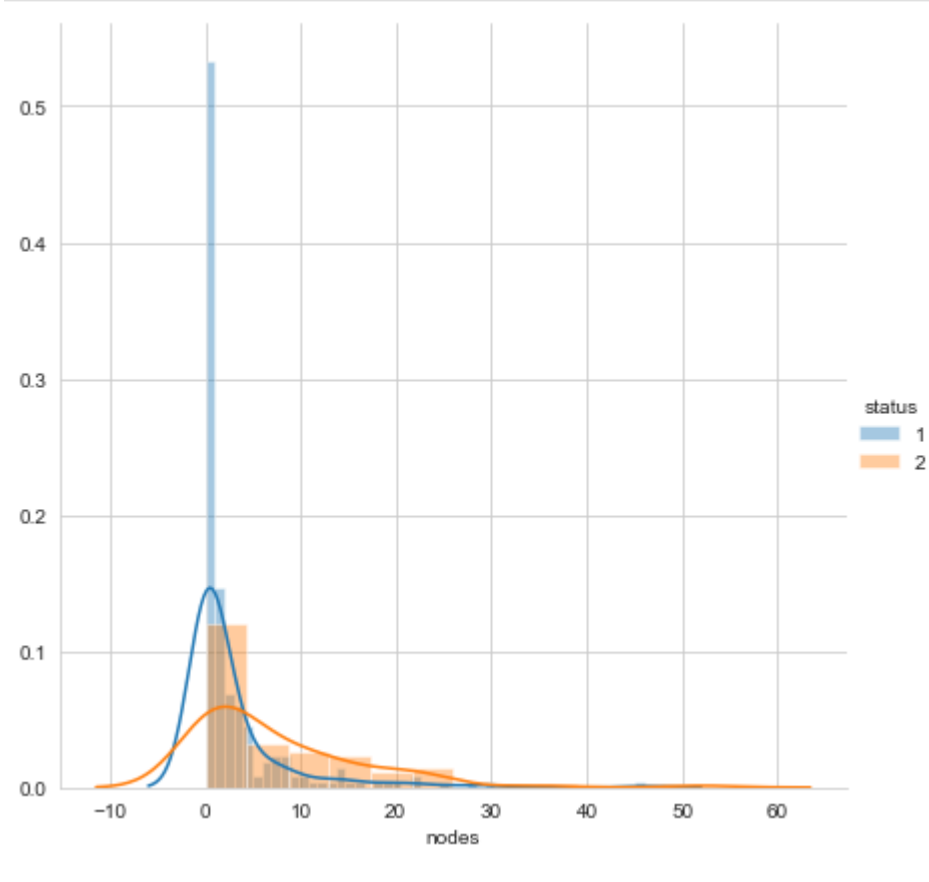
observation - very hard to understand , because mostly data overlap.... even the pdf conside

In [10]:
```python
sns.FacetGrid(hbrmn, hue="status", height=6) \
   .map(sns.distplot, "year") \
   .add_legend();
plt.show();
warnings.filterwarnings("ignore")
```



observation - very hard to understand , because mostly data overlap even pdf overlap

In [11]:
```python
sns.FacetGrid(hbrmn, hue="status", height=6) \
   .map(sns.distplot, "nodes") \
   .add_legend();
plt.show();
warnings.filterwarnings("ignore")
```



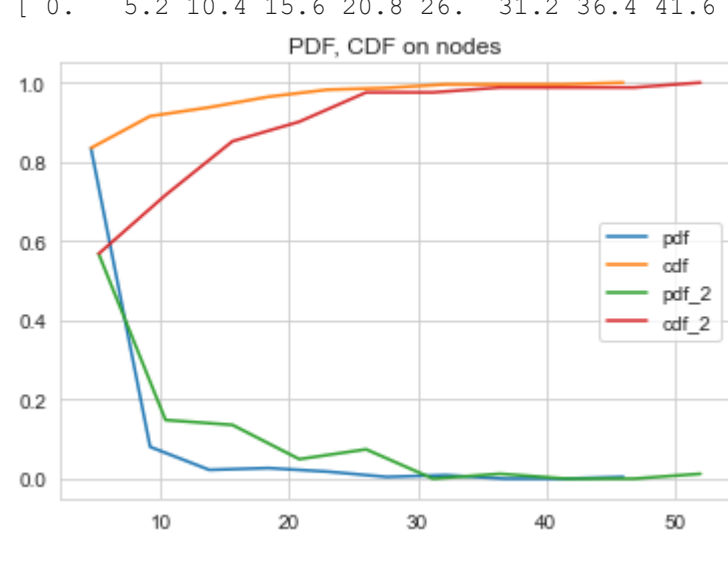
observation - more no. of people who survive have very less no. of the nodes

final observation -- nodes >>> age > year

PDF, CDF

In [12]:
```python
hbrmn_1 = hbrmn.loc[hbrmn["status"] == 1]
hbrmn_2 = hbrmn.loc[hbrmn["status"] == 2]

counts, bin_edges = np.histogram(hbrmn_1['nodes'], bins=10,
                                 density = True)
pdf = counts/(sum(counts))
print(pdf);
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf,label = 'pdf')
plt.plot(bin_edges[1:], cdf,label = "cdf")
plt.legend()

counts, bin_edges = np.histogram(hbrmn_2['nodes'], bins=10,
                                 density = True)
pdf = counts/(sum(counts))
print(pdf);
print(bin_edges)
cdf = np.cumsum(pdf)
plt.title("PDF, CDF on nodes")
plt.plot(bin_edges[1:],pdf,label = 'pdf_2')
plt.plot(bin_edges[1:], cdf,label = "cdf_2")
plt.legend()
warnings.filterwarnings("ignore")
```

```
[0.83482143 0.08035714 0.02232143 0.02678571 0.01785714 0.00446429
 0.00892857 0.         0.         0.00446429]
[ 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. ]
```



observations : as we see both pdf and cdf line in the graph, A.T.CDF - 81-82% survival chance have nodes less than equal to 4 or 5

observations : as we see both pdf and cdf line in the graph, A.T.CDF_2 - 58-59% non-survival chance have nodes less than equal to 4 or 5

In [13]:
```python
#Mean, Variance, Std-deviation,

print("means:")
print(np.mean(hbrmn_1['nodes']))
print(np.mean(hbrmn_2['nodes']))


print("\nSTD-dev:");
print(np.std(hbrmn_1['nodes']))
print(np.std(hbrmn_2['nodes']))
```

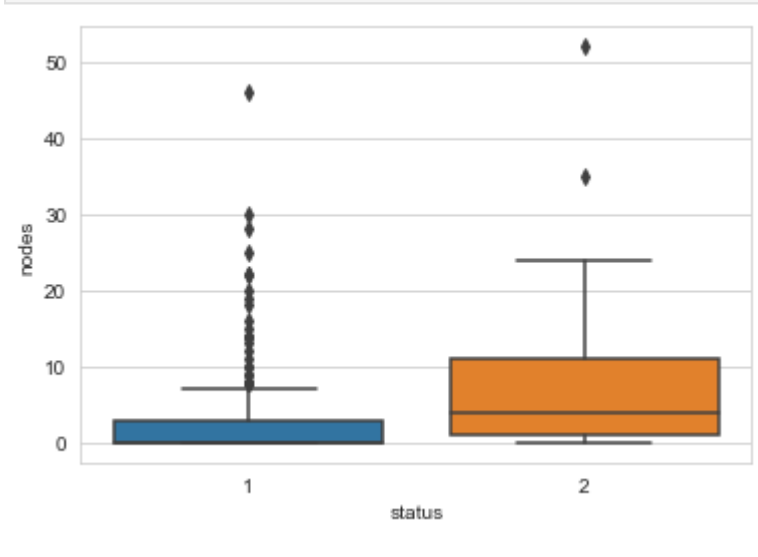
```
means:
2.799107142857143
7.45679012345679

STD-dev:
5.869092706952767
9.128776076761632
```

In [14]:
```python
#Median, Quantiles, Percentiles, IQR.

print("\nMedians:")
print(np.median(hbrmn_1['nodes']))
print(np.median(hbrmn_2['nodes']))

print("\nQuantiles:")
print(np.percentile(hbrmn_1['nodes'],np.arange(0, 100, 25)))
print(np.percentile(hbrmn_2['nodes'],np.arange(0, 100, 25)))


print("\n90th Percentiles:")
print(np.percentile(hbrmn_1['nodes'],90))
print(np.percentile(hbrmn_2['nodes'],90))

from statsmodels import robust
print ("\nMedian Absolute Deviation")
print(robust.mad(hbrmn_1['nodes']))
print(robust.mad(hbrmn_2['nodes']))
```

```
Medians:
0.0
4.0

Quantiles:
[0. 0. 0. 3.]
[ 0.  1.  4. 11.]

90th Percentiles:
8.0
20.0

Median Absolute Deviation
0.0
5.930408874022408
```

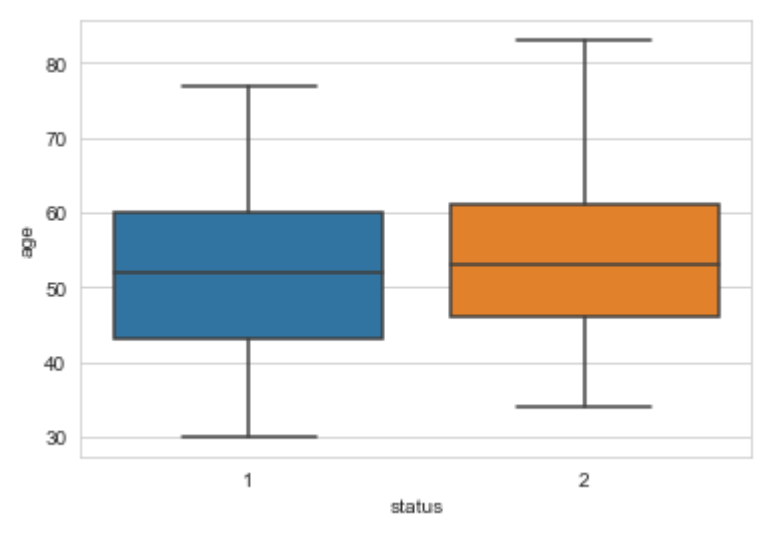
Box plot

In [15]:
```python
sns.boxplot(x='status',y='nodes', data=hbrmn)
plt.show()
```



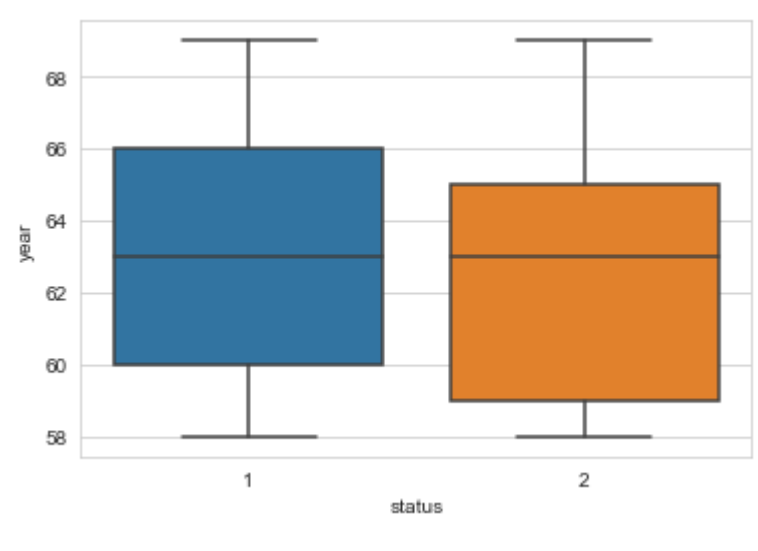
observations: here we see the interquartile range of the nodes, in servival - not more than 4

In [16]:
```python
sns.boxplot(x='status',y='age', data=hbrmn)
plt.show()
```



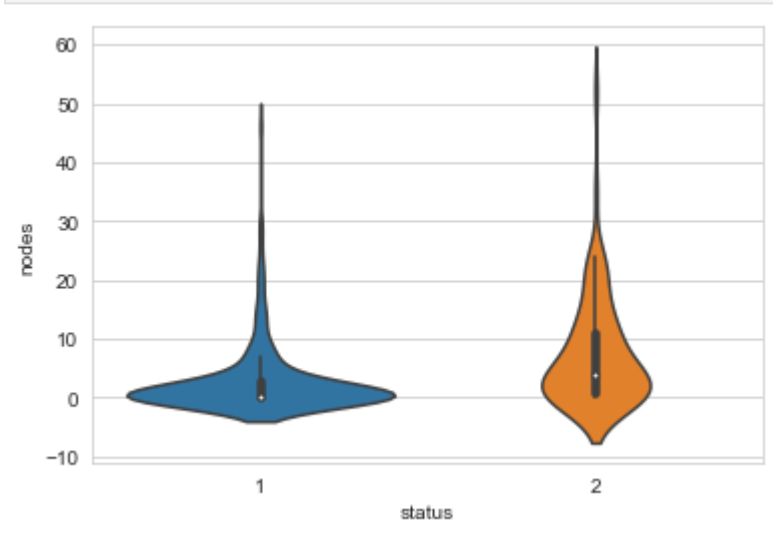
observation: interquartile range of servival in age factor have 44-60

In [17]:
```python
sns.boxplot(x='status',y='year', data=hbrmn)
plt.show()
```



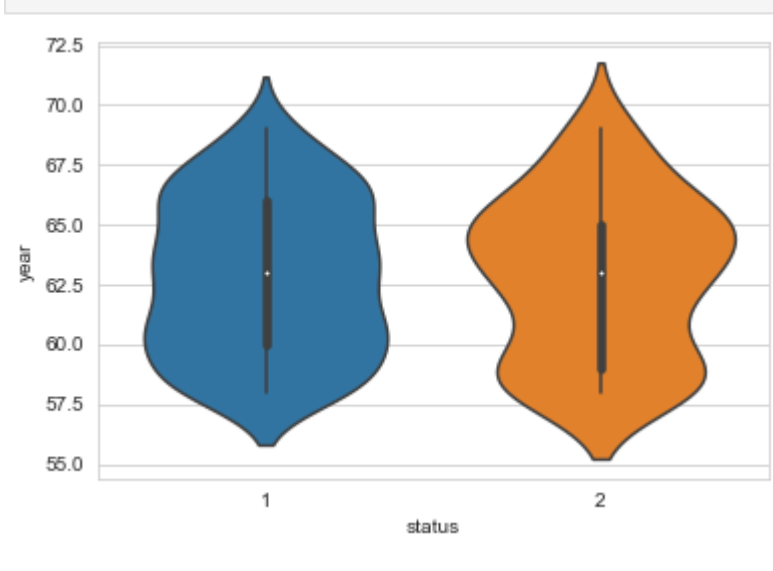
observation: here we see that interqartile range of servival is 60 - 66

In [18]:
```python
sns.violinplot(x="status", y="nodes", data=hbrmn, size=8)
plt.show()
```



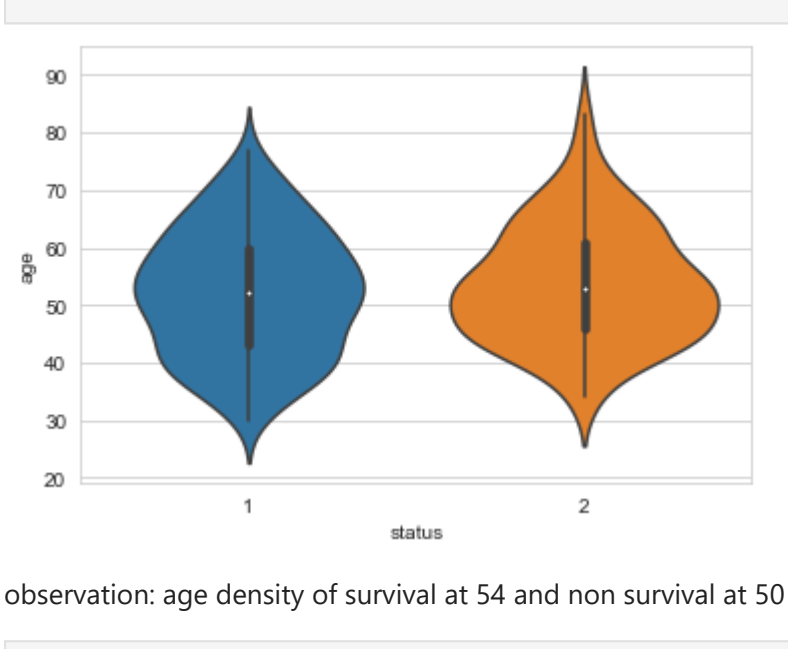
observation: node density is high on 0 at servival

In [19]:
```python
sns.violinplot(x="status", y="year", data=hbrmn, size=8)
plt.show()
```



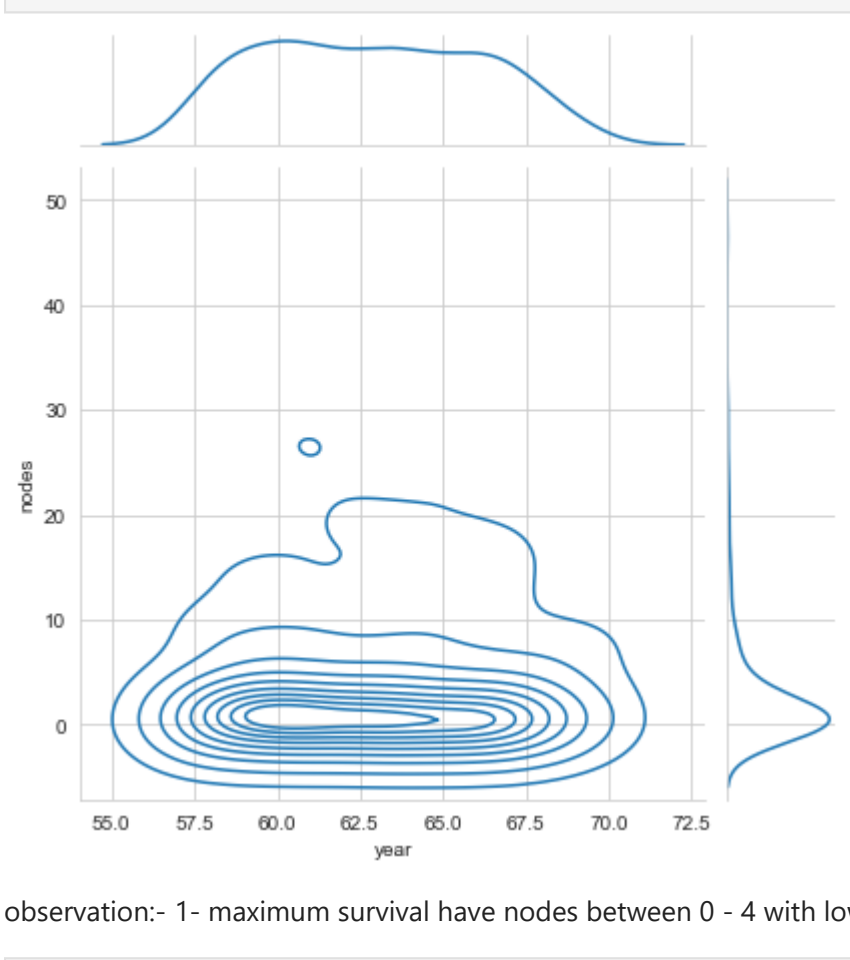
observation: year density is high on 60 at survival and on non survival at 64

In [20]:
```python
sns.violinplot(x="status", y="age", data=hbrmn, size=8)
plt.show()
```



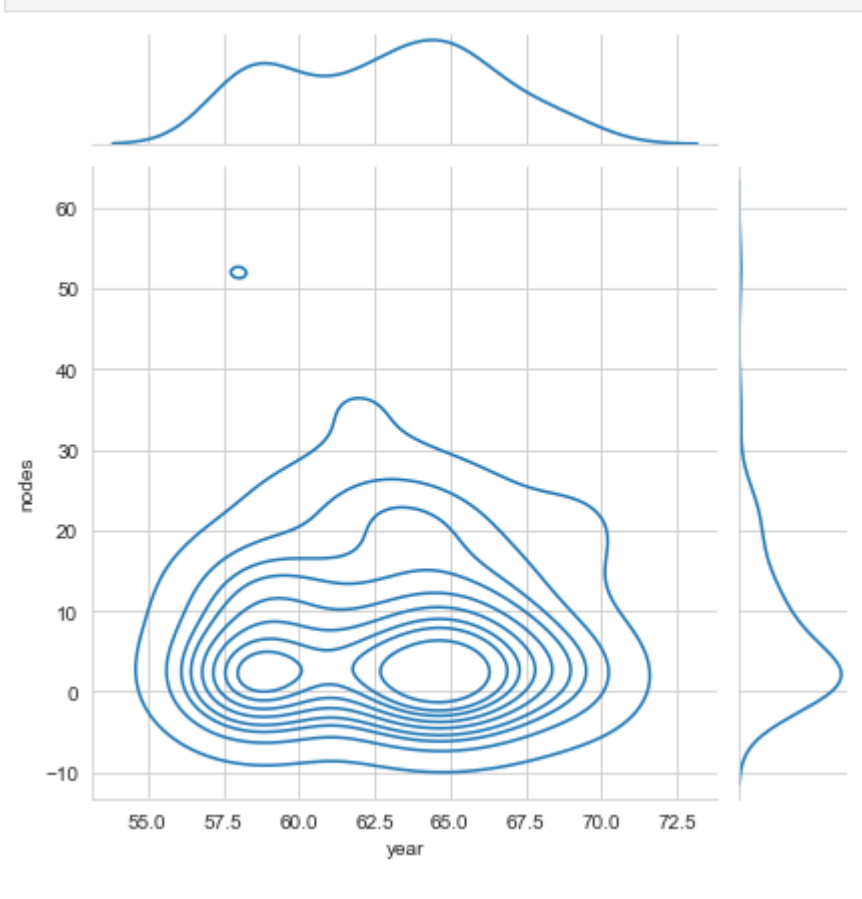
observation: age density of survival at 54 and non survival at 50

In [21]:
```python
sns.jointplot(x="year", y="nodes", data=hbrmn_1, kind="kde");
plt.show();
```



observation:- 1- maximum survival have nodes between 0 - 4 with low varience

In [22]:
```python
sns.jointplot(x="year", y="nodes", data=hbrmn_2, kind="kde");
plt.show();
```



observation: here in the non-survival the nodes have higher value what high varience

OBSERVATION

1 - Age/Node show the clear observation than rest of all (Here we can clearly visualize the data)

2- age/node graph (in age 40-60 death are more on having node lessthan 5)

3- maximum survival have nodes between 0 - 4

4- 81-82% survival chance have nodes less than equal to 4 or 5

5 - 58-59% non-survival chance have nodes less than equal to 4 or 5

6- age density of survival at 54 and non survival at 50 [age ------ dignose age]

7- interquartile range of servival in age factor have 44-60(more survival if detected node b/w 44-60)

8- year density is high on 60 at survival and on non survival at 64 (year[final --- recovered or death])

9- range of servival is 60 - 66 (more survival if node oprate b/w 60-66)