In [1]:

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

In [2]:

```
df=pd.read_csv("haberman.csv")
```

In [3]:

```
df
```

Out[3]:

	30	64	1	1.1
0	30	62	3	1
1	30	65	0	1
2	31	59	2	1
3	31	65	4	1
4	33	58	10	1
...	...	...	...	...
300	75	62	1	1
301	76	67	0	1
302	77	65	3	1
303	78	65	1	2
304	83	58	2	2

305 rows × 4 columns

In [4]:

```
df.shape
```

Out[4]:

```
(305, 4)
```

In [41]:

```
df.isnull().sum()
```

Out[41]:

```
30     0
64     0
1      0
1.1    0
dtype: int64
```

In [40]:

```
df["1.1"].value_counts()
```

Out[40]:

```
Yes    224
No      81
Name: 1.1, dtype: int64
```

Observation:

1. **There are 305 rows and 4 columns including class column.**
2. **There are no missing value.**
3. **This data set is unbalanced as it has 224 patient of one category and 81 patient of other category.**

In [33]:

```
df["1.1"]=df["1.1"].map({1:"Yes",2:"No"})
```

In [34]:

```
df.head()
```

Out[34]:

	30	64	1	1.1
0	30	62	3	Yes
1	30	65	0	Yes
2	31	59	2	Yes
3	31	65	4	Yes
4	33	58	10	Yes

In [35]:

```
df.describe()
```

Out[35]:

	30	64	1
count	305.000000	305.000000	305.000000
mean	52.531148	62.849180	4.036066
std	10.744024	3.254078	7.199370
min	30.000000	58.000000	0.000000
25%	44.000000	60.000000	0.000000
50%	52.000000	63.000000	1.000000
75%	61.000000	66.000000	4.000000
max	83.000000	69.000000	52.000000

In [36]:

```
survive_yes=df[df["1.1"]=="Yes"]
survive_no=df[df["1.1"]=="No"]
```

In [37]:

```
survive_yes.describe()
```

Out[37]:

	30	64	1
count	224.000000	224.000000	224.000000
mean	52.116071	62.857143	2.799107
std	10.937446	3.229231	5.882237
min	30.000000	58.000000	0.000000
25%	43.000000	60.000000	0.000000
50%	52.000000	63.000000	0.000000

	30	64	1
50%	52.000000	63.000000	0.000000
75%	60.000000	66.000000	3.000000
max	77.000000	69.000000	46.000000

In [38]:

```
survive_no.describe()
```

Out[38]:

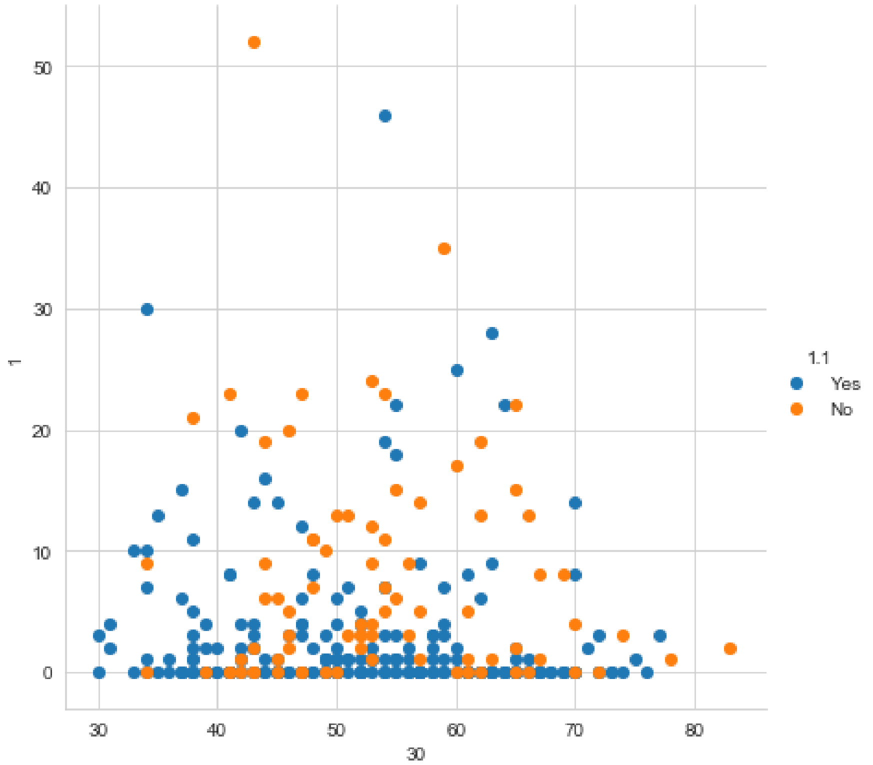
	30	64	1
count	81.000000	81.000000	81.000000
mean	53.679012	62.827160	7.456790
std	10.167137	3.342118	9.185654
min	34.000000	58.000000	0.000000
25%	46.000000	59.000000	1.000000
50%	53.000000	63.000000	4.000000
75%	61.000000	65.000000	11.000000
max	83.000000	69.000000	52.000000

Observation:

1. **The avarage age and year of operation of person in both the classes are approx same.**
2. **But the # of positive auxilary node is differ by approx 5. The survive class has less number of positive auxilary nodes compare to non-survive class.**

In [39]:

```
sns.set_style("whitegrid")
sns.FacetGrid(df,hue="1.1",height=6) \
   .map(plt.scatter,"30","1") \
   .add_legend()
plt.show()
```



Obsevation:

1. **From above 2D scatter plot we see that we cannot easily distinguish the two categories using the attribute**

"Age"(30) and "Number of positive auxilary nodes detected"(1).

2. The person whose # positive auxilary node is 0 or 1 are more likely to survive irrespective of their ages.
3. The person whose # positive auxilary node is 10 or more and age >=50 are less chances of survive.
4. There are very few people whose # positive auxilary node>=40,seems like they are outlier.

In [8]:

```
import plotly.express as px
fig=px.scatter_3d(df,x="30",y="1",z="64",color="1.1")
fig.show()
```

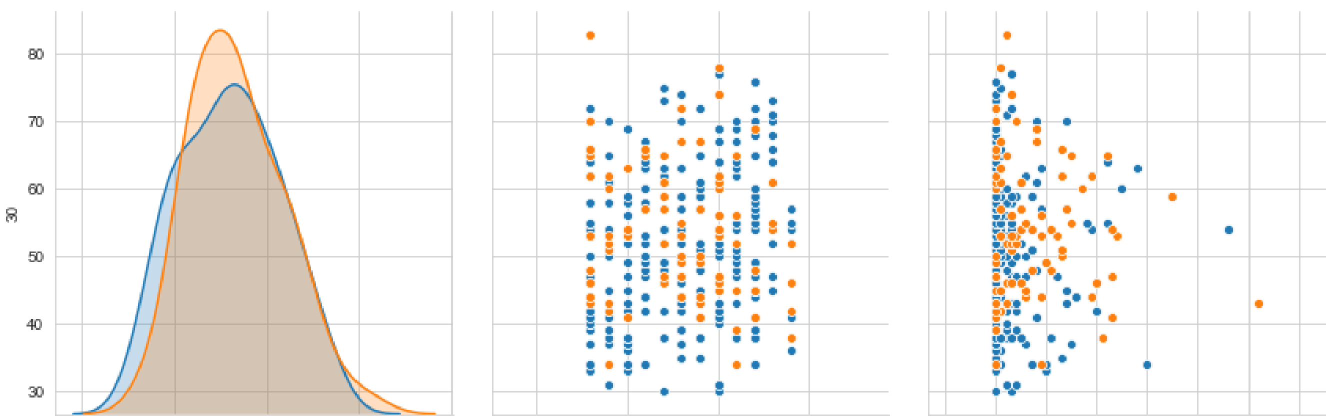
Observation: This 3D scatter plot is also not help us as we still cannot distinguish between two categoris. So let's try pair plot to see any combination will help us or not!!!!!

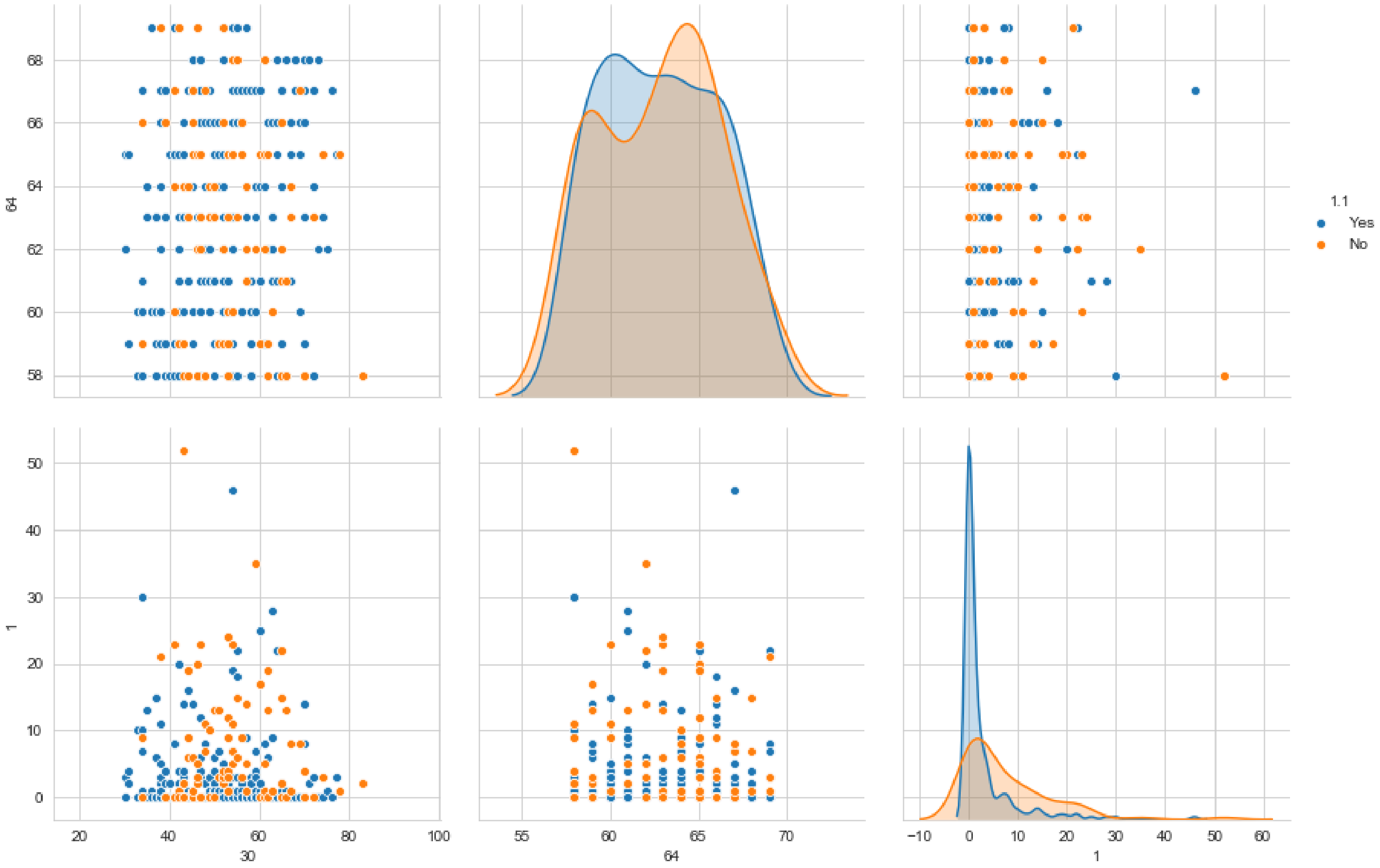
In [44]:

```
sns.set_style("whitegrid")
sns.pairplot(df,hue="1.1",height=4)
```

Out[44]:

```
<seaborn.axisgrid.PairGrid at 0x216c7fafa08>
```





Observation:

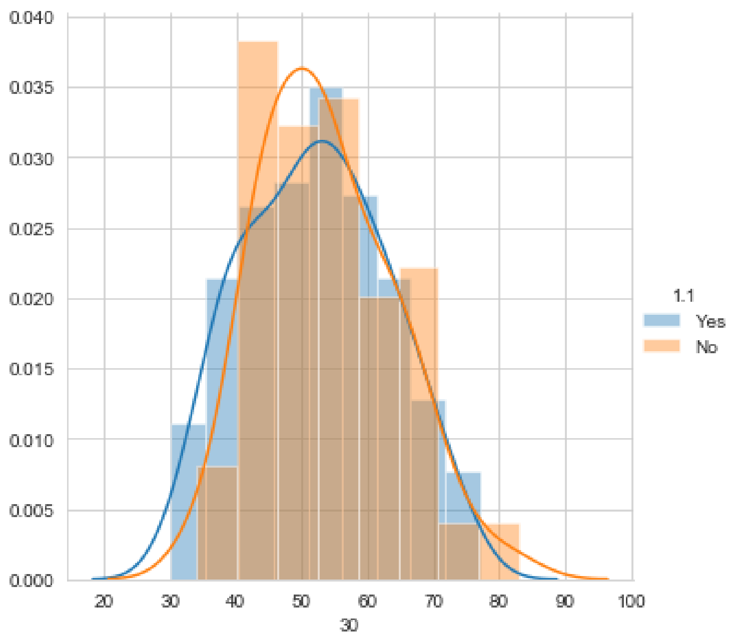
1. **From the above 3c2=3 plot we clearly see that two attribute/feature together cannot helpfull to achieve our goal.Although people having age<=40 are more likely to survived irrespective of year of operation.**

In [45]:

```
df_survive_more=df[df["1.1"]=="Yes"]
df_survive_less=df[df["1.1"]=="No"]
```

In [46]:

```
sns.set_style("whitegrid")
sns.FacetGrid(df,hue="1.1",height=5) \
   .map(sns.distplot,"30") \
   .add_legend()
plt.show()
```

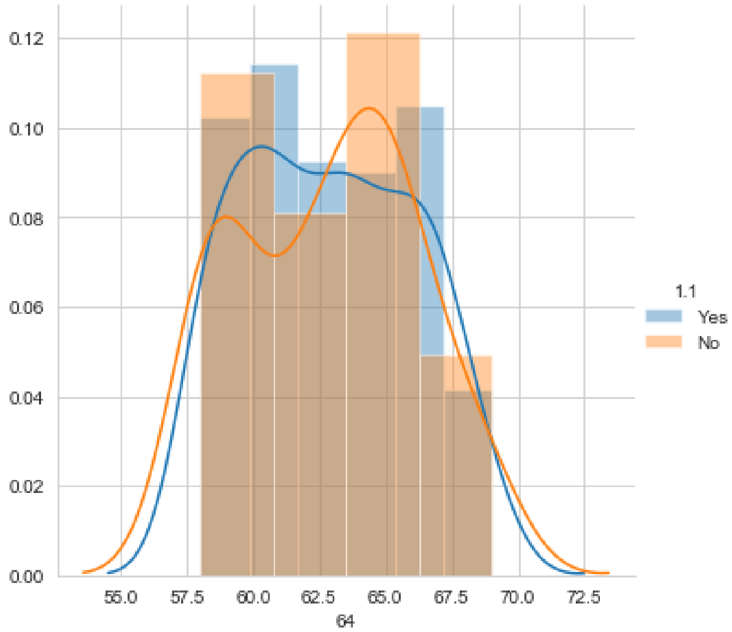


Observation:

1. **Two density plot are almost overlap,this is not good for our objective.**
2. **People whose age are in between 40-60 are less chances to survive and 60-76 the chances are equally likely.**
3. **People whose age are in between 20-40 are more chances to survive.**

In [47]:

```
sns.set_style("whitegrid")
sns.FacetGrid(df,hue="1.1",height=5) \
   .map(sns.distplot,"64") \
   .add_legend()
plt.show()
```

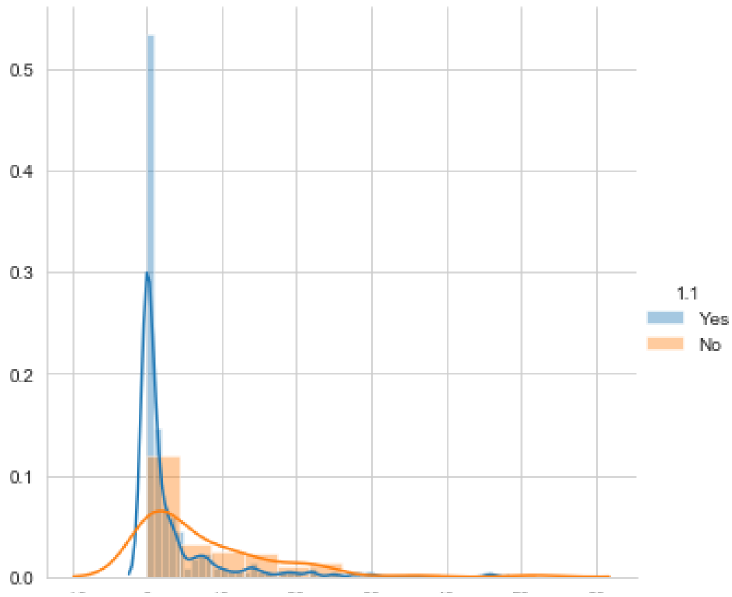


Observation:

1. **In the year 1965 , more unsuccessful opeartion happened and in the year 1961,more successful operation was happened.**

In [48]:

```
sns.set_style("whitegrid")
sns.FacetGrid(df,hue="1.1",height=5) \
   .map(sns.distplot,"1") \
   .add_legend()
plt.show()
```



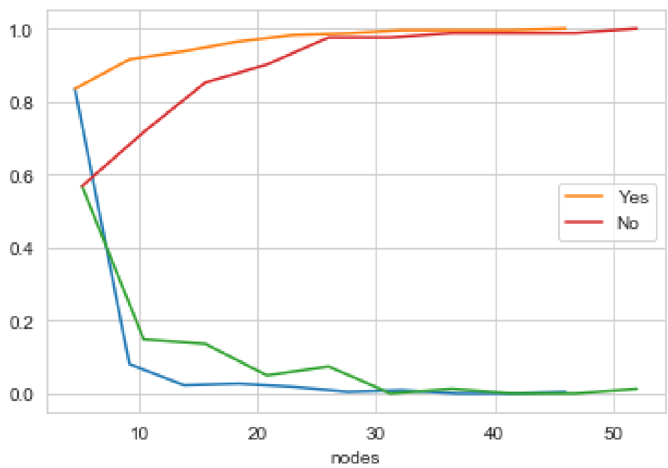
Observation:

1. **From the diagram we see that there are 30% people whose number of positive auxilary node is 0 and they surived.**
2. **People whose number of positive auxilary nodes are in between 4-28 are less chances of survived.**
3. **People whose number of positive auxilary nodes are 30 or more,they instantly died.**

In [57]:

```
counts,bin_edges=np.histogram(survive_yes["1"],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,label="Yes")
plt.xlabel("nodes")

counts,bin_edges=np.histogram(survive_no["1"],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,label="No")
plt.legend()


plt.show()
```

```
[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. ]
```



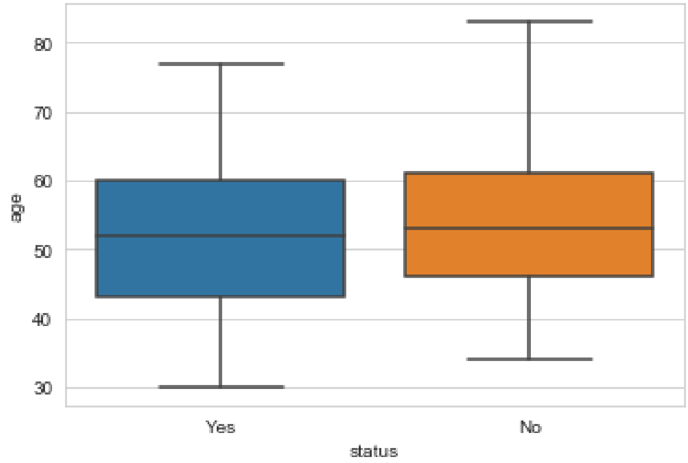
Observation:

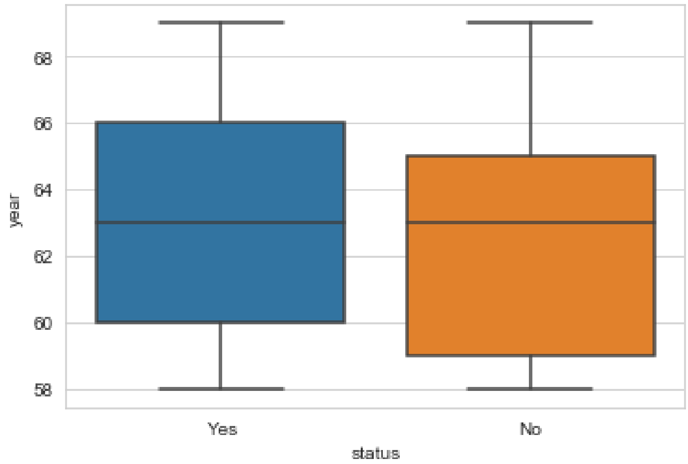
1. **83% people who survived having # of positive auxilary node <=4.**
2. **on the other hand there are people having 0 or 1 positive auxilary node but not survived.So,on the basis of this feature we cannot classify.**

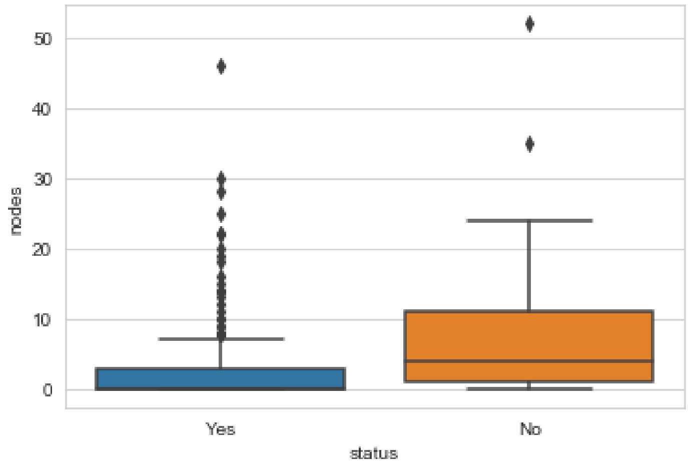
In [60]:

```
sns.boxplot(x="1.1",y="30",data=df)
plt.xlabel("status")
plt.ylabel("age")
plt.show()
```

```
sns.boxplot(x="1.1",y="64",data=df)
plt.xlabel("status")
plt.ylabel("year")
plt.show()
sns.boxplot(x="1.1",y="1",data=df)
plt.xlabel("status")
plt.ylabel("nodes")
plt.show()
```



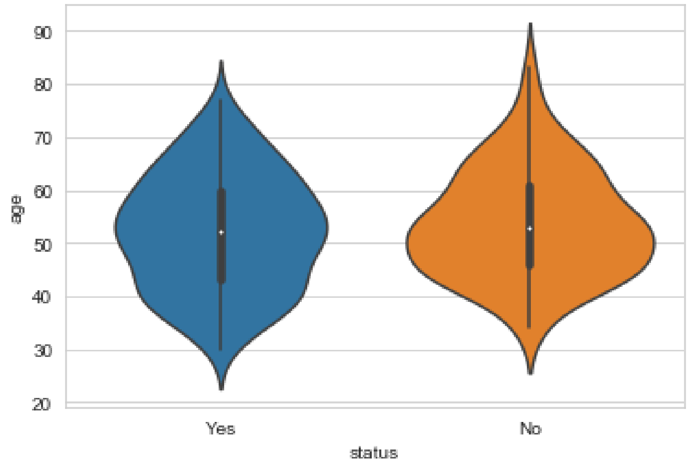


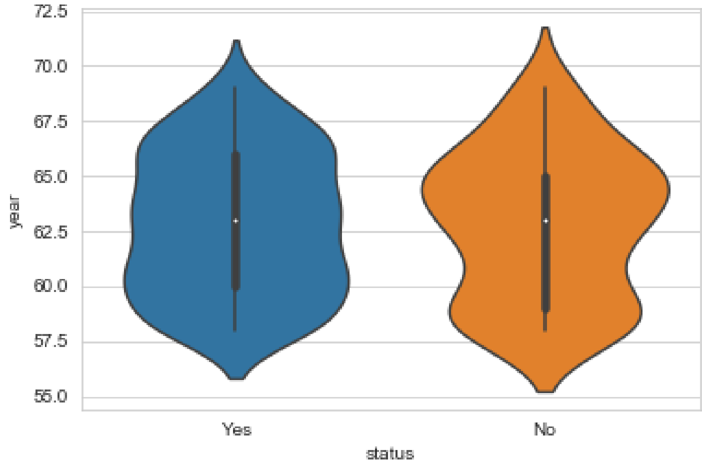


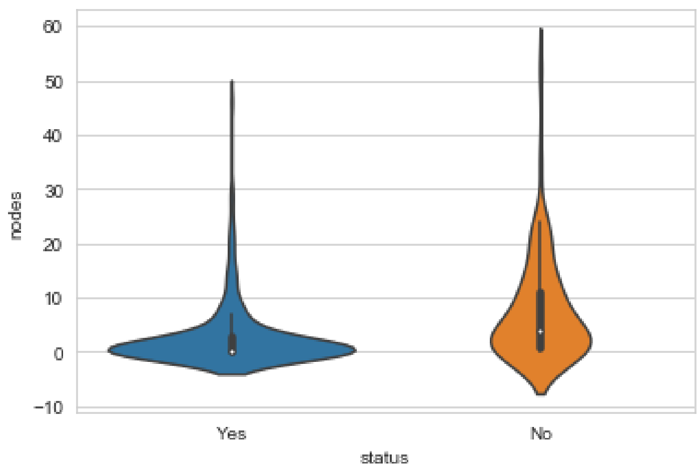
In [63]:

```
sns.violinplot(x="1.1",y="30",data=df)
plt.xlabel("status")
plt.ylabel("age")
plt.show()

sns.violinplot(x="1.1",y="64",data=df)
plt.xlabel("status")
plt.ylabel("year")
plt.show()

sns.violinplot(x="1.1",y="1",data=df)
plt.xlabel("status")
```

```
plt.ylabel("nodes")
plt.show()
```





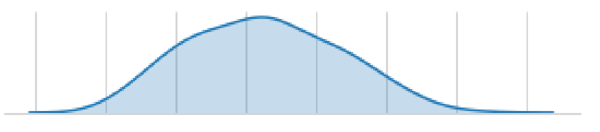


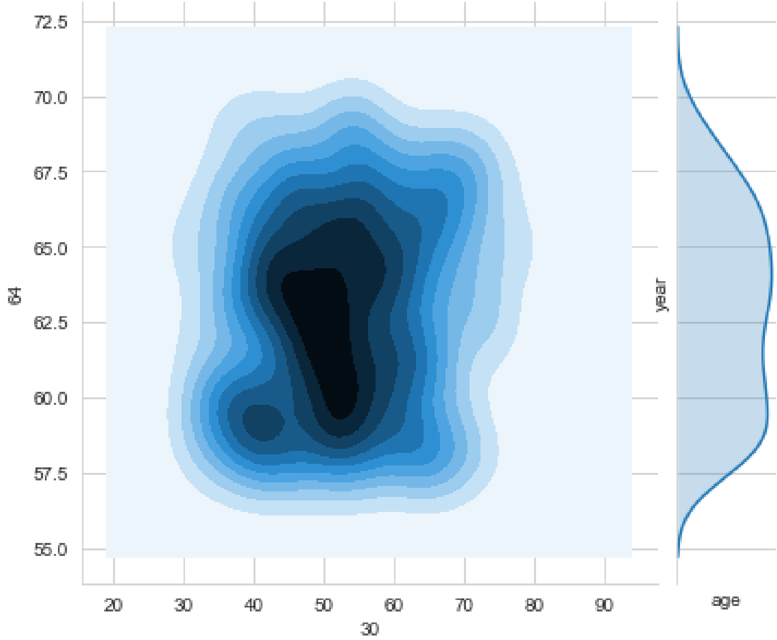
Observation:

1. **From the first two box plot we see that there are almost overlap between two class.**
2. **Box plot corresponding to node is much less overlap compare to other. People having nodes between 0-2 are survived and having nodes between 1-11 are not survived.**
3. **Same conclusion observed from violin plot.**

In [66]:

```
sns.jointplot(x="30",y="64",data=df,kind="kde")
plt.xlabel("age")
plt.ylabel("year")
plt.show()
```





Observation:

1. **Between 1958-1963,people having age 45-55 are operated much.**

Conclusion:

1. **Survival is inversely proportional to the number of positive auxilary nodes.But there are some people with exception.**
2. **The features are given in this dataset are not sufficient for classification.**