In [160	# Importing the requirement number as np import pandas as pd								
In [161	<pre>import pandas as pu import matplotlib.py  # To load the datase  df=pd.read_csv("Hear</pre>	et .	/")						
In [162	Exploratory Data A		t						
Out[162]:	df.head(10)  Gender Age Blood F  Male 70  Female 55	Pressure (mmHg) Cholesto	erol (mg/dL) Has I 262 253	Diabetes Smo No Yes	Never Atypical A	ngina	Treatment  Lifestyle Changes  Angioplasty		
	<ol> <li>Female 55</li> <li>Male 42</li> <li>Male 84</li> <li>Male 86</li> </ol>	95 106 187	253 295 270 296	Yes Yes No Yes	Current Typical A	ngina ngina Coronary Art	Angioplasty Angioplasty ery Bypass Graft (CABG) Medication		
	<ul><li>Female 66</li><li>Male 33</li><li>Male 84</li><li>Male 73</li></ul>	125 181 182 115	271 262 288 286	Yes Yes No Yes	Former Typical A Current Asympto Current Non-angina Never Asympto	matic Pain	Lifestyle Changes Lifestyle Changes Angioplasty		
In [163	9 Female 63 # To see last 10 col	174 Lumns of the dataset	254	Yes	Former Non-angina	Pain	Angioplasty		
Out[163]:	Gender Age Bloom 990 Female 85 991 Female 72	d Pressure (mmHg) Chole 168 194	esterol (mg/dL) Ha 208 181	as Diabetes Sr Yes No		tomatic	Treatment  Lifestyle Changes  Lifestyle Changes		
	<ul><li>992 Female 77</li><li>993 Female 77</li><li>994 Female 51</li></ul>	90 198 107	276 268 217	Yes Yes Yes	Never Non-angi Never Asymp Former Non-angi	nal Pain Coronary <i>i</i> tomatic nal Pain	Artery Bypass Graft (CABG)  Medication  Medication		
	<ul><li>995 Male 42</li><li>996 Male 80</li><li>997 Female 64</li><li>998 Female 84</li></ul>	125 186 108 123	193 267 174 195	Yes Yes Yes No	Never Atypica  Current Non-angi		Angioplasty Artery Bypass Graft (CABG) Artery Bypass Graft (CABG) Lifestyle Changes		
In [164	<pre># To know the shape df.shape</pre>	of the dataset	197	No	Former Atypica	Angina	Lifestyle Changes		
Out[164]: In [165…	(1000 0)	eview about the datan	frame						
	<pre>df.info()  <class #="" 'pandas.core.="" (total="" 1000="" column<="" columns="" data="" ent="" pre="" rangeindex:=""></class></pre>	ries, 0 to 999 8 columns): Non-Null Cou							
	0 Gender 1 Age 2 Blood Pressure 3 Cholesterol (mg 4 Has Diabetes	/dL) 1000 non-nul 1000 non-nul	object int64 int64 int64 int64						
	5 Smoking Status 6 Chest Pain Type 7 Treatment dtypes: int64(3), ob memory usage: 62.6+	1000 non-nul ject(5)	ll object						
In [166 Out[166]:		ge', 'Blood Pressure s', 'Smoking Status'	e (mmHg)', 'Cho						
In [167	<pre># To generate descri df.describe()</pre>	•	the dataset						
Out[167]:	count 1000.000000 mean 60.338000 std 17.317496	1000.000000 145.440000 31.756525	1000.000000 223.789000 42.787817						
	min     30.000000       25%     45.000000       50%     60.500000       75%     76.000000	90.000000 118.000000 146.000000 173.000000	150.000000 185.000000 225.500000 259.000000						
In [168	max 89.000000  # To know the dataty	199.000000 vpe of each dataset	299.000000						
Out[168]:	df.dtypes  Gender  Age Blood Pressure (mmHe) Cholesterol (mg/dL)	int64							
	Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object	object object object object							
In [169 Out[169]:	# To know the how magenderCount=df['GendgenderCount  Female 510		oatients are pi	resent					
In [170	Male 490 Name: Gender, dtype  # To know the count hasDiabetesCount=df[	of patients with Dia							
Out[170]:	hasDiabetesCount  Yes 517 No 483 Name: Has Diabetes,  # To know the count		moking hahit o	r not					
Out[171]:	hasSomkingHabitCount hasSomkingHabitCount Never 352 Current 325	=df['Smoking Status'							
In [172	Former 323 Name: Smoking Status # To know the count hasChestPainCount=df	of type of Chest Pai							
Out[172]:	hasChestPainCount  Non-anginal Pain Asymptomatic Typical Angina Atypical Angina	261 255 243 241	. ) دعانت						
In [173	Name: Chest Pain Ty  # To know count of t  hasTreatmentCount=df hasTreatmentCount	pe, dtype: int64  Type of treatment red		ent					
Out[173]:	Lifestyle Changes Coronary Artery Bypa Angioplasty Medication Name: Treatment, dt	ass Graft (CABG)	269 252 247 232						
In [174	<pre># to know about how df.isnull().sum()</pre>								
Out[174]:	Condor								
In [175	Smoking Status Chest Pain Type Treatment dtype: int64  # Importing label en	0 0 0							
	<pre>from sklearn import # Creating label enc label_encoder = prep</pre>	preprocessing	der()						
In [176 In [177	<pre># Converting the Gen df1['Gender']= label</pre>								
In [178 Out[178]:	df1['Gender'].value_ 0 510 1 490 Name: Gender, dtype								
-	<pre># Converting the Has df1['Has Diabetes']= df1['Has Diabetes'].</pre>	abel_encoder.fit_t			es'])				
Out[180]: In [181	1 517 0 483 Name: Has Diabetes,		ical column to	Numerical					
In [182 Out[182]:	<pre>df1['Smoking Status' df1['Smoking Status' 2 352</pre>		t_transform(df:	1['Smoking	Status'])				
	0 325 1 323 Name: Smoking Status	est Paint Type Catego			1				
In [183	df1['Chest Pain Type	e']= label_encoder.fi	it transform/di	f1['Chast D					
-	2 261	e'].value_counts()	LL_LT ANSTOTIII( u	TIE CHEST P	ain Type'])				
In [184 Out[184]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty	pe, dtype: int64 eatment Categorical o	column to Nume	rical	ain Type'])				
In [184 Out[184]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la df1['Treatment'].val 2 269 1 252	pe, dtype: int64 eatment Categorical o	column to Nume	rical	ain Type'])				
In [184 Out[184]: In [185 In [186 Out[186]:	2 261 0 255 3 243 1 241 Name: Chest Pain Type # Converting the Tree df1['Treatment']= land df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dtypes	pe, dtype: int64  eatment Categorical of  abel_encoder.fit_trand  ue_counts()  ype: int64	column to Nume	rical	ain Type'])				
In [184 Out[184]: In [185 In [186 Out[186]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHe Cholesterol (mg/dL) Has Diabetes Smoking Status	pe, dtype: int64  eatment Categorical of the catego	column to Nume	rical	ain Type'])				
In [184 Out[184]: In [185 In [186 Out[186]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHe) Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form	pe, dtype: int64  eatment Categorical of the catego	column to Nume nsform(df1['Tre	rical eatment'])		nerical column			
In [184 Out[184]: In [185 In [186 Out[186]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHe Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F  0 1 70	pe, dtype: int64  eatment Categorical of abel_encoder.fit_trandue_counts()  ype: int64  int32 int64 int64 int64 int32 int32 int32 int32 int32 int32  n above that all the 5  Pressure (mmHg) Cholester 181	Column to Numeral sform(df1['Tre	rical eatment'])  Diabetes Smo	cansformed into numbers of the control of the contr	Type Treatment 3 2			
In [184 Out[184]:  In [185  In [186 Out[186]:  In [187  Out[187]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHe) Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86	pe, dtype: int64  eatment Categorical of abel_encoder.fit_transcript.  due_counts()  ype: int64  int32 int64 int64 int32	Categorical Coerol (mg/dL) Has I 262 253 295 270 296	rical eatment']) Diabetes Smo	ansformed into nun	Type Treatment			
In [184 Out[184]:  In [185  In [186 Out[186]:  In [187  Out[187]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHecholesterol (mg/dL)) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F  0 1 70 1 0 55 2 1 42 3 1 84	pe, dtype: int64  eatment Categorical of abel_encoder.fit_transcript.  due_counts()  ype: int64  int32 int64 int64 int64 int32	Categorical Co erol (mg/dL) Has I 262 253 295 270	plumn are tr	cansformed into number of the street pain 2 2 0 2 0 2	Type Treatment  3 2 1 0 3 0 1 1			
In [184 Out[184]:  In [185  In [186 Out[186]:  In [187  Out[187]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHe Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84	pe, dtype: int64  catment Categorical of abel_encoder.fit_transcript.  dbel_encoder.fit_transcript.  due_counts()  ype: int64  int32     int64     int64     int32     int32     int32     int32     int32     int32  n above that all the 5  Pressure (mmHg) Cholestor  181     103     95     106     187     125     181     182     115     174	Categorical Complete Categorical	plumn are tropiabetes Smood 1 1 1 0 1 1 0 0	cansformed into numerous characteristics. Chest Pain 2 2 0 2 0 1 0 0 0	Type Treatment  3			
In [184 Out[184]:  In [185  In [187 Out[187]:  In [188 Out[188]:	2 261 0 255 3 243 1 241 Name: Chest Pain Typ # Converting the Tree df1['Treatment']= la df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty df1.dtypes Gender Age Blood Pressure (mmHcholesterol (mg/dL)) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing requried from sklearn.preproceut from the above we can of	pe, dtype: int64  catment Categorical of abel_encoder.fit_transue_counts()  ype: int64  int32 int64 g) int64 int32 int32 int32 int32 int32  n above that all the 5  Pressure (mmHg) Choleston 181 103 95 106 187 125 181 182 115 174  On  Observe that if we wish we w	Categorical Compensation  Categorical Compen	plumn are tropiabetes Smoo  0  1  1  0  1  1	exansformed into numerous changes status. Chest Pain 2 2 2 0 2 0 1 0 0 2 1 1 0 0 0 2 1 1	Type Treatment  3			
In [184 Out[184]:  In [185  In [187 Out[187]:  In [188 Out[188]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dt; df1.dtypes  Gender Age Blood Pressure (mmHcholesterol (mg/dL)) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preprocessaler = StandardScaler # Performing Normali df2=df1.copy()	pe, dtype: int64  atment Categorical of abel_encoder.fit_transue_counts()  ype: int64  int32 int64 int64 int32 int	Categorical Competitions of the control of the cont	cholestero.	cansformed into number of the street states and states are stated as a state of the street states and states are stated as a state of the state of t	Type Treatment  3	Cholesterol(mg/dL)	'Cholesterol (mg/dL)	'11)
In [184 Out[184]:  In [185  In [187 Out[187]:  In [188 Out[188]:	2 261 0 255 3 243 1 241 Name: Chest Pain Typ # Converting the Tree df1['Treatment']= la df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty df1.dtypes Gender Age Blood Pressure (mmHcholesterol (mg/dL)) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10) Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization From the above we can of # Importing requried from sklearn.preprocescaler = StandardScaler # Performing Normali df2=df1.copy() df2[['Age', 'Blood Performing Normali df2=df1.copy() df2['Age', 'Blood Performing Normali df2=df1.copy() df2.head(10)	pe, dtype: int64  Patment Categorical of abel_encoder.fit_transue_counts()  The second of the second	Categorical Composition  Categorical Compositi	plumn are treeted by the seatment of the seatm	cansformed into numbers of the state of the	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',	Cholesterol (mg/dL)	']])
In [184 Out [184]:  In [185  In [186 Out [187]:  In [188 Out [188]:  In [190  In [191	2 261 0 255 3 243 1 241 Name: Chest Pain Typ # Converting the Tree df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty df1.dtypes  Gender Age Blood Pressure (mmHeydL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  From the above we can of # Importing requried from sklearn.preprocescaler = StandardScale # Performing Normali df2=df1.copy() df2[['Age', 'Blood Pender Age Blood Pende	pe, dtype: int64  Patment Categorical of abel_encoder.fit_transue_counts()  Pype: int64  Int32	Categorical Compensions of the control of the contr	plumn are tropial points and the seatment of t	cansformed into number of the street pain 2 2 2 0 2 1 1 Column aler.fit_transform  Smoking Status Chest 2 2 2 0 2 2 0 2 2 1 1 2 2 2 2 2 2 2 2 2	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1	'Cholesterol (mg/dL)	']])
In [184 Out [184]:  In [185  In [186 Out [187]:  In [188 Out [188]:  In [190  In [191	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tree df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dty  df1.dtypes  Gender Age Blood Pressure (mmHecholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preprocescaler = StandardScaler # Performing Normali df2=df1.copy() df2[['Age', 'Blood Potential Company of the shows we can	pe, dtype: int64  catment Categorical of the lencoder.fit_transite	Categorical Complete Column to Numeral Section (Mg/dL) Has I (Column to Mg/dL) Has I (Column to Mg/dL) (Mg/dL)	plumn are tree tree tree tree tree tree tree	eansformed into num  oking Status Chest Pain  2  2  0  1  0  0  2  1  Column  aler.fit_transform  Smoking Status Chest  2  2  0  1  1  0  1  1  1  1  2  2  1  1  1  1  2  1  1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  ment 2 0 0	'Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185  In [186 Out [187]:  In [188 Out [188]:  In [190  In [191	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tree df1['Treatment']= la  df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dt;  df1.dtypes Gender Age Blood Pressure (mmHecholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preprocescaler = StandardSca  # Performing Normali df2=df1.copy() df2[['Age', 'Blood Performing Normali df2=df1.copy() df2[['Age', 'Blood Performing Normali df2=df1.copy() df2-f1.copy() d	pe, dtype: int64  catment Categorical of the catment Categorical of the categorical of th	Categorical Complete	plumn are tropiabetes Smoo  O  1  1  O  1  1  O  1  1  1  O  1  1	cansformed into num  cking Status Chest Pain  2  2  0  1  0  0  2  1  Column  aler.fit_transform  Smoking Status Chest  2  2  0  2  1  0  2  1  0  2  1  0  2  2  1  1  1  1  2  2  2  2  2  1  2  2	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2	'Cholesterol (mg/dL)	']])
In [184 Out [184]:  In [185  In [186 Out [187]:  In [188 Out [188]:  In [190  In [191	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dt; df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing required from sklearn.preproc scaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P df2.head(10)  Gender Age B 0 1 0.558212 1 0 -0.308397 2 1 1.367048 4 1 1.482595 5 0 0.32716 6 1 -1.579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization # Plotting the bar of plt.bar(genderCount. plt.title('Males v/s)	pe, dtype: int64  patment Categorical of the lencoder.fit_transcription and lencoder.fit_transcr	Categorical Composition of the c	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 1 1 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	'1])
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tree df1['Treatment']= la df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmHcholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Performing Normali df2=df1.copy() df2[['Age', 'Blood P df2.head(10)  Gender Age Biood F 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization # Plotting the bar of plt.bar(genderCount. plt.bar(genderCount.	pe, dtype: int64  catment Categorical of the lencoder.fit_transcript and the lencoder.fit_tran	Categorical Composition of the c	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 1 1 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment'].val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes  Gender Age Blood Pressure (mmHcholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preproc scaler = StandardSca  # Performing Normali df2=df1.copy() df2[['Age', 'Blood P  df2.head(10)  Gender Age Blood F 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 3.682595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.bar(genderCount.) plt.titlae('Males ver') plt.tylabel('Count') plt.ylabel('Count')	pe, dtype: int64  catment Categorical of the lencoder.fit_transcript and the lencoder.fit_tran	Categorical Comparison (df1['Tree and color of the color	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 0 0 2 1 1 0 0 0 1 1 0 0 0 2 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la  df1['Treatment']= la  df1['Treatment'] = la  df1['Treatment'] = la  df1['Treatment'] = la  df1['Treatment'] = val  2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes  Gender Age Blood Pressure (mmHcholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing required from sklearn.preprocescaler = StandardSca  # Performing Normali df2=df1.copy() df2[['Age', 'Blood P  df2.head(10)  Gender Age B 0 1 0.558212 1 0 -0.308945 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.59459 3 1 1.367048 4 1 1.482595 7 1 1.367048 4 1 1.482595 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.bar (genderCount.) plt.bar (genderCount.) plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Count') plt.ylabel('Count') plt.show()	pe, dtype: int64  catment Categorical of the lencoder.fit_transcript and the lencoder.fit_tran	Categorical Comparison (df1['Tree and color of the color	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 0 0 2 1 1 0 0 0 1 1 0 0 0 2 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tree df1['Treatment']= la df1['Treatment']. val 2 269 1 252 0 247 3 232 Name: Treatment, dt; df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10) Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing requried from sklearn.preproc scaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P df2.head(10)  Gender Age Bi 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 4 1 1.482595 5 0 0.3153794  Data Visualization # Plotting the bar of plt.bar(genderCount.plt.txlabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Count') plt.show()	pe, dtype: int64  catment Categorical of the lencoder.fit_transcript and the lencoder.fit_tran	Categorical Comparison (df1['Tree and color of the color	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 0 0 2 1 1 0 0 0 1 1 0 0 0 2 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tree df1['Treatment'] = la df1['Treatment'] = val 2 269 1 252 0 247 3 232 Name: Treatment, dt; df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing required from sklearn.preproc scaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P df2.head(10)  Gender Age B 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 9 0 0.153794  Data Visualization # Plotting the bar of plt.bar(genderCount.) plt.title('Males v/s plt.xlabel('Gender') plt.show()  Data Visualization # Plotting the bar of plt.bar(genderCount.) plt.show()  Data Visualization # Plotting the bar of plt.bar(genderCount.) plt.ylabel('Gender') plt.ylabel('Count') plt.show()	pe, dtype: int64  patment Categorical of the catego	Categorical Complete Serol (mg/dL)  Pressure and Colesterol (mg/dL)  Categorical Complete Serol (mg/dL	plumn are tropiabetes Smood 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1	eansformed into num  oking Status Chest Pain 2 2 2 0 1 0 0 2 1 Column aler.fit_transform  Smoking Status Chest 2 2 0 1 1 0 0 0 2 1 1 0 0 2 1 1 0 1 0 0 2 1 1 0 0 0 1 1 0 0 0 2 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [189  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Tyr # Converting the Tree df1['Treatment']= la  df1['Treatment']= val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmHcCholesterol (mg/dL)) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing required from sklearn.preprocescaler = StandardSca # Performing Normali df2=df1.copy() df2['Age', 'Blood P  df2.head(10)  Gender Age B 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 4 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.bar(genderCount.plt.titleb('Gender')) plt.bar(genderCount.plt.titleb('Gender')) plt.ylabel('Count') plt.show()  500 -  400 -	pe, dtype: int64  patment Categorical of the pattern of the patter	Categorical Common (df1['Tree and of the column to Numer and the column to Numer and the column to Numer and Column to Numer a	Diabetes Smo  O  1  1  O  1  1  O  1  1  O  1  1  O  1  1	ransformed into num  oking Status Chest Pain  2 2 0 2 0 1 0 0 2 1  column  aler.fit_transform  Smoking Status Chest  2 2 0 2 0 1 0 0 2 1 0 0 1 0 0 2 1 0 0 0 2 1 0 0 0 0	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Tyl # Converting the Tre df1['Treatment']= la df1['Treatment']- val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization From the above we can observe form ff2=df1.copy() df2[['Age', 'Blood Person of the service of the servi	pe, dtype: int64  patment Categorical of the pattern of the processor of the pattern of the patt	Categorical Composition (df1['Tree and of the color of th	cholestero  Cholestero  Cholestero  Cholestero  Cholestero  1  1  0  1  1  0  1  1  0  1  1  0  1  1	ransformed into num  oking Status Chest Pain  2 2 2 0 1 0 2 1 0 2 1 column  aler.fit_transform  Smoking Status Chest  2 2 0 1 0 0 2 1 0 1 0 0 2 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1['Treatment']= la df1['Treatment']- val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmHcholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10) Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization From the above we can of # Importing required from sklearn.preprocescaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P  df2.head(10)  Gender Age Bi 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization # Plotting the bar of plt.bar(genderCount.plt.titlael('Gender') plt.ylabel('Count') plt.show()  Comp  # Plotting the bar of plt.titlael('Gender') plt.ylabel('Count') plt.show()  Comp  # Plotting the bar of plt.titlael('Count') plt.show()	pee, dtype: int64  patment Categorical of the path of the path of how many Personate of how many Personate of the path of how many Personate of ho	Categorical Composition (df1['Tree and of the color of th	cholestero  Cholestero  Cholestero  Cholestero  Cholestero  1  1  0  1  1  0  1  1  0  1  1  0  1  1	ransformed into num  oking Status Chest Pain  2 2 2 0 1 0 2 1 0 2 1 column  aler.fit_transform  Smoking Status Chest  2 2 0 1 0 0 2 1 0 1 0 0 2 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment']= la df1['Treatment']= la df1['Treatment']= val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood Pressure 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing required from sklearn.preprocess from the above we can obtain the state of	pee, dtype: int64  patment Categorical of the path of the path of how many Personate of how many Personate of the path of how many Personate of ho	Categorical Composition (df1['Tree and of the color of th	cholestero  Cholestero  Cholestero  Cholestero  Cholestero  1  1  0  1  1  0  1  1  0  1  1  0  1  1	ransformed into num  oking Status Chest Pain  2 2 2 0 1 0 2 1 0 2 1 column  aler.fit_transform  Smoking Status Chest  2 2 0 1 0 0 2 1 0 1 0 0 2 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment'] - val  2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood R 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preproc scaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P  df2.head(10)  Gender Age Blood  # Importing requried from sklearn.preproc scaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P  df2.head(10)  Gender Age Blood  # Indication	pee, dtype: int64  patment Categorical of the path of the path of how many Personate of how many Personate of the path of how many Personate of ho	Categorical Composition (df1['Tree and of the color of th	cholestero  Cholestero  Cholestero  Cholestero  Cholestero  1  1  0  1  1  0  1  1  0  1  1  0  1  1	ransformed into num  oking Status Chest Pain  2 2 2 0 1 0 2 1 0 2 1 column  aler.fit_transform  Smoking Status Chest  2 2 0 1 0 0 2 1 0 1 0 0 2 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185  In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1['Treatment']= la df1['	pe, dtype: int64  patment Categorical of the catego	Categorical Composition (df1['Tree and of the color of th	cholestero  Choles	ransformed into num  oking Status Chest Pain  2 2 2 0 1 0 2 1 0 2 1 column  aler.fit_transform  Smoking Status Chest  2 2 0 1 0 0 2 1 0 1 0 0 2 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1	Type Treatment  3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 0 1 3 1 2 2 2 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, 241 Name: Chest Pain Ty, # Converting the Tre df1['Treatment']= la  df1['Treatment'] . val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes  Gender Gender Age Blood Pressure (mmH-Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing required from sklearn.preprocsaler = StandardSca # Performing Normali df2=df1.copy() df2[['Age', 'Blood P df2.head(10)  Gender Age Bi 0 1 0.558212 1 0 -0.308397 2 1 -1.059459 3 1 1.367048 4 1 1.482595 5 0 0.327116 6 1 -1.579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization # Plotting the bar of plt.titlabel('Count') plt.tylabel('Count') plt.show()  **Plotting the bar of plt.tylabel('Count') plt.show()  **Complete Source **Pertorming the bar of plt.tylabel('Count') plt.show()  **Complete Source **Document of the bar of plt.tylabel('Count') plt.show()  **Complete Source **Document of the bar of plt.tylabel('Count') plt.show()  **Complete Source **Document of the bar of plt.tylabel('Count') plt.show()  **Document of the bar of plt.tylabel('Count') plt.show()	pe, dtype: int64  patment Categorical of the Lencoder.fit_transcription and transcription and transcriptio	Categorical Colorer (mg/dL)  erol (mg/dL) Has I  262 253 295 270 296 271 262 288 286 254  erol (mg/dL)  ardScaler  de Pressure and colesterol (mg/dL)  andScaler  de Pressure and colesterol (mg/dL)  and colesterol (mg/dL)  and colorer (mg/dL)	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 2 0 2 1  columns Age, Blood P  1 Column  aler.fit_transform  Smoking Status Chest  2 2 2 0 2 0 2 1 0 2 1 ghtgreen'])	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	Cholesterol (mg/dL)	'11)
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1['Treatment'] = la df1['Treatment']. val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization From the above we can observe form # Importing required from sklearn.preprocedure = StandardSca # Performing Normali df2=[f1.copy()] df2[['Age', 'Blood Performing Normali df1.df2[['Age', 'Blood Performing Normali df1.df2[['Age', 'Blood Performing Normali df2=[f1.copy()] df2[['Age', 'Blood Performing Normali df2=[f1.cop	pe, dtype: int64  natment Categorical of the Lencoder.fit_transcript and the Lencoder.fit_tran	Categorical Complete Section (and colors of the section of the sec	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 0 1 0 0 2 1 column aler.fit_transform  Smoking Status Chest 2 2 0 2 0 1 0 2 1  column aler.fit_transform  Smoking Status Chest  2 1 2 3 4 6 7 7 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1['Treatment'] = la df1['Treatment']. val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization From the above we can observe form # Importing required from sklearn.preprocedure = StandardSca # Performing Normali df2=[f1.copy()] df2[['Age', 'Blood Performing Normali df1.df2[['Age', 'Blood Performing Normali df1.df2[['Age', 'Blood Performing Normali df2=[f1.copy()] df2[['Age', 'Blood Performing Normali df2=[f1.cop	pe, dtype: int64  patment Categorical of the Land and the	Categorical Complete Section (and colors of the section of the sec	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 0 1 0 0 2 1 column aler.fit_transform  Smoking Status Chest 2 2 0 2 0 1 0 2 1  column aler.fit_transform  Smoking Status Chest  2 1 2 3 4 6 7 7 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	2 261 0 255 3 243 1 241 Name: Chest Pain Ty,  # Converting the Tree df1['Treatment'] - Val 2 269 1 252 0 247 3 232 Name: Treatment, dt; df1.dtypes Gender Age Blood Pressure (mmH- cholesterol (mg/dL) Has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood fo 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preprocescaler = StandardSca  # Performing Normali df2=df1.copy() df2['Age', 'Blood Performing Normali df2=df1.copy() df2['Age', 'Blood Performing Normali df2=df1.copy() df2['Age', 'Blood Performing Normali df2=df1.copy() df2.head(10)  Gender Age Bi 0 1 0.558212 1 0 0.308397 2 1 1.367048 4 1 1.482595 5 0 0.32116 6 1 1.1579425 7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.bar (gender Count. plt.title('Males v/s plt.ylabel('Gount') plt.show()  Comp  250 -  400 -  400 -  # Plotting the bar of plt.bar (gender Count. plt.title('Males v/s plt.ylabel('Gount') plt.show()  Comp  250 -  400 -  400 -  400 -  500 -  400 -  500 -  600	pe, dtype: int64  natment Categorical of the Lencoder.fit_transcript and the Lencoder.fit_tran	Categorical Complete Section (and colors of the section of the sec	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 0 1 0 0 2 1 column aler.fit_transform  Smoking Status Chest 2 2 0 2 0 1 0 2 1  column aler.fit_transform  Smoking Status Chest  2 1 2 3 4 6 7 7 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	# Performing Normalization  # Importing required from sklearn.preprocsaler = StandardSca  # Performing Normali df2=df1.copy() df2[['Age', 'Blood Person of the short of the sh	pe, dtype: int64  natment Categorical of the Lencoder.fit_transcript and the Lencoder.fit_tran	Categorical Complete Section (and colors of the section of the sec	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 0 1 0 0 2 1 column aler.fit_transform  Smoking Status Chest 2 2 0 2 0 1 0 2 1  column aler.fit_transform  Smoking Status Chest  2 1 2 3 4 6 7 7 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192	2 261 6 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tre df1['Treatment'] - val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1('Treatment', dt: df1('Treatment', dt: df1('Treatment, dt: df1('Treat	pe, dtype: int64  natment Categorical of the Lencoder.fit_transcript and the Lencoder.fit_tran	Categorical Complete Section (and colors of the section of the sec	cholestero  Choles	ansformed into num  king Status Chest Pain  2 2 2 0 0 1 0 0 2 1 column aler.fit_transform  Smoking Status Chest 2 2 0 2 0 1 0 2 1  column aler.fit_transform  Smoking Status Chest  2 1 2 3 4 6 7 7 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 0 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1['Treatment'] - val 2 269 1 252 0 247 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- has Diabetes Smoking Status Chest Pain Type Treatment dtype: object  We can observe form  df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preproc scaler = Standardsca  # Performing Normali df2=df1.copy() df2[rAge', 'Blood P df2.head(1e)  Gender Age Bi 0 1 0.55812 1 0 0.308397 2 1 1.059459 3 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.bar(genderCount. plt.title('Males v/s plt.ylabel('Geoder') plt.show()  Data Visualization  # Plotting the bar of plt.title('Males v/s plt.ylabel('Geoder') plt.show()  Comp  300 -  300 -  300 -  300 -  300 -  300 -  300 -  300 -  400 -  400 -  400 -  400 -  400 -  50 -  60 -  7 -  7 1 1.367048 8 1 0.731534 9 0 0.153794  Data Visualization  # Plotting the bar of plt.title('Males v/s plt.ylabel('Geoder') plt.show()  Comp  300 -	pe, dtype: int64  patment Categorical of the Catego	Categorical Complete Column to Numer and State Color (Mg/dL) Has I 262 253 295 270 296 271 262 288 286 254 264 264 264 264 264 264 264 264 264 26	plumn are tree places Smoon on the contract of	ansformed into num  king Status Chest Pain  2 2 0 1 0 0 2 1 columns Age, Blood P   I Column  aler.fit_transform  Smoking Status Chest  2 2 0 2 0 1 0 0 2 1 0 0 2 1 ghtgreen'])   lightblue', 'lights  ghtgreen'])	Type Treatment  3	Blood Pressure (mmHg)',  Pent 2 0 0 1 3 1 2 2 0 0 0	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 6 255 3 243 1 241 Name: Chest Pain Ty, # Converting the Tree df1('Treatment') = Nat d	pe, dtype: int64  patment Categorical of the catego	Categorical Commission (df1['Tree insform(df1['Tree insform(df1['Tree insform(df1['Tree insform(df1['Tree insform(df1['Tree insform(df1['Tree insform(df1['Tree insform(df1]')])	plumn are tree places and the state of the s	ansformed into num  king Status Chest Pain  2 2 0 0 1 0 0 2 1 column  Smoking Status Chest  2 2 0 2 0 1 0 0 2 1  ghtgreen'])  lightblue', 'light'  ghtgreen'])	Type Treatment 3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 1 3 1 2 2 0 0 0 d'])	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 9 255 3 243 1 255 3 243 1 241 Name: Chest Pain Ty,  # Converting the Tree df1['Treatment'] - la df1['Treatment'] - la df1['Treatment'] - val 2 269 1 227 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Chalesterolt (mg/dL) Has Databet (mg/dL) Has Databet (mg/dL) Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood F 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing requried from sklearn.preprocessaler = standardsca # Performing Normali df2=df1.copy() df2['Age', 'Blood Pl df2.head(10)  Gender Age Blood F 0 1 0.558212 1 0 0.308397 2 1 1.059459 3 1 1.3670459 4 1 1.3670459 4 1 1.3670459 4 1 1.3670459 4 1 1.3670459 4 1 1.3670459 5 0 1.327145 6 1 1.5736704 8 1 0.731534 9 0 0.153794  Data Visualizatior  # Plotting the bar of plt.bir(igenderCount') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Comp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  # Plotting the bar of plt.tritle('Count') plt.show()  Tomp  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()  # Plotting the bar of plt.tritle('Count') plt.show()  # Plotting the bar of plt.tritle('Ghas ison plt.ylabel('Count') plt.show()	pe, dtype: int64  patment Categorical of the catego	Categorical Commensus of the state of the st	olumn are transported by the seatment of the s	ansformed into numerical status   Chest Pain	Type Treatment 3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 1 3 1 2 2 0 0 0 d'])	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 9 253 1 253 1 253 1 241 Name: Chest Pain Ty,  # Converting the Tree df1['Treatment'] = la df1['Treatment'] = la df1['Treatment'] = val 2 269 1 227 3 232 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmH- Chalesterol (mg/dL) Has Diabeterol (mg/dL) Gender Age Blood for the solid (mg/dL) Gender Age Blood for the solid (mg/dL) Gender Age Blood for the solid (mg/dL)  # 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization  # Importing required from sklearn.preprocescaler = standardsca  # Performing Normali df2=df1.copy() df2['Age', 'Blood Pl df2.head(10)  Gender Age Blood for the solid (mg/dl) Gender Age Blood for the solid (mg/dl) Gender Age Blood for the solid (mg/dl) ff2['Age', 'Blood Pl df2.head(10)  Gender Age Blood for the solid (mg/dl) ff2['Age', 'Blood Pl df2.head(10)  Gender Age Blood for the solid (mg/dl) ff2['Age', 'Blood Pl df2.head(10)  Gender Age Blood for the solid (mg/dl) ff2['Age', 'Blood Pl df2.head(10)  Gender Age Blood for the solid (mg/dl) ff2['Age', 'Blood Pl df2.head(10)  # Plotting the bar or the solid for the solid fo	pe, dtype: int64  patment Categorical of the catego	Categorical Commensus of the state of the st	olumn are transported by the seatment of the s	ansformed into numerical status   Chest Pain	Type Treatment 3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 1 3 1 2 2 0 0 0 d'])	Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 9 253 1 241 Name: Chest Pain Ty, # Converting the Tre df1['Treatment'] = la df1.dtypes Gender Age Blood Pressure (mmhchables) Gender Age Blood Rowning Status Chest Pain Type Treatment dtype: object We can observe form df1.head(10)  Gender Age Blood Rowning Status Chest Pain Type Treatment dtype: object 0 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing requried from sklearn.preproceder = Standardsea # Performing Normalid df2=df1.copy() df2['Age', 'Blood Pressure   Barriage   ff2=df1.copy() df2['Age', 'Blood Pressure   ff2-lost flat ('Gount') ff2-lost flat ('Gount') ff2-lost flat ('Gount') ff3-lost flat ('Gount') ff3-l	pe, dtype: int64  patment Categorical of the catego	Categorical Commensus of the state of the st	olumn are transported by the seatment of the s	ansformed into numerical status   Chest Pain	Type Treatment 3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 1 3 1 2 2 0 0 0 d'])	'Cholesterol (mg/dL)	
In [184 Out [184]:  In [185 In [186 Out [187]:  In [187 Out [187]:  In [190  In [191 Out [191]:  In [192  In [194  In [194	2 261 9 253 3 243 1 241 Name: Chest Pain Ty # Converting the Tre df1['Treatment'] - la df1['Treatment'] - la df1['Treatment'] - val 2 265 9 247 3 247 Name: Treatment, dt: df1.dtypes Gender Age Blood Pressure (mmHcholesterol (mg/dL) Has Diabetes's Smoking Status Chest Pain Type Treatment dtype: object  We can observe form df1.head(10)  Gender Age Blood for 1 70 1 0 55 2 1 42 3 1 84 4 1 86 5 0 66 6 1 33 7 1 84 8 1 73 9 0 63  Data Normalization # Importing required from sklearn.preprocessed for the shown we can of # Importing required from sklearn.preprocescaler = StandardSca # Performing Normali df2=df1.copy() df2['Age', 'Blood P df2.head(10)  Gender Age Bi 0 1 0.558212 1 0 0.3089479 2 1 1.059493 3 1 1.367048 4 1 1.48299 5 0 0.327116 6 1 1.579425 7 1 1.367048 9 0 0.15374  Data Visualization # Plotting the bar of plt.birthele'(Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.ylabel('Gender') plt.show()  Comp  350 - 200 - 2	pe, dtype: int64  patment Categorical of the catego	Categorical Commensus of the state of the st	olumn are transported by the seatment of the s	ansformed into numerical status   Chest Pain	Type Treatment 3	Cholesterol(mg/dL)  Blood Pressure (mmHg)',  nent 2 0 1 3 1 2 2 0 0 0 d'])	'Cholesterol (mg/dL)	

Credential's

Name : Mayank Anand