In [17]:

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

#df = pd.read_excel(r'C:\Users\user\Downloads\PythonWorkspace\iPythonNotebook\demo.xlsx', sheet_name='Sheet1')
#df = read_pdf(r'C:\Users\user\Downloads\git\DataScience\PythonWorkspace\Assignment\EdaAssignment\Dataset.pdf')
df = pd.read_csv(r'C:\Users\user\Downloads\git\DataScience\PythonWorkspace\Assignment\EdaAssignment\Datasetcsv.csv')
df
```

Out[17]:

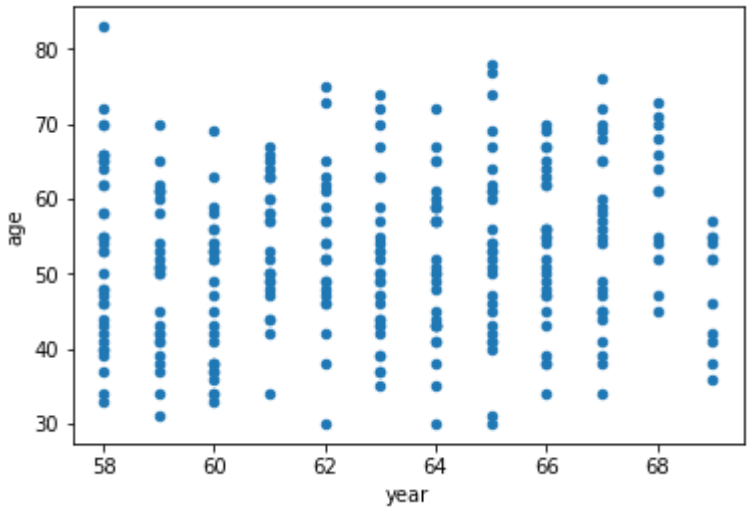
	age	year	nodes	status
0	30	64	1	1
1	30	62	3	1
2	30	65	0	1
3	31	59	2	1
4	31	65	4	1
5	33	58	10	1
6	33	60	0	1
7	34	59	0	2
8	34	66	9	2
9	34	58	30	1
10	34	60	1	1
11	34	61	10	1
12	34	67	7	1
13	34	60	0	1
14	35	64	13	1
15	35	63	0	1
16	36	60	1	1
17	36	69	0	1
18	37	60	0	1
19	37	63	0	1
20	37	58	0	1
21	37	59	6	1
22	37	60	15	1
23	37	63	0	1
24	38	69	21	2
25	38	59	2	1
26	38	60	0	1
27	38	60	0	1
28	38	62	3	1
29	38	64	1	1
...	...	...	...	...
276	67	66	0	1
277	67	61	0	1
278	67	65	0	1
279	68	67	0	1
280	68	68	0	1
281	69	67	8	2

	age	year	nodes	status
282	69	60	0	1
283	69	65	0	1
284	69	66	0	1
285	70	58	0	2
286	70	58	4	2
287	70	66	14	1
288	70	67	0	1
289	70	68	0	1
290	70	59	8	1
291	70	63	0	1
292	71	68	2	1
293	72	63	0	2
294	72	58	0	1
295	72	64	0	1
296	72	67	3	1
297	73	62	0	1
298	73	68	0	1
299	74	65	3	2
300	74	63	0	1
301	75	62	1	1
302	76	67	0	1
303	77	65	3	1
304	78	65	1	2
305	83	58	2	2

306 rows × 4 columns

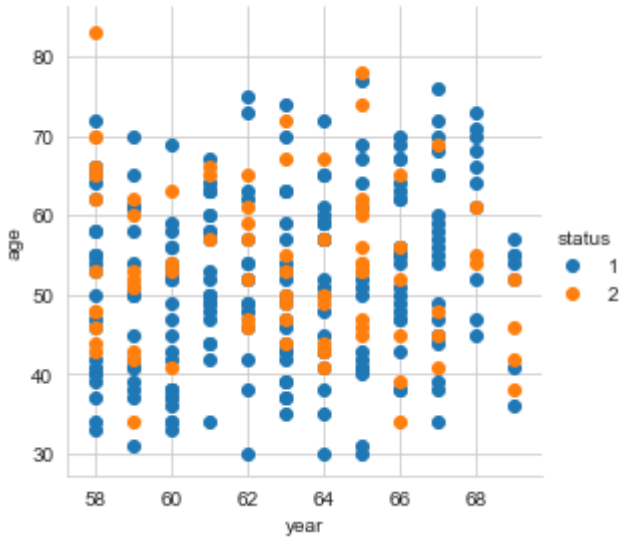
In [18]:

```
#2D scatter plot
df.plot(kind='scatter',x='year',y='age')
plt.show()

#Observations
#from the following figure, we can conclude the average patient's ages that belongs to
 cancer as well as the year in which they have gone through the cancer treatment
```



In [19]:

```
#using seaborn
sns.set_style("whitegrid")
sns.FacetGrid(df,hue="status",height=4).map(plt.scatter,"year","age").add_legend()
plt.show()
df['status'].value_counts()

#Observations:
# 1. out of total patients there are 225 patients having status 1 and 81 patients havin
g status 2
# 2. here we can also identify the approx result of the ages of patient with years havi
ng success status i.e 1 or 2 (since the dots of status 1 and 2 are overlapping so we ca
n't get the accurate result)

#Note: According to habermans's file, status 1 means the patient survived 5yrs or longe
r but the patient having status 2 means the patient died within 5year
```



Out[19]:

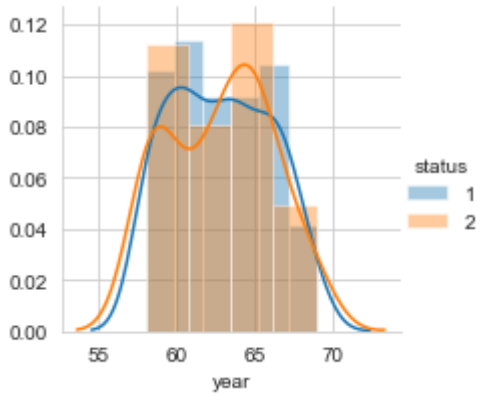
```
1    225
2     81
Name: status, dtype: int64
```

In [21]:

```
#using pair plot
plt.close()
sns.set_style("whitegrid")
sns.pairplot(df,hue="status",height=3)
plt.show()

#Obsevations:
 # Here are some figures having coordinate axis names (status,age),(status,year),(statu
s,nodes),(age,status),(year,status),(nodes,status) through which we can create models u
sing if-else condition
 # Also the figure having coordinate axis name (nodes,nodes) is suitable for finding pd
f(probability density function) of status 1 and other figures with pdf are overlapping.
```

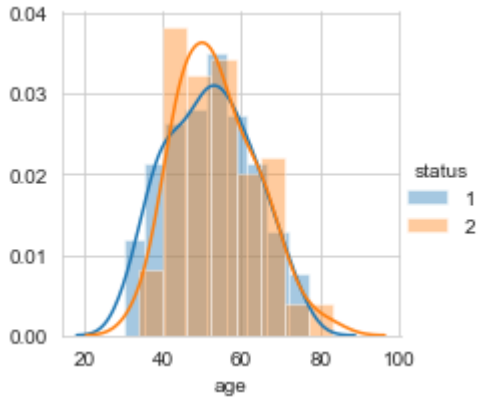
In [22]:

```
sns.FacetGrid(df,hue="status",height=3).map(sns.distplot,"year").add_legend()
plt.show()

#Observations
##From the following figure, we can seee that, on the year 60 the success status i.e 1
 is high but the year between 64 and 65 the success status is 2 i.e the patient died mo
re within these period according to habermans's file.
```



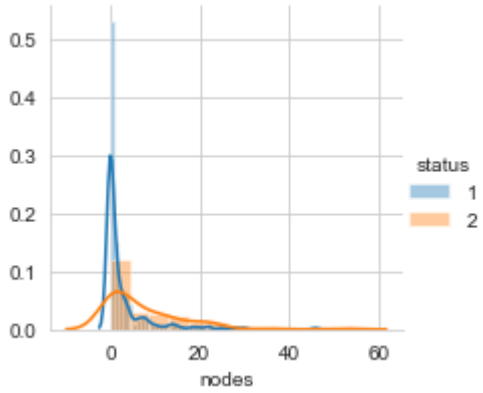
In [23]:

```
sns.FacetGrid(df,hue="status",height=3).map(sns.distplot,"age").add_legend()
plt.show()

#Observations
##From the following figure, we can conclude that patients belonging to the age 45 to 5
0 have less success status i.e 2(patient have died more between these ages)
```



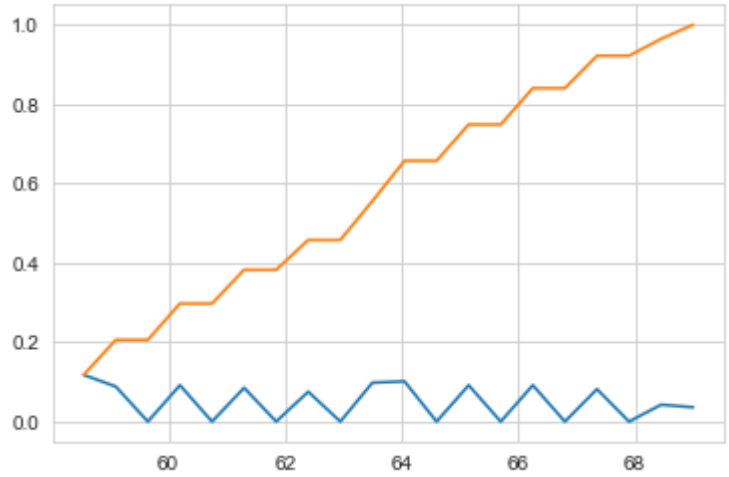
In [24]:

```
sns.FacetGrid(df,hue="status",height=3).map(sns.distplot,"nodes").add_legend()
plt.show()

#Observations
##From the following figure, we can conclude that the patient with nodes 0 is having mo
re success status 1
```



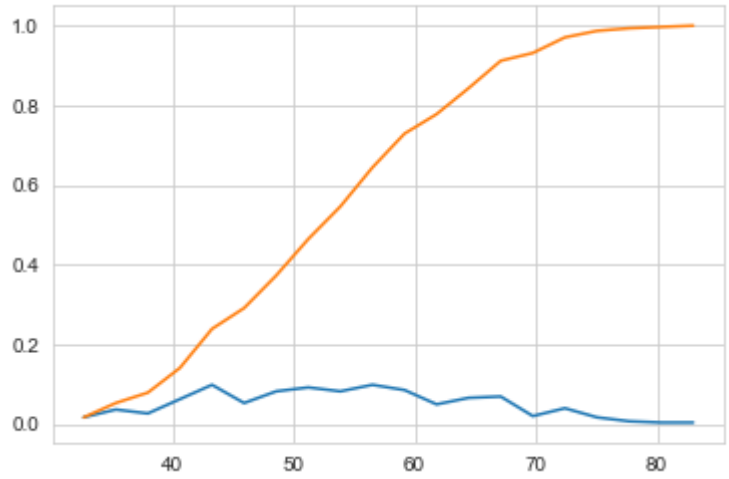
In [36]:

```
total_count, bin_edges = np.histogram(df["year"], bins = 20, density = True)
pdf = total_count/(sum(total_count))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)

plt.show()

#Observations
## From the following figure, we can conclude the percentage of patient gone through th
e cancer treatment with respect to the years using cdf and pdf.
###For eg:
    # 63.5% of total patient have gone through the cancer operation from the year 58 to
64
    # 96% of the patient have gone through the cancer treatment from the year 58 to 68
```



In [39]:

```
total_count, bin_edges = np.histogram(df["age"], bins = 20, density = True)
pdf = total_count/(sum(total_count))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)


plt.show()

#Observations
##From the following figure, we can identify the percentage of the different ages of pa
tient who have gone through cancer treatment in the hospital
```



In [8]:

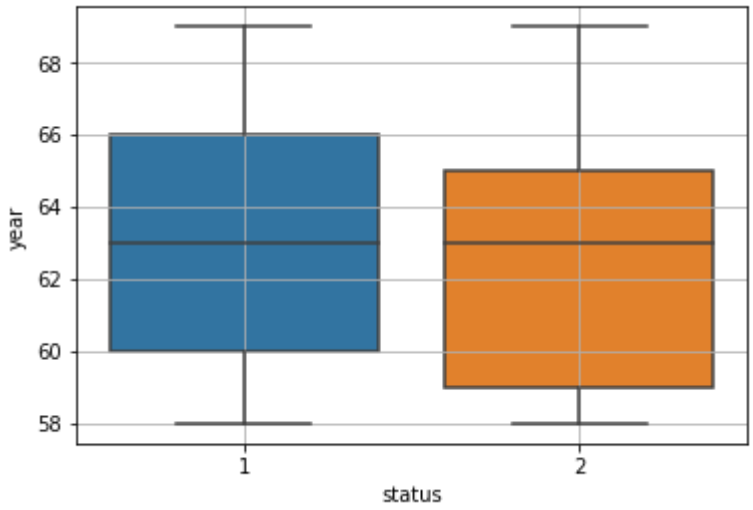
```
#----------code to find the mean value starts-----------
df_status1 = np.mean(df[df["status"]==1])
print("---------mean value of age, year, nodes for status 1------------")
print(df_status1)

df_status2 = np.mean(df[df["status"]==2])
print("---------mean value of age, year, nodes for status 2------------")
print(df_status2)
#----------code to find the mean value ends-------------

#----------code to find the standard value starts-----------
df_std_status1 = np.std(df[df["status"]==1])
print("---------standard deviation, i.e the average deviation of the points from the me
an value for status 1------------")
print(df_std_status1)

df_std_status2 = np.std(df[df["status"]==2])
print("---------standard deviation, i.e the average deviation of the points from the me
an value for status 2------------")
print(df_std_status2)
#----------code to find the standard value ends-----------

##Following are the mean value, standard deviations of all the columns for status 1 and
status 2
```

```
---------mean value of age, year, nodes for status 1------------
age       52.017778
year      62.862222
nodes      2.791111
status     1.000000
dtype: float64
---------mean value of age, year, nodes for status 2------------
age       53.679012
year      62.827160
nodes      7.456790
status     2.000000
dtype: float64
---------standard deviation, i.e the average deviation of the points from
the mean value for status 1------------
age       10.987655
year       3.215745
nodes      5.857258
status     0.000000
dtype: float64
---------standard deviation, i.e the average deviation of the points from
the mean value for status 2------------
age       10.104182
year       3.321424
nodes      9.128776
status     0.000000
dtype: float64
```

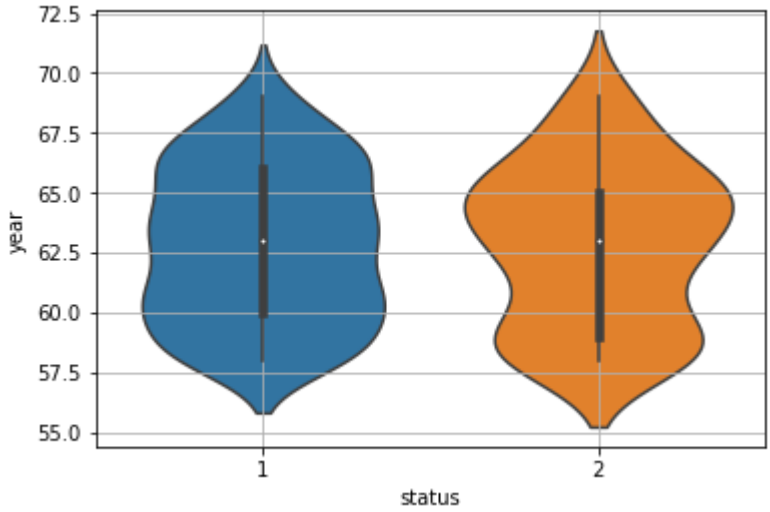
In [16]:

```
sns.boxplot(x="status",y="year",data = df)
plt.grid()
print("-------------box plot----------")
plt.show()

sns.violinplot(x="status",y="year",data = df, size = 10)
print("----------violin plot-----------")
plt.grid()
plt.show()
#Observations
##From the following figure, we can conclude that there are 25 percentile of the patien
t form the year 58 to 60 having status 1
## and there are 25 percentile of the patient from the year 58 to 59 having status 2
```

```
-------------box plot----------
```




```
----------violin plot-----------
```



In [ ]: