Import Libries and Dataset.

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

Reading Dataset

```
data=pd.read_csv('CHD_preprocessed.csv')
```

Display Top Five Rows of The Dataset.

```
data.head()
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHy
0	1	39	1	0	0.0	0.0	0	
1	0	46	0	0	0.0	0.0	0	
2	1	48	0	1	20.0	0.0	0	
3	0	61	1	1	30.0	0.0	0	
4	0	46	1	1	23.0	0.0	0	

Oops.. No translation found.

Display Last 5 Rows of Dataset

```
data.tail()
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	pr
4128	1	50	0	1	1.0	0.0	0	
4129	1	51	1	1	43.0	0.0	0	
4130	0	48	0	1	20.0	0.0	0	
4131	0	44	0	1	15.0	0.0	0	
4132	0	52	0	0	0.0	0.0	0	

Find Shape of The Dataset.

```
data.shape
```

```
(4133, 16)
```

```
print("Number of column",data.shape[0])
print("Number Rows",data.shape[1])
```

```
Number of column 4133
Number Rows 16
```

Getting Information from Dataset.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4133 entries, 0 to 4132
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   male             4133 non-null   int64  
 1   age              4133 non-null   int64  
 2   education        4133 non-null   int64  
 3   currentSmoker    4133 non-null   int64  
 4   cigsPerDay       4133 non-null   float64
 5   BPMeds           4133 non-null   float64
 6   prevalentStroke  4133 non-null   int64  
 7   prevalentHyp     4133 non-null   int64  
 8   diabetes         4133 non-null   int64  
 9   totChol          4133 non-null   float64
 10  sysBP            4133 non-null   float64
 11  diaBP            4133 non-null   float64
 12  BMI              4133 non-null   float64
 13  heartRate        4133 non-null   float64
 14  glucose          4133 non-null   float64
 15  TenYearCHD       4133 non-null   int64  
dtypes: float64(8), int64(8)
memory usage: 516.8 KB
```

Check Null Values in Dataset.

```
data.isnull()
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	
0	False	False	False	False	False	False	False	
1	False	False	False	False	False	False	False	
2	False	False	False	False	False	False	False	
3	False	False	False	False	False	False	False	
4	False	False	False	False	False	False	False	
...	...	...	...	...	...	...	...	
4128	False	False	False	False	False	False	False	
4129	False	False	False	False	False	False	False	
4130	False	False	False	False	False	False	False	
4131	False	False	False	False	False	False	False	
4132	False	False	False	False	False	False	False	

4133 rows × 16 columns

```
data.isnull().sum()
```

```
male               0
age                0
education          0
currentSmoker      0
cigsPerDay         0
BPMeds             0
prevalentStroke    0
```

```
prevalentHyp       0
diabetes           0
totChol            0
sysBP              0
diaBP              0
BMI                0
heartRate          0
glucose            0
TenYearCHD         0
dtype: int64
```

Checking Duplicates Data and Droping Them.

```
data_dup=data.duplicated().any()
print(data_dup)
```

```
False
```

Get Overall Statistics of Dataset.

```
data.describe()
```

	male	age	education	currentSmoker	cigsPerDay	BPMed
count	4133.000000	4133.000000	4133.000000	4133.000000	4133.000000	4133.00000
mean	0.427293	49.557222	0.280668	0.494798	9.101621	0.03435
std	0.494745	8.561628	0.449380	0.500033	11.918440	0.18216
min	0.000000	32.000000	0.000000	0.000000	0.000000	0.00000
25%	0.000000	42.000000	0.000000	0.000000	0.000000	0.00000
50%	0.000000	49.000000	0.000000	0.000000	0.000000	0.00000
75%	1.000000	56.000000	1.000000	1.000000	20.000000	0.00000
max	1.000000	70.000000	1.000000	1.000000	70.000000	1.00000

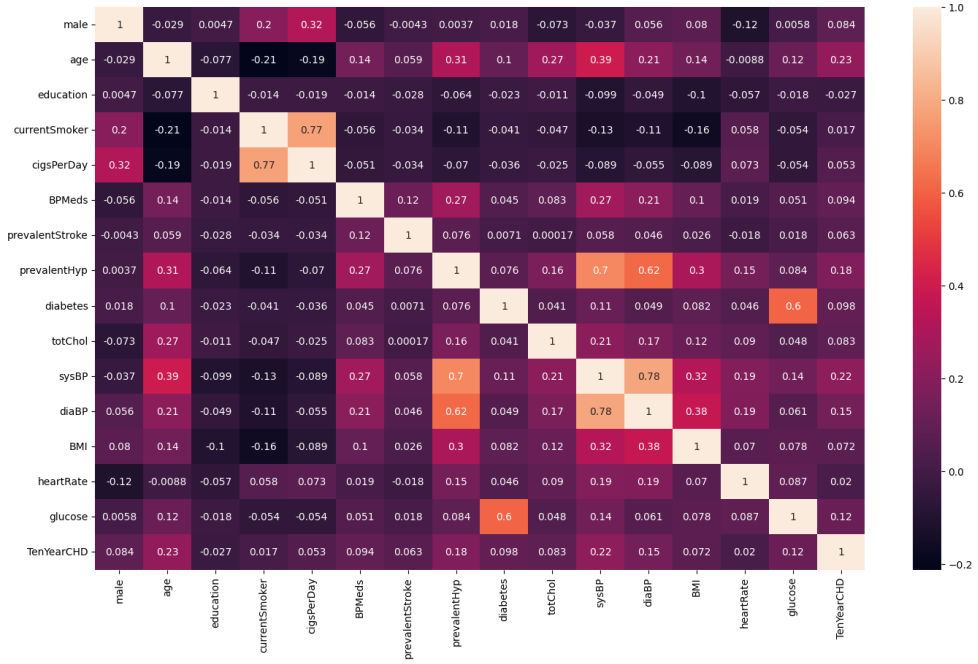
Data Correlation Matrix.

```
data.corr()
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds
male	1.000000	-0.029085	0.004725	0.199750	0.320773	-0.055519
age	-0.029085	1.000000	-0.076576	-0.212415	-0.192079	0.142893
education	0.004725	-0.076576	1.000000	-0.013964	-0.018521	-0.014353
currentSmoker	0.199750	-0.212415	-0.013964	1.000000	0.771739	-0.056488
cigsPerDay	0.320773	-0.192079	-0.018521	0.771739	1.000000	-0.050877
BPMeds	-0.055519	0.142893	-0.014353	-0.056488	-0.050877	1.000000
prevalentStroke	-0.004304	0.058712	-0.027895	-0.033515	-0.033658	0.122337
prevalentHyp	0.003700	0.309546	-0.063900	-0.105899	-0.069803	0.272050

```
plt.figure(figsize=(17,10))
sns.heatmap(data.corr(),annot=True)
```

```
<Axes: >
```



How Many Have Smoker and How Many Not:

```
data.columns
```

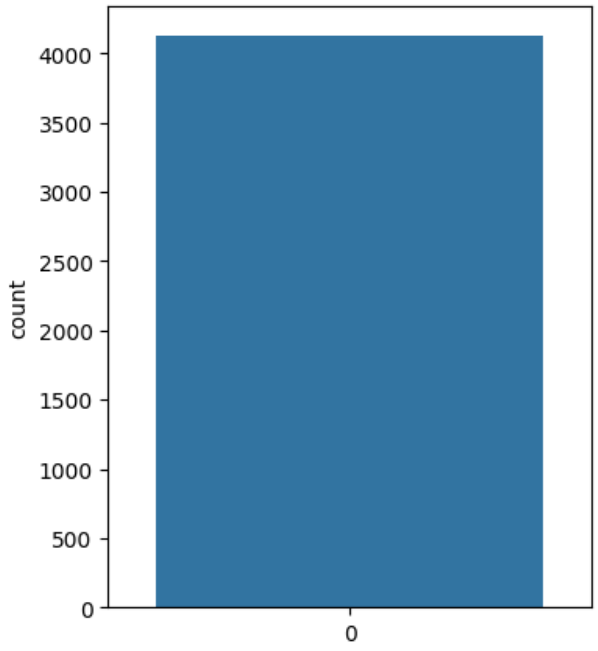
```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

```
data['currentSmoker'].value_counts()
```

```
0    2088
1    2045
Name: currentSmoker, dtype: int64
```

```
plt.figure(figsize=(4,5))
sns.countplot(data['currentSmoker'])
```

```
<Axes: ylabel='count'>
```



Find Count of Male and Female in Dataset.

```
data.columns
```

```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

```
data['male'].value_counts()
```

```
0    2367
1    1766
Name: male, dtype: int64
```

```
sns.countplot(data['male'])
plt.show()
```

```
<Axes: ylabel='count'>
```

Check Age Distribution.

```
sns.distplot(data['age'])
plt.show()
```

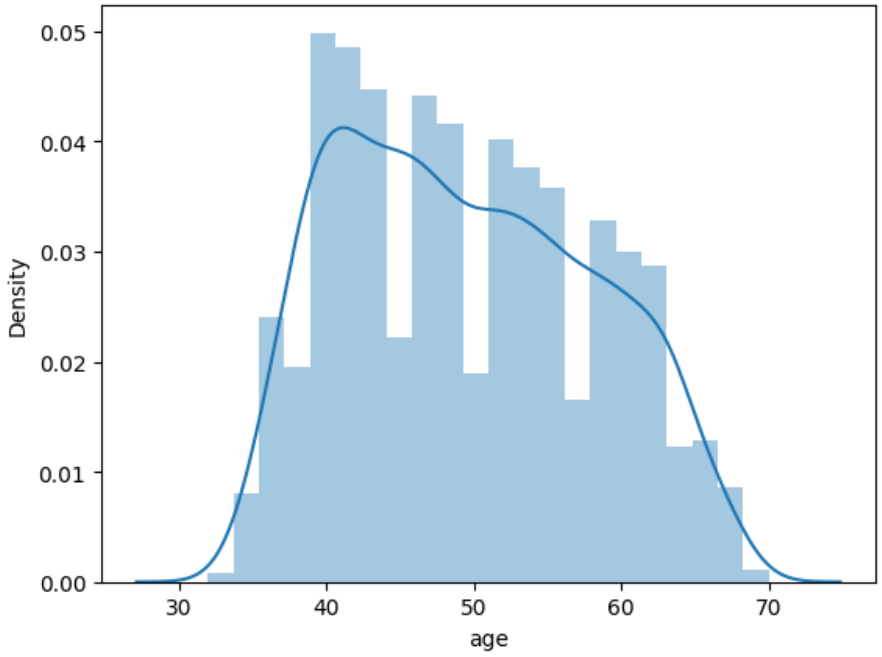
```
<ipython-input-44-dbfeb16865c8>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(data['age'])
```



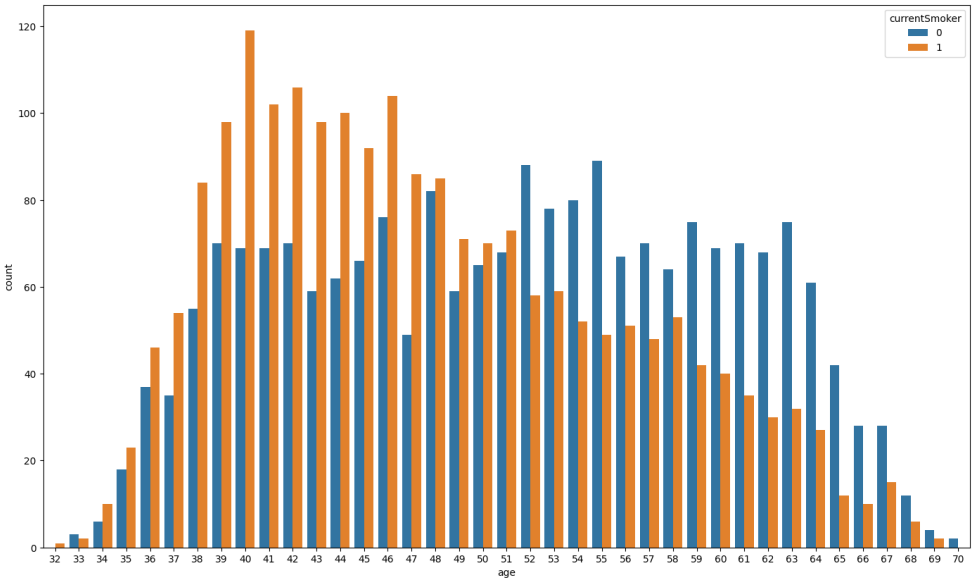
Show Age Distribution As Per CurrentSmoker.

```
data.columns
```

```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

```
plt.figure(figsize=(17,10))
sns.countplot(x="age",hue="currentSmoker",data=data)
#plt.legend(lavels=["Smoker","Non Smokers"])
```

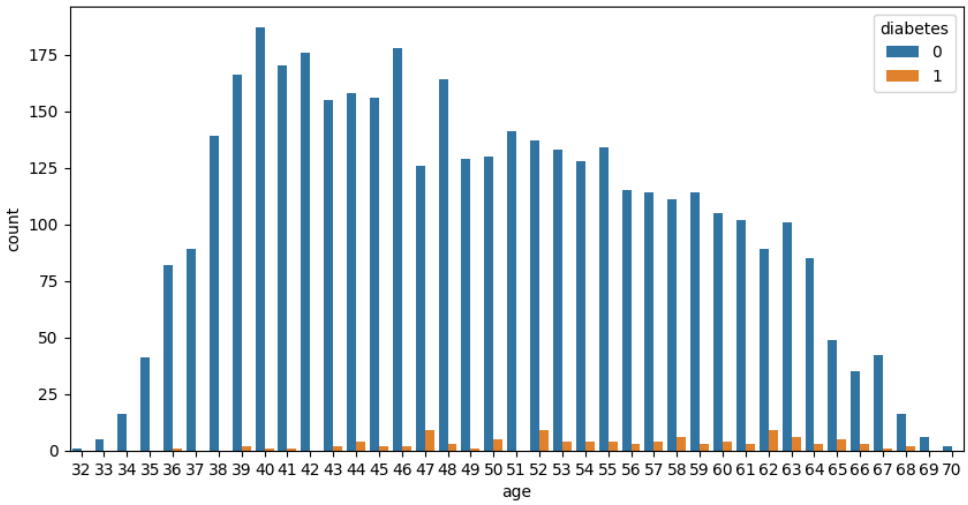
```
<Axes: xlabel='age', ylabel='count'>
```



Show Age Distribution as Per Diabetes.

```
plt.figure(figsize=(10,5))
sns.countplot(x="age",hue="diabetes",data=data)
```

```
<Axes: xlabel='age', ylabel='count'>
```



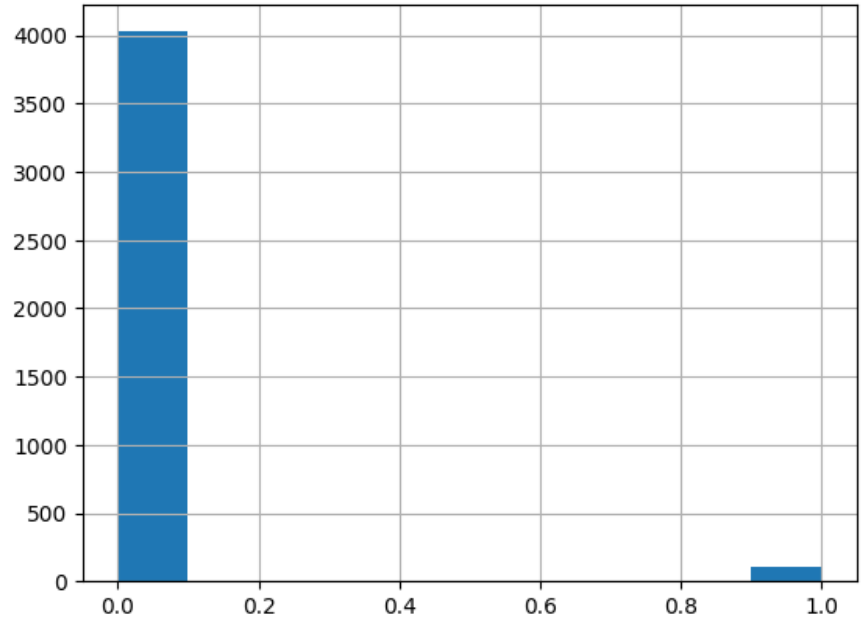
Check Diabetes Distribution

```
data.columns
```

```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

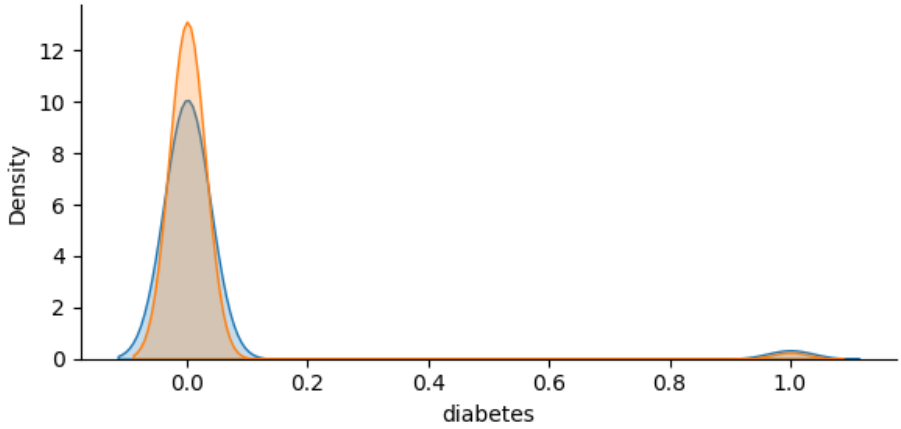
```
data['diabetes'].hist()
```

```
<Axes: >
```



Compare Diabetes as per Smokers

```
g=sns.FacetGrid(data,hue="currentSmoker",aspect=2)
g.map(sns.kdeplot,'diabetes',fill=True)
plt.show()
#plt.legend(labels=['Smoker','Non Smoker'])
```



Plot Continues Variables.

```
data.columns
```

```
Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds',
       'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP',
       'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'],
      dtype='object')
```

```
cate_val=[]
cont_val=[]

for column in data.columns:
  if data[column].nunique() <=10:
    cate_val.append(column)
  else:
      cont_val.append(column)
```

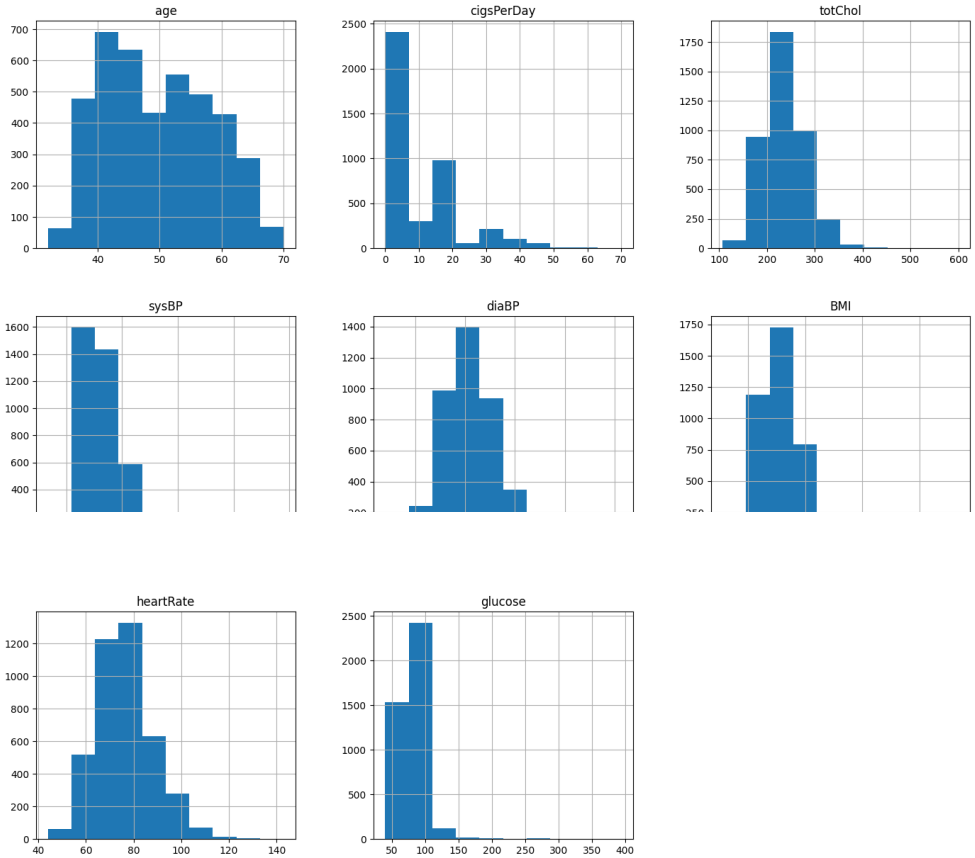
```
cate_val
```

```
['male',
 'education',
 'currentSmoker',
 'BPMeds',
 'prevalentStroke',
 'prevalentHyp',
 'diabetes',
 'TenYearCHD']
```

```
cont_val
```

```
['age',
 'cigsPerDay',
 'totChol',
 'sysBP',
 'diaBP',
 'BMI',
 'heartRate',
 'glucose']
```

```
data.hist(cont_val,figsize=(17,15))
plt.show()
plt.tight_layout()
```




```
<Figure size 640x480 with 0 Axes>
```