

```
In [1]: import pandas as pd
import numpy as np
from sklearn import preprocessing
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)
import warnings
warnings.simplefilter(action='ignore')
```

```
In [2]: df=pd.read_csv(r"C:\Users\Lenovo\OneDrive\Desktop\Data Sets\framingham.csv")
df
```

Out[2]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0
...
4233	1	50	1.0	1	1.0	0.0	0	1	0	313.0	179.0
4234	1	51	3.0	1	43.0	0.0	0	0	0	207.0	126.5
4235	0	48	2.0	1	20.0	NaN	0	0	0	248.0	131.0
4236	0	44	1.0	1	15.0	0.0	0	0	0	210.0	126.5
4237	0	52	2.0	0	0.0	0.0	0	0	0	269.0	133.5

4238 rows × 16 columns

```
In [3]: df.head()
```

Out[3]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	dial
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0	71
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0	81
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5	81
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0	91
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0	81

```
In [4]: df.tail()
```

Out[4]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP
4233	1	50	1.0	1	1.0	0.0	0	1	0	313.0	179.0
4234	1	51	3.0	1	43.0	0.0	0	0	0	207.0	126.5
4235	0	48	2.0	1	20.0	NaN	0	0	0	248.0	131.0
4236	0	44	1.0	1	15.0	0.0	0	0	0	210.0	126.5
4237	0	52	2.0	0	0.0	0.0	0	0	0	269.0	133.5

```
In [7]: df.shape
```

```
Out[7]: (4238, 16)
```

```
In [8]: df.describe()
```

```
Out[8]:
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes
count	4238.000000	4238.000000	4133.000000	4238.000000	4209.000000	4185.000000	4238.000000	4238.000000	4238.000000
mean	0.429212	49.584946	1.978950	0.494101	9.003089	0.029630	0.005899	0.310524	0.141250
std	0.495022	8.572160	1.019791	0.500024	11.920094	0.169584	0.076587	0.462763	0.348145
min	0.000000	32.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	42.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	49.000000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	1.000000	56.000000	3.000000	1.000000	20.000000	0.000000	0.000000	1.000000	0.000000
max	1.000000	70.000000	4.000000	1.000000	70.000000	1.000000	1.000000	1.000000	1.000000

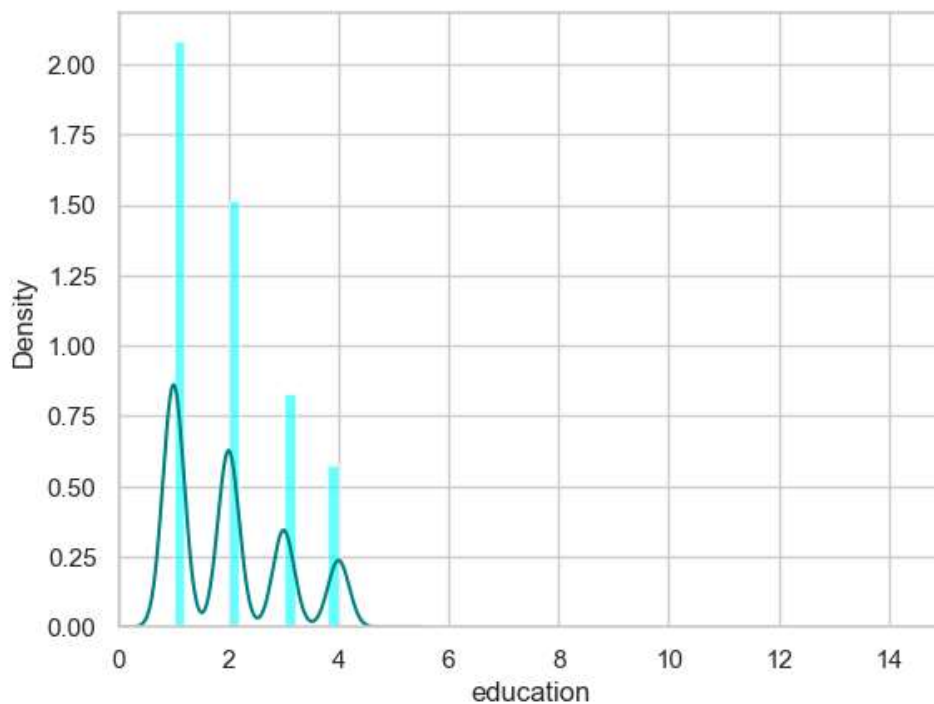
```
In [9]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to 4237
Data columns (total 16 columns):
#   Column                Non-Null Count  Dtype  
---  -
0   male                   4238 non-null  int64  
1   age                    4238 non-null  int64  
2   education              4133 non-null  float64
3   currentSmoker          4238 non-null  int64  
4   cigsPerDay              4209 non-null  float64
5   BPMeds                  4185 non-null  float64
6   prevalentStroke         4238 non-null  int64  
7   prevalentHyp            4238 non-null  int64  
8   diabetes                4238 non-null  int64  
9   totChol                 4188 non-null  float64
10  sysBP                   4238 non-null  float64
11  diaBP                   4238 non-null  float64
12  BMI                     4219 non-null  float64
13  heartRate               4237 non-null  float64
14  glucose                 3850 non-null  float64
15  TenYearCHD              4238 non-null  int64  
dtypes: float64(9), int64(7)
memory usage: 529.9 KB
```

```
In [10]: df.isna().any()
```

```
Out[10]: male                False
age                False
education           True
currentSmoker       False
cigsPerDay          True
BPMeds              True
prevalentStroke     False
prevalentHyp        False
diabetes            False
totChol             True
sysBP               False
diaBP               False
BMI                 True
heartRate           True
glucose             True
TenYearCHD          False
dtype: bool
```

```
In [11]: ax=df["education"].hist(bins=15,density=True,stacked=True,color='cyan',alpha=0.6)
df["education"].plot(kind='density',color='teal')
ax.set(xlabel='education')
plt.xlim(-0,15)
plt.show()
```



```
In [12]: print(df["education"].mean(skipna=True))
print(df["education"].median(skipna=True))
```

```
1.9789499153157513
2.0
```

```
In [13]: print(df['glucose'].isnull().sum()/df.shape[0]*100)
```

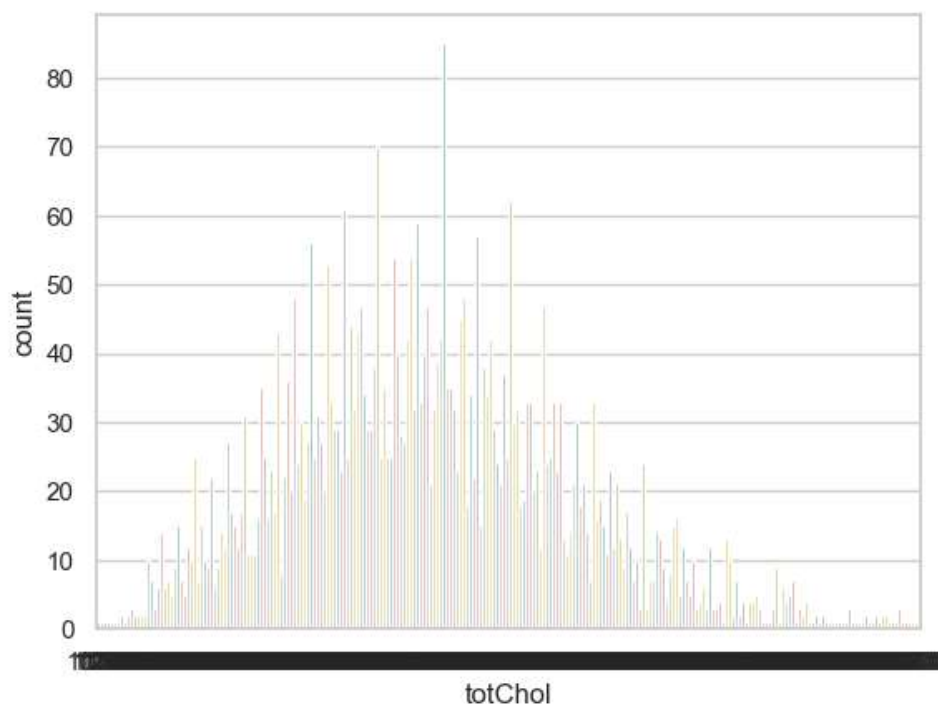
```
9.155261915998112
```

```
In [14]: print(df['totChol'].isnull().sum()/df.shape[0]*100)
```

```
1.1798017932987257
```

```
In [15]: print(df['totChol'].value_counts())
sns.countplot(x='totChol',data=df,palette='Set2')
plt.show()
```

```
totChol
240.0    85
220.0    70
260.0    62
210.0    61
232.0    59
..
392.0     1
405.0     1
359.0     1
398.0     1
119.0     1
Name: count, Length: 248, dtype: int64
```



```
In [16]: print(df['totChol'].value_counts().idxmax())
```

```
240.0
```

```
In [17]: data=df.copy()
data["education"].fillna(df["education"].median(skipna=True),inplace=True)
data["totChol"].fillna(df["totChol"].value_counts().idxmax(),inplace=True)
data.drop('glucose',axis=1,inplace=True)
```

```
In [18]: data.isna().sum()
```

```
Out[18]: male          0
age              0
education        0
currentSmoker    0
cigsPerDay       29
BPMeds           53
prevalentStroke  0
prevalentHyp     0
diabetes         0
totChol          0
sysBP            0
diaBP            0
BMI              19
heartRate        1
TenYearCHD       0
dtype: int64
```

```
In [19]: pd.set_option('display.max_rows',4238)
pd.set_option('display.max_columns',16)
```

```
In [20]: pd.set_option('display.width',50)
```

```
In [21]: print('This DataFrame has %d Rows and %d Columns'%(df.shape))
```

This DataFrame has 4238 Rows and 16 Columns

```
In [22]: features_matrix=df.iloc[:,0:15]
target_vector=df.iloc[:,-2]
```

```
In [23]: print('The Features Matrix Has %d Rows And %d Column(s)'%(features_matrix.shape))
print('The Target Matrix Has %d Rows And %d Column(s)'%(np.array(target_vector).reshape(-1,1).shape))
```

The Features Matrix Has 4238 Rows And 15 Column(s)
The Target Matrix Has 4238 Rows And 1 Column(s)

```
In [24]: df['education'].mean()
```

```
Out[24]: 1.9789499153157513
```

```
In [25]: df['cigsPerDay'].mean()
```

```
Out[25]: 9.003088619624615
```

```
In [28]: df['heartRate'].median()
```

```
Out[28]: 75.0
```

```
In [29]: df['BPMeds'].mean()
```

```
Out[29]: 0.02962962962962963
```

```
In [30]: df["glucose"].fillna(df["glucose"].median(skipna=True),inplace=True)
df
```

Out[30]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0
5	0	43	2.0	0	0.0	0.0	0	1	0	228.0	180.0
6	0	63	1.0	0	0.0	0.0	0	0	0	205.0	138.0
7	0	45	2.0	1	20.0	0.0	0	0	0	313.0	100.0
8	1	52	1.0	0	0.0	0.0	0	1	0	260.0	141.5
9	1	43	1.0	1	30.0	0.0	0	1	0	225.0	162.0
10	0	50	1.0	0	0.0	0.0	0	0	0	254.0	133.0

```
In [31]: df.isnull().sum()
```

```
Out[31]: male          0
age          0
education     105
currentSmoker 0
cigsPerDay    29
BPMeds        53
prevalentStroke 0
prevalentHyp  0
diabetes      0
totChol       50
sysBP         0
diaBP         0
BMI           19
heartRate     1
glucose       0
TenYearCHD    0
dtype: int64
```

```
In [32]: df['education'].fillna(df['education'].median(skipna=True),inplace=True)
df['totChol'].fillna(df['totChol'].median(skipna=True),inplace=True)
df['BMI'].fillna(df['BMI'].median(skipna=True),inplace=True)
df['heartRate'].fillna(df['heartRate'].median(skipna=True),inplace=True)
df['BPMeds'].fillna(df['BPMeds'].median(skipna=True),inplace=True)
df['cigsPerDay'].fillna(df['cigsPerDay'].median(skipna=True),inplace=True)
```

```
In [33]: df.isna().any()
```

```
Out[33]: male           False
age             False
education       False
currentSmoker   False
cigsPerDay      False
BPMeds          False
prevalentStroke False
prevalentHyp    False
diabetes        False
totChol         False
sysBP           False
diaBP           False
BMI             False
heartRate       False
glucose         False
TenYearCHD      False
dtype: bool
```

```
In [34]: df.drop('glucose',axis=1,inplace=True)
```

```
In [35]: df.isna().any()
```

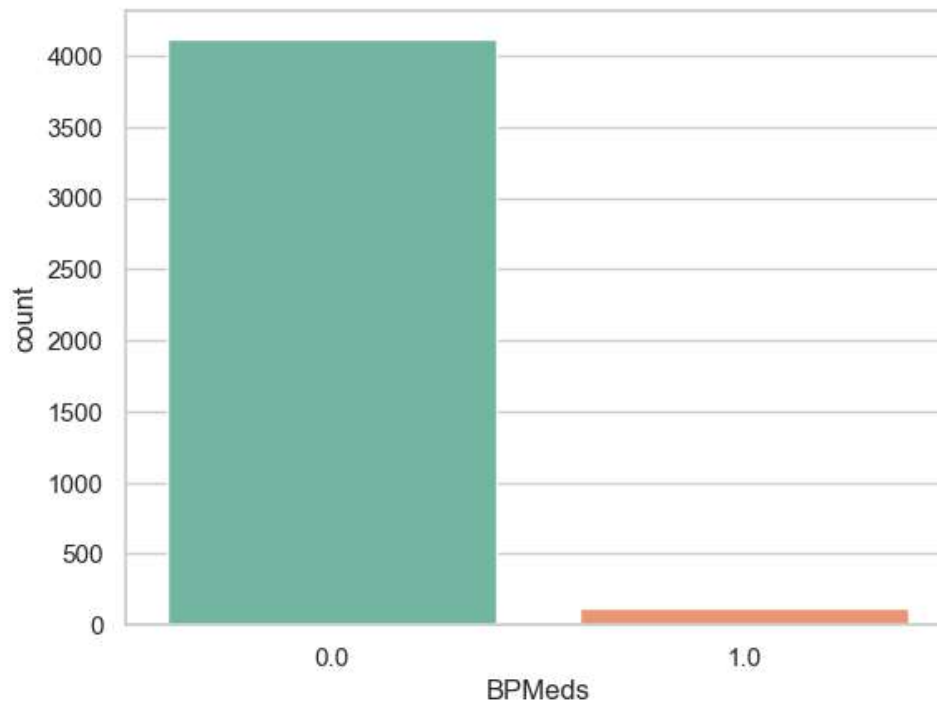
```
Out[35]: male           False
age             False
education       False
currentSmoker   False
cigsPerDay      False
BPMeds          False
prevalentStroke False
prevalentHyp    False
diabetes        False
totChol         False
sysBP           False
diaBP           False
BMI             False
heartRate       False
TenYearCHD      False
dtype: bool
```

```
In [36]: print(df["cigsPerDay"].mean(skipna=True))
print(df["cigsPerDay"].median(skipna=True))
print((df['BPMeds'].isnull().sum()/df.shape[0]*100))
print((df['BMI'].isnull().sum()/df.shape[0]*100))
print((df['heartRate'].isnull().sum()/df.shape[0]*100))
```

```
8.941481831052384
0.0
0.0
0.0
0.0
```

```
In [37]: print(df['BPMeds'].value_counts())  
sns.countplot(x='BPMeds',data=df,palette='Set2')  
plt.show()
```

```
BPMeds  
0.0    4114  
1.0     124  
Name: count, dtype: int64
```

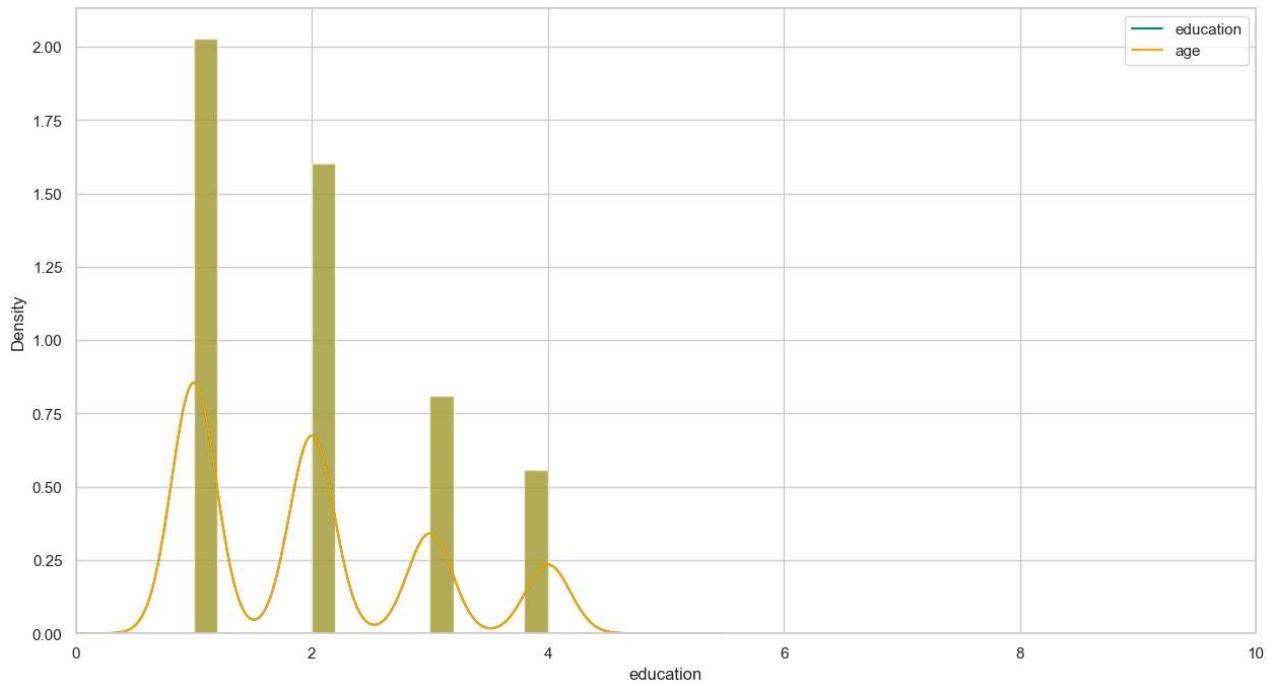


```
In [38]: print(df['heartRate'].value_counts().idxmax())
```

```
75.0
```



```
In [39]: plt.figure(figsize=(15,8))
ax=df["education"].hist(bins=15,density=True,stacked=True,color='teal',alpha=0.6)
df["education"].plot(kind='density',color='teal')
ax=data["education"].hist(bins=15,density=True,stacked=True,color='orange',alpha=0.5)
data["education"].plot(kind='density',color='orange')
ax.legend(["education","age"])
ax.set(xlabel='education')
plt.xlim(-0,10)
plt.show()
```



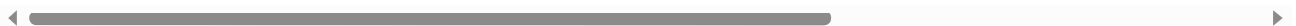
```
In [40]: data['Disease']=np.where((data["prevalentHyp"]+data["prevalentStroke"])>0,0,1)
data.drop('prevalentHyp',axis=1,inplace=True)
data.drop('prevalentStroke',axis=1,inplace=True)
```

```
In [41]: training=pd.get_dummies(data,columns=["currentSmoker","totChol","sysBP"])
training.drop('TenYearCHD',axis=1,inplace=True)
training.drop('male',axis=1,inplace=True)
training.drop('diaBP',axis=1,inplace=True)
final_train=training
final_train.head()
```

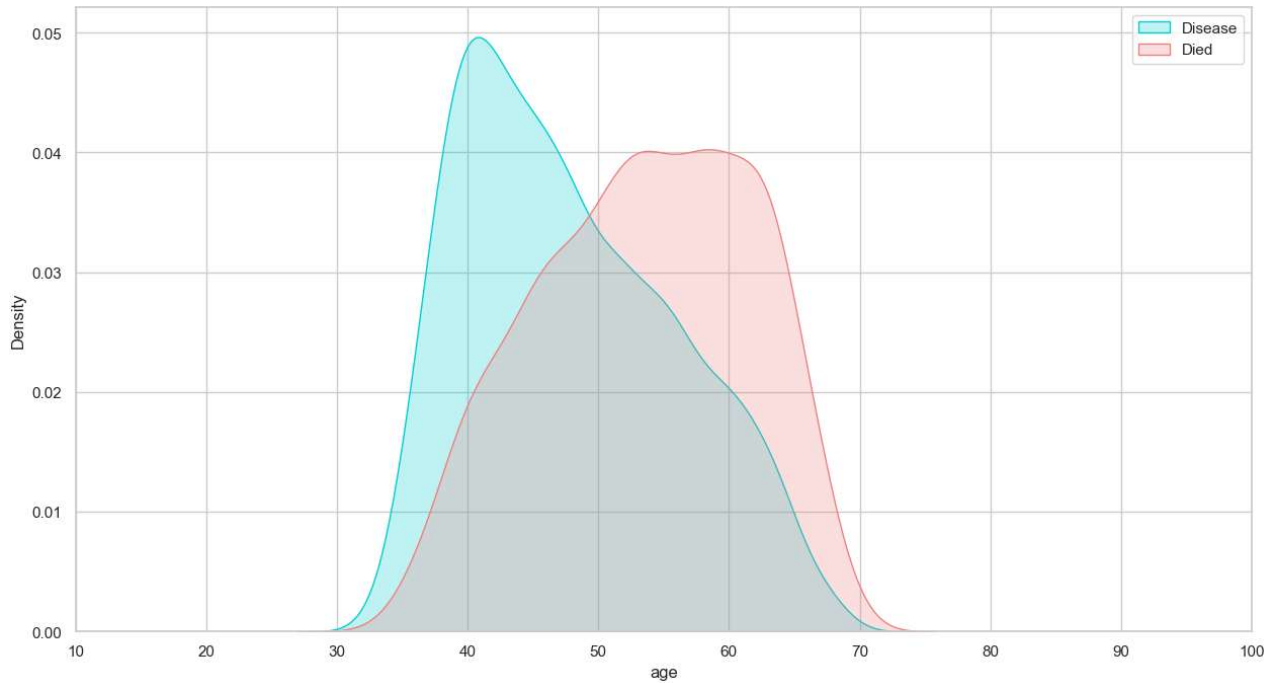
Out[41]:

	age	education	cigsPerDay	BPMeds	diabetes	BMI	heartRate	Disease	...	sysBP_220.0	sysBP_230.0	sysBP_232.0	s
0	39	4.0	0.0	0.0	0	26.97	80.0	1	...	False	False	False	
1	46	2.0	0.0	0.0	0	28.73	95.0	1	...	False	False	False	
2	48	1.0	20.0	0.0	0	25.34	75.0	1	...	False	False	False	
3	61	3.0	30.0	0.0	0	28.58	65.0	0	...	False	False	False	
4	46	3.0	23.0	0.0	0	23.10	85.0	1	...	False	False	False	

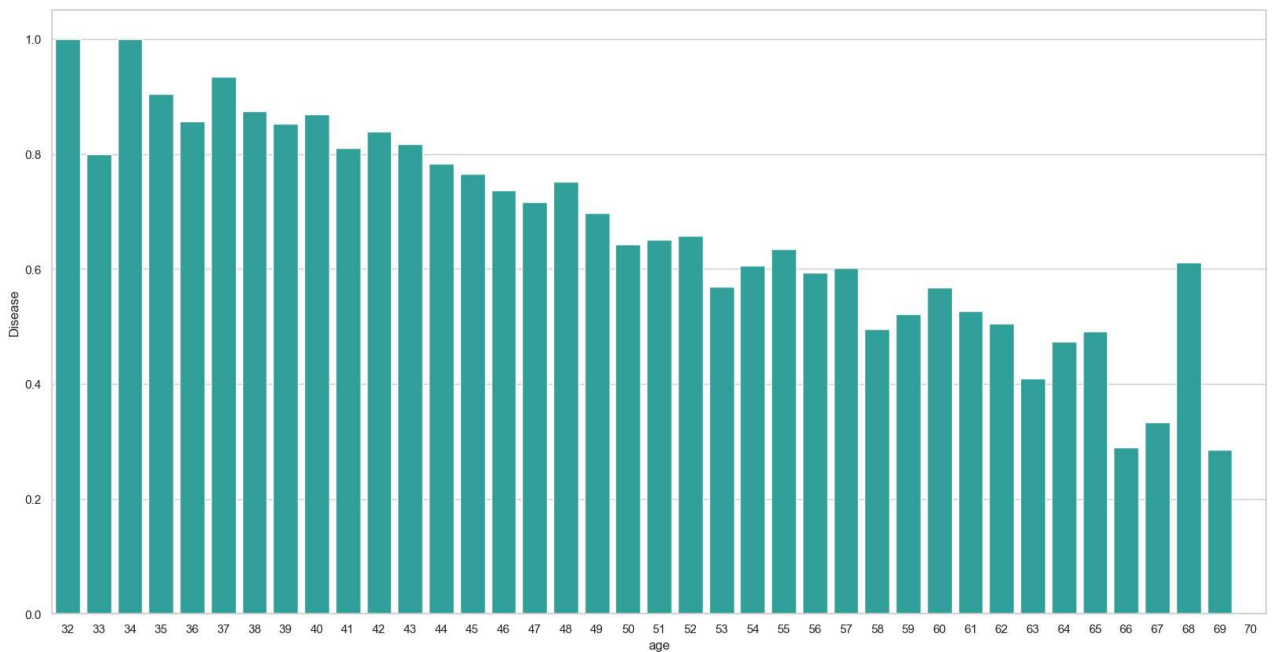
5 rows × 492 columns



```
In [42]: plt.figure(figsize=(15,8))
ax = sns.kdeplot(final_train["age"][final_train.Disease == 1],color="darkturquoise",shade=True)
sns.kdeplot(final_train["age"][final_train.Disease == 0],color="lightcoral",shade=True)
plt.legend(['Disease', 'Died'])
ax.set(xlabel='age')
plt.xlim(10,100)
plt.show()
```



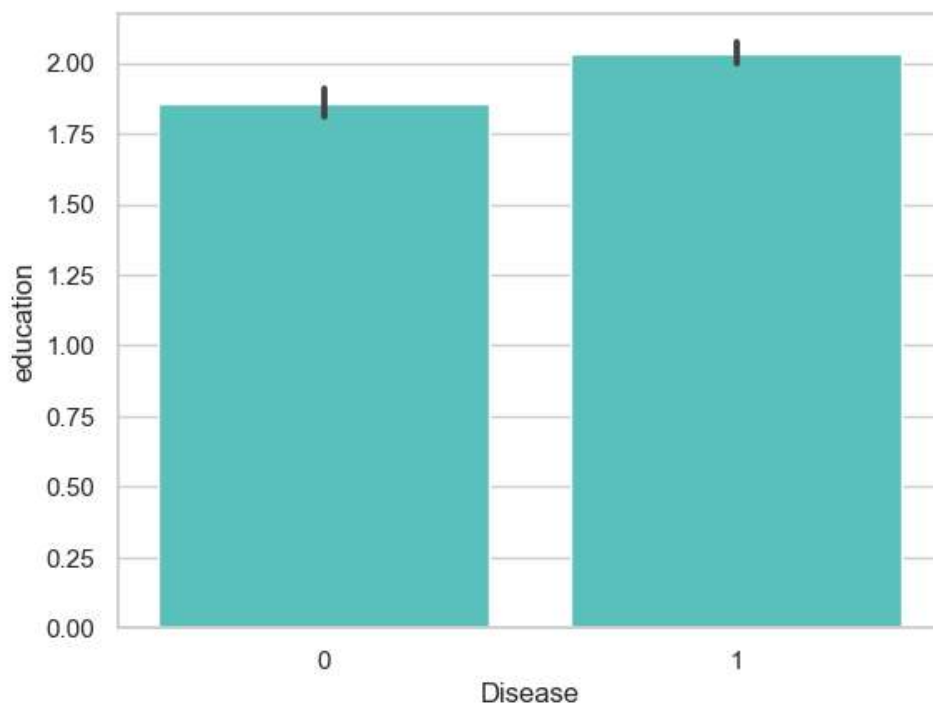
```
In [43]: plt.figure(figsize=(20,10))
avg_survival_byage=final_train[["age", "Disease"]].groupby(['age'],as_index=False).mean()
g=sns.barplot(x='age',y='Disease',data=avg_survival_byage,color="LightSeaGreen")
plt.show()
```



```
In [44]: final_train['IsMinor']=np.where(final_train['age']<=16,1,0)
print(final_train['IsMinor'])
```

```
0      0
1      0
2      0
3      0
4      0
5      0
6      0
7      0
8      0
9      0
10     0
11     0
12     0
13     0
14     0
15     0
16     0
17     0
18     0
~      ~
```

```
In [45]: sns.barplot(x='Disease',y='education',data=final_train,color="mediumturquoise")
plt.show()
```



```
In [46]: import seaborn as sns
import matplotlib.pyplot as plt
sns.barplot(x='diabetes', y='age', data=df, color="aquamarine")
plt.show()
```

