

```
In [1]: import pandas as pd
import seaborn as sns

In [7]: heart=pd.read_csv('Heart_Disease.csv')
heart

Out[7]:
```

	sex	age	education	smokingStatus	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	CHDRisk
0	male	39	4	no	0	0	0	0	no	195	106.0	70.0	26.97	80	77	no
1	female	46	2	no	0	0	0	0	no	250	121.0	81.0	28.73	95	76	no
2	male	48	1	yes	20	0	0	0	no	245	127.5	80.0	25.34	75	70	no
3	female	61	3	yes	30	0	0	1	no	225	150.0	95.0	28.58	65	103	yes
4	female	46	3	yes	23	0	0	0	no	285	130.0	84.0	23.10	85	85	no
...
3669	male	54	3	yes	25	0	0	0	no	208	137.5	82.5	25.58	75	63	yes
3670	male	54	3	yes	25	0	0	0	no	208	137.5	82.5	25.58	75	63	yes
3671	male	54	3	yes	25	0	0	0	no	208	137.5	82.5	25.58	75	63	yes
3672	male	54	3	yes	25	0	0	0	no	208	137.5	82.5	25.58	75	63	yes
3673	male	50	2	yes	25	0	0	0	no	208	137.5	82.5	25.97	69	68	yes

3674 rows x 16 columns

```
In [8]: #1.Data Understanding
heart.head()

Out[8]:
```

	sex	age	education	smokingStatus	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	CHDRisk
0	male	39	4	no	0	0	0	0	no	195	106.0	70.0	26.97	80	77	no
1	female	46	2	no	0	0	0	0	no	250	121.0	81.0	28.73	95	76	no
2	male	48	1	yes	20	0	0	0	no	245	127.5	80.0	25.34	75	70	no
3	female	61	3	yes	30	0	0	1	no	225	150.0	95.0	28.58	65	103	yes
4	female	46	3	yes	23	0	0	0	no	285	130.0	84.0	23.10	85	85	no

```
In [9]: #1.Data Understanding
heart.describe()
```

```
Out[9]:
```

	age	education	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	totChol	sysBP	diaBP	BMI	heartRate	glucose
count	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000	3674.000000
mean	49.577300	1.984213	9.092270	0.030212	0.005716	0.310016	236.761840	132.38024	82.906505	25.783038	75.719652	81.769461
std	8.546068	1.022891	11.938399	0.171194	0.075307	0.462563	44.039295	22.04683	11.948024	4.056048	11.957171	23.884454
min	32.000000	1.000000	0.000000	0.000000	0.000000	0.000000	113.000000	83.50000	48.000000	15.540000	44.000000	40.000000
25%	42.000000	1.000000	0.000000	0.000000	0.000000	0.000000	206.000000	117.00000	75.000000	23.080000	68.000000	71.000000
50%	49.000000	2.000000	0.000000	0.000000	0.000000	0.000000	234.000000	128.00000	82.000000	25.400000	75.000000	78.000000
75%	56.000000	3.000000	20.000000	0.000000	0.000000	1.000000	263.000000	143.50000	89.500000	27.990000	82.000000	87.000000
max	70.000000	4.000000	70.000000	1.000000	1.000000	1.000000	600.000000	295.00000	142.500000	56.800000	143.000000	394.000000

```
In [10]: #2.Initial check up
heart.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3674 entries, 0 to 3673
Data columns (total 16 columns):
 #   Column              Non-Null Count  Dtype
---  --
 0   sex                 3663 non-null   object
 1   age                 3674 non-null   int64
 2   education           3674 non-null   int64
 3   smokingStatus       3661 non-null   object
 4   cigsPerDay          3674 non-null   int64
 5   BPMeds              3674 non-null   int64
 6   prevalentStroke     3674 non-null   int64
 7   prevalentHyp        3674 non-null   int64
 8   diabetes            3674 non-null   object
 9   totChol             3674 non-null   float64
10   sysBP              3674 non-null   float64
11   diaBP              3674 non-null   float64
12   BMI                 3674 non-null   float64
13   heartRate           3674 non-null   int64
14   glucose             3674 non-null   int64
15   CHDRisk             3674 non-null   object
dtypes: float64(4), int64(8), object(4)
memory usage: 459+ KB
```

```
In [11]: #Adding Questions
#1.sex wise heart disease
#2.education wise heart disease
#3.CHDRisk wise heart disease
#4.heartRate wise diabetes
#5.sex wise smokingStatus
#6.diaBP wise CHDRisk
#7.education wise prevalentHyp
#8.education wise diabetes
#9.heartRate wise glucose

In [35]: #1.sex wise heart disease
heart['sex'].value_counts(normalize=True)*100
```

```
Out[35]:
```

sex	count	proportion
female	55.446355	15.090259
male	44.553645	12.109741

```
In [36]: #2.education wise heart disease
heart['education'].value_counts(normalize=True)*100
```

```
Out[36]:
```

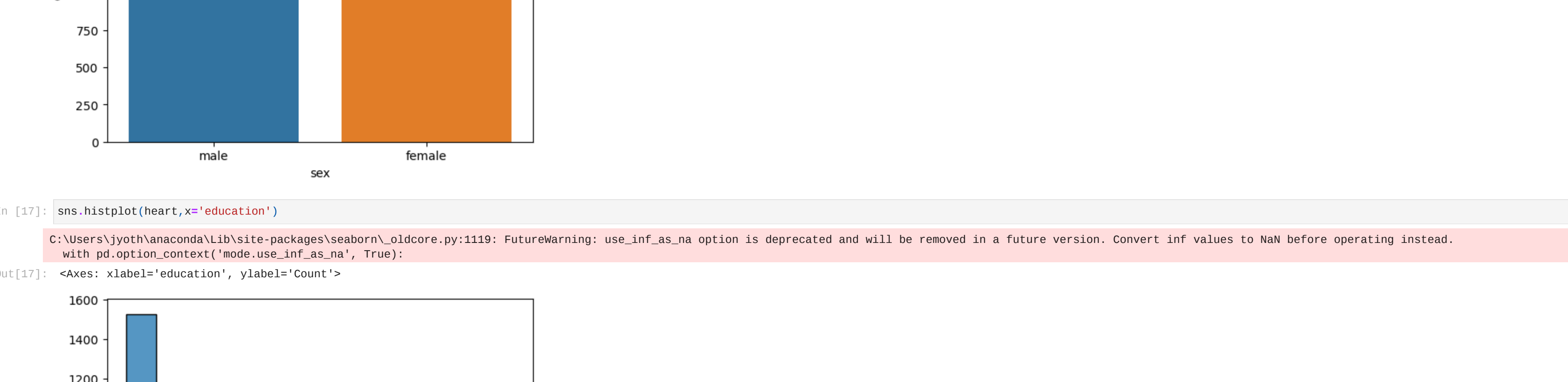
education	count	proportion
1	41.562330	11.309555
2	29.994556	8.156199
3	15.902559	4.330881
4	11.549555	3.145346

```
In [37]: #3.CHDRisk wise heart disease
heart['CHDRisk'].value_counts(normalize=True)*100
```

```
Out[37]:
```

CHDRisk	count	proportion
no	84.349483	22.954656
yes	15.650517	4.245344

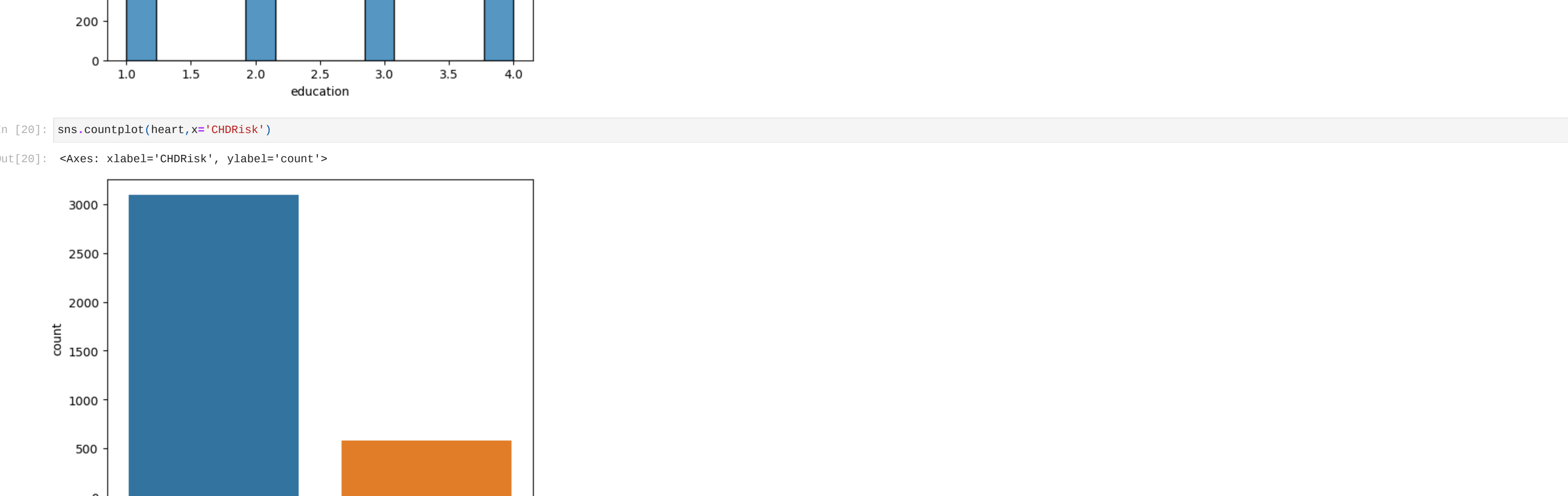
```
In [16]: sns.countplot(heart,x='sex')
```



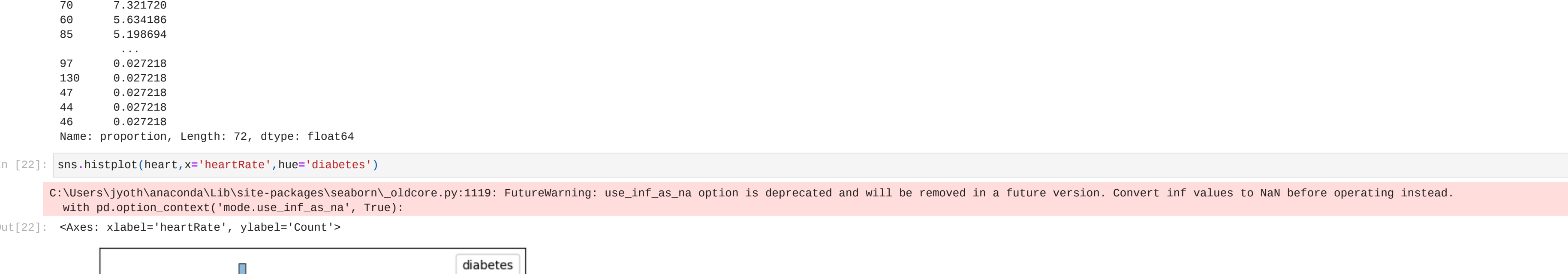
```
In [17]: sns.histplot(heart,x='education')

C:\Users\jyoth\anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):

Out[17]: <Axes: xlabel='education', ylabel='count'>
```



```
In [20]: sns.countplot(heart,x='CHDRisk')
```



```
In [38]: #4.heartRate wise diabetes
heart['heartRate'].value_counts(normalize=True)*100

Out[38]:
```

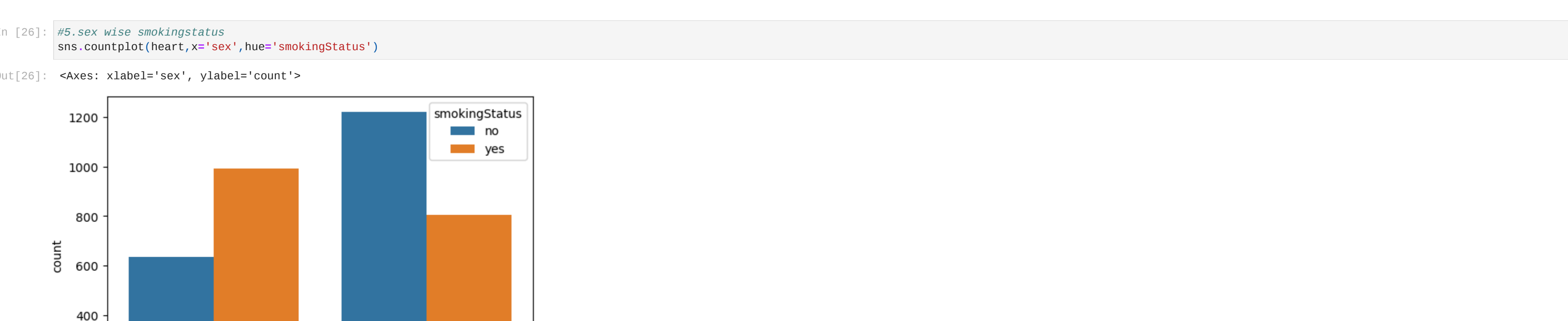
heartRate	count	proportion
75	14.207948	0.383581
80	9.145346	0.248919
70	7.321720	0.199556
60	5.634186	0.153308
85	5.198694	0.141500
...
97	0.027218	0.000741
130	0.027218	0.000741
47	0.027218	0.000741
44	0.027218	0.000741
46	0.027218	0.000741

```
Name: proportion, Length: 72, dtype: float64
```

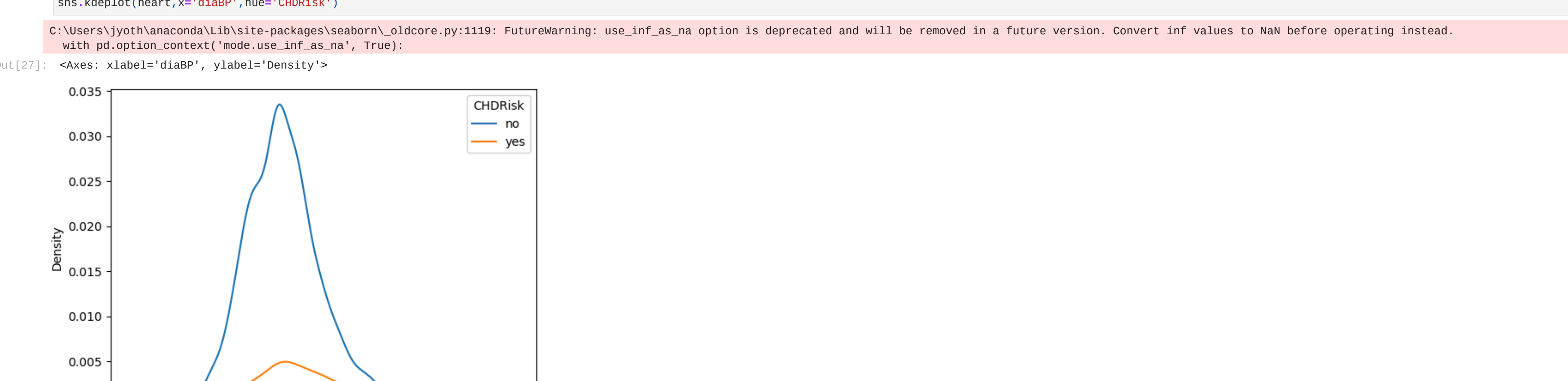
```
In [22]: sns.histplot(heart,x='heartRate',hue='diabetes')

C:\Users\jyoth\anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):

Out[22]: <Axes: xlabel='heartRate', ylabel='count'>
```



```
In [26]: #5.sex wise smokingStatus
sns.countplot(heart,x='sex',hue='smokingStatus')
```



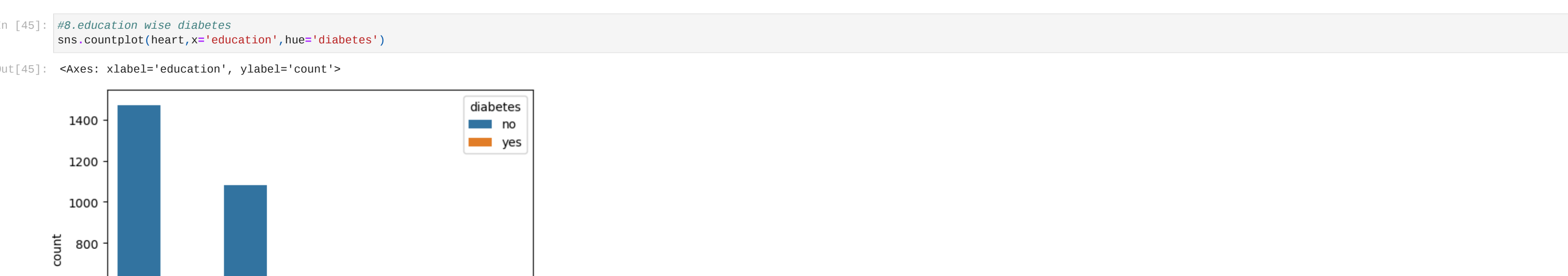
```
In [27]: #6.diaBP wise CHDRisk
sns.kdeplot(heart,x='diaBP',hue='CHDRisk')

C:\Users\jyoth\anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):

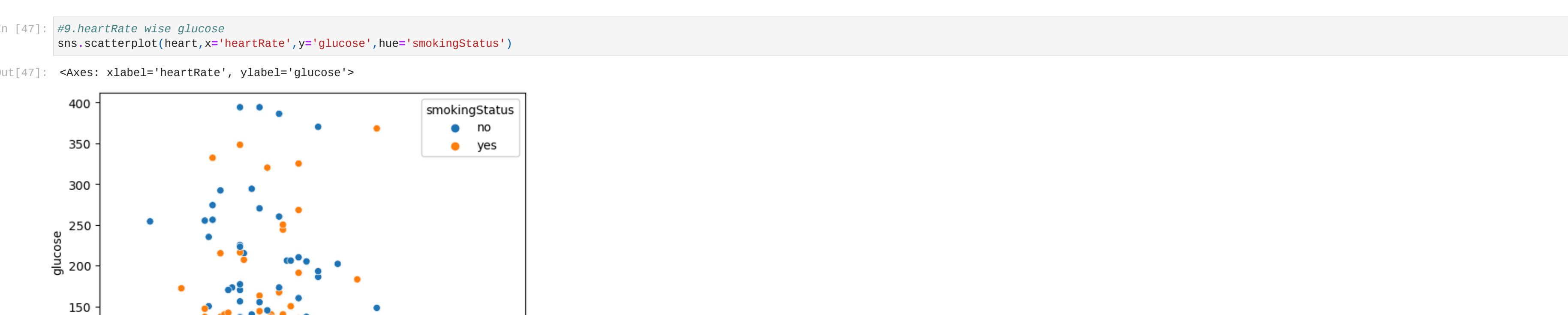
Out[27]: <Axes: xlabel='diaBP', ylabel='Density'>
```



```
In [31]: #7.education wise prevalentHyp
sns.histplot(heart,x='education',hue='prevalentHyp')
```



```
In [45]: #8.education wise diabetes
sns.countplot(heart,x='education',hue='diabetes')
```



```
In [47]: #9.heartRate wise glucose
sns.scatterplot(heart,x='heartRate',y='glucose',hue='smokingStatus')
```



```
In [ ]: #Insights
#55% female and 44% male suffering from heart disease
#education status 1 in 42% 2 in 29% 3 in 18% 4 in 11% of people
#84% are not facing CHDrisk and 15% are facing CHDrisk disease
#1600 male and 2000 females are suffering from heart disease
#1500 students is high in education
#78% of heartrate is high in above 500
#600 males are not smoking 1000 males are smoking and 1200 females are smoking but 800 females are not smoking
#90% are not facing diabp and 10% are facing diabp
#prevalenthyp is more in 2.8
#diabetes is more in in 1 education status
#smokingstatus is high in heartrate
#above 3000 people are not facing CHDrisk but above 500 people are facing CHDrisk
```