. Perform the following operations using Python on the Air quality and Heart Diseases data sets

a. Data cleaning b. Data integration c. Data transformation d. Error correcting e. Data model building

In [5]:
```
import pandas as pd
import numpy as np
import  seaborn as sns
import matplotlib.pyplot as plt
import random as rd
```

In [6]:
```
ds=pd.read_csv("heart.csv")
ds
```

Out[6]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	ta
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	

303 rows × 14 columns

In [7]:
```
df=pd.read_csv("data.csv",encoding="ISO-8859-1")
df
```

```
C:\Users\Swanand\AppData\Local\Temp\ipykernel_22200\3203580699.py:1: DtypeWarning: Columns (0) have mixed types. Specify dtype option on import or set low_memory=False.
  df=pd.read_csv("data.csv",encoding="ISO-8859-1")
```

Out[7]:

	stn_code	sampling_date	state	location	agency	type	so2	no2
0	150.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4
1	151.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0
2	152.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5
3	150.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7
4	151.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5
...	...	...	...	...	...	...	...	...
435737	SAMP	24-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	22.0	50.0
435738	SAMP	29-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	20.0	46.0
435739	NaN	NaN	andaman-and-nicobar-islands	NaN	NaN	NaN	NaN	NaN
435740	NaN	NaN	Lakshadweep	NaN	NaN	NaN	NaN	NaN
435741	NaN	NaN	Tripura	NaN	NaN	NaN	NaN	NaN

435742 rows × 13 columns

#Data Cleaning

In [8]:
```
ds.head()
```

Out[8]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	targ
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	

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

Out[9]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	rspm	spm
0	150.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4	NaN	NaN
1	151.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0	NaN	NaN
2	152.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5	NaN	NaN
3	150.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7	NaN	NaN
4	151.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5	NaN	NaN

In [10]:
```
ds.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   age       303 non-null    int64  
 1   sex       303 non-null    int64  
 2   cp        303 non-null    int64  
 3   trestbps  303 non-null    int64  
 4   chol      303 non-null    int64  
 5   fbs       303 non-null    int64  
 6   restecg   303 non-null    int64  
 7   thalach   303 non-null    int64  
 8   exang     303 non-null    int64  
 9   oldpeak   303 non-null    float64
 10  slope     303 non-null    int64  
 11  ca        303 non-null    int64  
 12  thal      303 non-null    int64  
 13  target    303 non-null    int64  
dtypes: float64(1), int64(13)
memory usage: 33.3 KB
```

In [11]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 435742 entries, 0 to 435741
Data columns (total 13 columns):
 #   Column                        Non-Null Count   Dtype  
---  ------                        --------------   -----  
 0   stn_code                      291665 non-null  object 
 1   sampling_date                 435739 non-null  object 
 2   state                         435742 non-null  object 
 3   location                      435739 non-null  object 
 4   agency                        286261 non-null  object 
 5   type                          430349 non-null  object 
 6   so2                           401096 non-null  float64
 7   no2                           419509 non-null  float64
 8   rspm                          395520 non-null  float64
 9   spm                           198355 non-null  float64
 10  location_monitoring_station   408251 non-null  object 
 11  pm2_5                         9314 non-null    float64
 12  date                          435735 non-null  object 
dtypes: float64(5), object(8)
memory usage: 43.2+ MB
```

In [12]:
```
ds.isnull().sum()
```

Out[12]:
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

In [13]:
```
df.isnull().sum()
```

Out[13]:
```
stn_code                     144077
sampling_date                     3
state                             0
location                          3
agency                       149481
type                           5393
so2                           34646
no2                           16233
rspm                          40222
spm                          237387
location_monitoring_station   27491
pm2_5                        426428
date                              7
dtype: int64
```

In [14]:
```
ds.dropna()
```

Out[14]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	ta
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	

303 rows × 14 columns

In [15]:
```
df.dropna()
```

Out[15]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	rspm	spm	location_mo

In [16]:
```
#Data integration
```

In [17]:
```
ds1=ds.iloc[[1,3,5,4,22,43,54,67,7,8,9,50,10,11]]
ds2=df.loc[111:999,['state', 'location', 'so2', 'rspm']]
```

In [18]:
```
ds1
```

Out[18]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	tar
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	
5	57	1	0	140	192	0	1	148	0	0.4	1	0	1	
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	
22	42	1	0	140	226	0	1	178	0	0.0	2	0	2	
43	53	0	0	130	264	0	0	143	0	0.4	1	0	2	
54	63	0	2	135	252	0	0	172	0	0.0	2	0	2	
67	45	0	1	130	234	0	0	175	0	0.6	1	0	2	
7	44	1	1	120	263	0	1	173	0	0.0	2	0	3	
8	52	1	2	172	199	1	1	162	0	0.5	2	0	3	
9	57	1	2	150	168	0	1	174	0	1.6	2	0	2	
50	51	0	2	130	256	0	0	149	0	0.5	2	0	2	
10	54	1	0	140	239	0	1	160	0	1.2	2	0	2	
11	48	0	2	130	275	0	1	139	0	0.2	2	0	2	

In [19]:
```
ds2
```

Out[19]:

	state	location	so2	rspm
111	Andhra Pradesh	Hyderabad	4.9	NaN
112	Andhra Pradesh	Vishakhapatnam	NaN	NaN
113	Andhra Pradesh	Vishakhapatnam	11.2	NaN
114	Andhra Pradesh	Vishakhapatnam	4.5	NaN
115	Andhra Pradesh	Hyderabad	6.2	NaN
...	...	...	...	...
995	Andhra Pradesh	Hyderabad	2.8	NaN
996	Andhra Pradesh	Hyderabad	5.0	NaN
997	Andhra Pradesh	Hyderabad	5.5	NaN
998	Andhra Pradesh	Hyderabad	5.8	NaN
999	Andhra Pradesh	Hyderabad	5.9	NaN

889 rows × 4 columns

In [20]:
```
ds_integration=pd.concat([ds2,ds1])
ds_integration
```

Out[20]:

	state	location	so2	rspm	age	sex	cp	trestbps	chol	fbs	restecg	t
111	Andhra Pradesh	Hyderabad	4.9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
112	Andhra Pradesh	Vishakhapatnam	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
113	Andhra Pradesh	Vishakhapatnam	11.2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
114	Andhra Pradesh	Vishakhapatnam	4.5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
115	Andhra Pradesh	Hyderabad	6.2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	...	...	...	...	...	...	...	...	...	...	...	
8	NaN	NaN	NaN	NaN	52.0	1.0	2.0	172.0	199.0	1.0	1.0	
9	NaN	NaN	NaN	NaN	57.0	1.0	2.0	150.0	168.0	0.0	1.0	
50	NaN	NaN	NaN	NaN	51.0	0.0	2.0	130.0	256.0	0.0	0.0	
10	NaN	NaN	NaN	NaN	54.0	1.0	0.0	140.0	239.0	0.0	1.0	
11	NaN	NaN	NaN	NaN	48.0	0.0	2.0	130.0	275.0	0.0	1.0	

903 rows × 18 columns

#Data Transformation

In [21]:
```
ds_integration.transpose()
df.drop(columns = "so2")
```

Out[21]:

	stn_code	sampling_date	state	location	agency	type	no2	rspm
0	150.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	17.4	NaN
1	151.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	7.0	NaN
2	152.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	28.5	NaN
3	150.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	14.7	NaN
4	151.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	7.5	NaN
...	...	...	...	...	...	...	...	...
435737	SAMP	24-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	50.0	143.0
435738	SAMP	29-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	46.0	171.0
435739	NaN	NaN	andaman-and-nicobar-islands	NaN	NaN	NaN	NaN	NaN
435740	NaN	NaN	Lakshadweep	NaN	NaN	NaN	NaN	NaN
435741	NaN	NaN	Tripura	NaN	NaN	NaN	NaN	NaN

435742 rows × 12 columns

Error correction

In [22]:
```
df.melt()
```

Out[22]:

	variable	value
0	stn_code	150.0
1	stn_code	151.0
2	stn_code	152.0
3	stn_code	150.0
4	stn_code	151.0
...	...	...
5664641	date	2015-12-24
5664642	date	2015-12-29
5664643	date	NaN
5664644	date	NaN
5664645	date	NaN

5664646 rows × 2 columns

In [23]:
```
df_merged=pd.concat([df,ds])
df_merged
```

Out[23]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	rspm	sp
0	150.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4	NaN	N
1	151.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0	NaN	N
2	152.0	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5	NaN	N
3	150.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7	NaN	N
4	151.0	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5	NaN	N
...	...	...	...	...	...	...	...	...	...	
298	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
299	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
300	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
301	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
302	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N

436045 rows × 27 columns

In [24]:
```
df['stn_code'].unique()
```

Out[24]:
```
array([150.0, 151.0, 152.0, 95.0, 202.0, 203.0, 232.0, 233.0, 234.0,
       241.0, 365.0, 393.0, 394.0, 371.0, 387.0, 388.0, nan, 582.0, 583.0,
       581.0, 466.0, 577.0, 580.0, 468.0, 465.0, 389.0, 462.0, 469.0,
       467.0, 585.0, 579.0, 470.0, 578.0, 748.0, 742.0, 750.0, 745.0,
       749.0, 755.0, 743.0, 740.0, 746.0, 753.0, 757.0, 758.0, 756.0,
       751.0, 739.0, 584.0, 752.0, 741.0, 787.0, 788.0, 193.0, 194.0,
       195.0, 218.0, 217.0, 242.0, 520.0, 542.0, 566.0, 538.0, 539.0,
       603.0, 602.0, 519.0, 596.0, 541.0, 587.0, 595.0, 597.0, 604.0,
       607.0, 567.0, 536.0, 594.0, 586.0, 605.0, 537.0, 46.0, 45.0, 44.0,
       104.0, 171.0, 172.0, 173.0, 174.0, 210.0, 211.0, 284.0, 106.0,
       107.0, 108.0, 263.0, 264.0, 463.0, 464.0, 249.0, 364.0, 407.0,
       65.0, 67.0, 245.0, 368.0, 223.0, 447.0, 669.0, 235.0, 558.0, 115.0,
       116.0, 114.0, 560.0, '560', '60', '59', '58', '57', '56', '55',
       '144', '145', '146', '531', '345', '36', '37', '246', '327', '435',
       '633', '634', '632', '630', '628', '631', '629', '780', '775',
       '774', '776', '777', '778', '781', '49', '47', '50', '48', '51',
       '23', '22', '21', '100', '101', '102', '103', '97', '99', '153',
       '154', '155', '189', '190', '220', '221', '252', '253', '256',
       '257', '247', '248', '319', '367', '347', '374', '333', '334',
       '813', '814', '815', 'SAMP', '42', '43', '196', '275', '330',
       '331', '390', '414', '34', '35', '117', '118', '132', '119', '268',
       '271', '339', '449', '564', '563', '461', '530', '565', '670',
       '671', '665', '667', '668', '666', '794', '795', '184', '482',
       '507', '44', '46', '332', '402', '351', '382', 615.0, 351.0, 382.0,
       332.0, 612.0, 611.0, 402.0, 614.0, 41.0, 39.0, 40.0, 77.0, 78.0,
       79.0, 80.0, 81.0, 404.0, 405.0, 406.0, 457.0, 598.0, 460.0, 674.0,
       673.0, 432.0, 459.0, 458.0, 431.0, 488.0, 328.0, 684.0, 672.0,
       679.0, 675.0, 683.0, 676.0, 677.0, 680.0, 685.0, 678.0, 29.0, 30.0,
       31.0, 32.0, 33.0, 105.0, 147.0, 148.0, 149.0, 187.0, 188.0, 185.0,
       186.0, 181.0, 179.0, 180.0, 182.0, 311.0, 618.0, 617.0, 338.0,
       562.0, 346.0, 621.0, 620.0, 361.0, 360.0, 359.0, 623.0, 419.0,
       357.0, 358.0, 619.0, 624.0, 622.0, 546.0, 82.0, 83.0, 84.0, 66.0,
       91.0, 130.0, 129.0, 131.0, 126.0, 128.0, 127.0, 125.0, 123.0,
       124.0, 122.0, 198.0, 199.0, 200.0, 201.0, 222.0, 248.0, 247.0,
       525.0, 524.0, 523.0, 479.0, 478.0, 246.0, 532.0, 343.0, 342.0,
       515.0, 514.0, 516.0, 527.0, 526.0, 528.0, 529.0, 796.0, 797.0,
       798.0, 799.0, 800.0, 801.0, 802.0, 803.0, 804.0, 53.0, 52.0, 54.0,
       170.0, 167.0, 166.0, 165.0, 133.0, 137.0, 136.0, 135.0, 134.0,
       94.0, 138.0, 169.0, 168.0, 175.0, 213.0, 214.0, 230.0, 254.0,
       255.0, 267.0, 269.0, 270.0, 259.0, 282.0, 266.0, 281.0, 283.0,
       265.0, 287.0, 288.0, 289.0, 290.0, 280.0, 299.0, 300.0, 304.0,
       305.0, 303.0, 312.0, 313.0, 314.0, 381.0, 379.0, 349.0, 547.0,
       548.0, 549.0, 511.0, 512.0, 513.0, 649.0, 396.0, 640.0, 638.0,
       639.0, 644.0, 645.0, 508.0, 509.0, 510.0, 641.0, 642.0, 643.0,
       489.0, 490.0, 569.0, 570.0, 571.0, 491.0, 492.0, 493.0, 494.0,
       495.0, 496.0, 572.0, 573.0, 576.0, 647.0, 648.0, 700.0, 701.0,
       646.0, 707.0, 706.0, 703.0, 705.0, 704.0, 710.0, 708.0, 574.0,
       575.0, 702.0, 445.0, 711.0, 120.0, 121.0, 340.0, 568.0, 588.0,
       608.0, 699.0, 698.0, 782.0, 450.0, 451.0, 452.0, 738.0, 737.0,
       731.0, 732.0, 734.0, 733.0, 735.0, 736.0, 317.0, 318.0, 448.0,
       610.0, 609.0, 70.0, 68.0, 69.0, 226.0, 227.0, 231.0, 229.0, 228.0,
       322.0, 370.0, 428.0, 429.0, 424.0, 423.0, 426.0, 425.0, 427.0,
       471.0, 688.0, 687.0, 686.0, 689.0, 690.0, 697.0, 696.0, 695.0,
       694.0, 691.0, 692.0, 693.0, 552.0, 816.0, 817.0, 818.0, 819.0,
       64.0, 93.0, 92.0, 337.0, 784.0, 785.0, 786.0, 63.0, 62.0, 61.0,
       76.0, 176.0, 177.0, 178.0, 244.0, 301.0, 302.0, 335.0, 487.0,
       486.0, 506.0, 504.0, 505.0, 483.0, 353.0, 485.0, 484.0, 422.0,
       420.0, 421.0, 590.0, 600.0, 599.0, 392.0, 355.0, 789.0, 790.0,
       791.0, 792.0, 16.0, 17.0, 20.0, 18.0, 19.0, 158.0, 157.0, 156.0,
       219.0, 243.0, 250.0, 273.0, 279.0, 291.0, 296.0, 297.0, 298.0,
       272.0, 274.0, 293.0, 294.0, 295.0, 325.0, 326.0, 372.0, 373.0,
```

```
376.0, 408.0, 410.0, 409.0, 413.0, 411.0, 412.0, 320.0, 321.0,
38.0, 71.0, 72.0, 73.0, 74.0, 75.0, 161.0, 160.0, 159.0, 205.0,
206.0, 207.0, 10001.0, 10002.0, 10003.0, 20001.0, 20005.0, 20006.0,
30001.0, 30003.0, 30004.0, 239.0, 240.0, 237.0, 238.0, 260.0,
261.0, 306.0, 307.0, 308.0, 309.0, 375.0, 366.0, 769.0, 770.0,
771.0, 772.0, 773.0, 764.0, 765.0, 766.0, 767.0, 762.0, 763.0,
760.0, 761.0, 759.0, 768.0, 744.0, 754.0, 1.0, 7.0, 2.0, 3.0, 4.0,
6.0, 5.0, 90.0, 109.0, 110.0, 89.0, 113.0, 112.0, 85.0, 141.0,
142.0, 143.0, 111.0, 140.0, 139.0, 98.0, 87.0, 86.0, 192.0, 191.0,
209.0, 208.0, 212.0, 215.0, 224.0, 225.0, 258.0, 216.0, 278.0,
276.0, 277.0, 391.0, 395.0, 362.0, 323.0, 324.0, 369.0, 378.0,
399.0, 400.0, 401.0, 403.0, 377.0, 397.0, 398.0, 554.0, 555.0,
517.0, 518.0, 534.0, 535.0, 550.0, 551.0, 553.0, 415.0, 416.0,
417.0, 714.0, 715.0, 729.0, 728.0, 730.0, 721.0, 722.0, 723.0,
439.0, '534', '535', '109', '113', '377', '397', '398', '724',
'725', '550', '551', '712', '713', '403', '378', '718', '719',
'720', '716', '717', '727', '726', '362', '553', '555', '554', '7',
'6', '714', '715', '139', '140', '258', '369', '730', '729', '728',
'212', '98', '86', '391', '395', '723', '722', '721', '1', '417',
'416', '415', '323', '324', '399', '400', '401', '141', '517',
'518', '805', '806', '807', '89', '90', '637', '625', '635', '627',
'636', '626', '13', '14', '15', '8', '9', '10', '11', '164', '163',
'162', '383', '384', '385', '386', '593', '592', '655', '653',
'654', '591', '663', '664', '474', '473', '475', '472', '476',
'477', '662', '660', '661', '657', '659', '656', '658', '652',
'650', '651', '348'], dtype=object)
```

In [25]:
```
df.stn_code.value_counts()
```

Out[25]:
```
193.0    1428
519.0    1280
708.0    1273
541.0    1270
710.0    1269
         ...
560.0       2
224.0       1
209.0       1
207.0       1
275         1
Name: stn_code, Length: 803, dtype: int64
```

In [27]:
```
from sklearn import linear_model ,metrics

X=ds[["age"]]
Y=ds[["thal"]]
```

In [28]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=0.2,random_s
```

In [29]:
```
len(X_train)
```

Out[29]:
```
242
```

In [30]:
```
len(X_test)
```

Out[30]:
```
61
```

In [31]:
```
ds.shape
```

Out[31]:
```
(303, 14)
```

In [32]:
```
reg=linear_model.LinearRegression()
```

In [33]:
```
print(X_train)
```

```
     age
62    52
127   67
111   57
287   57
108   50
..   ...
203   68
255   45
72    29
235   51
37    54

[242 rows x 1 columns]
```

In [34]:
```
model=reg.fit(X_train,Y_train)
r_sq=reg.score(X_train,Y_train)
```

In [35]:
```
print("determination coefficient :",r_sq)
```

```
determination coefficient : 0.005160903314535115
```

In [36]:
```
print("intercept:",model.intercept_)
```

```
intercept: [2.0284109]
```

In [37]:
```
print("slope:",model.coef_)
```

```
slope: [[0.00478833]]
```

In [38]:
```
Y_pred=model.predict(X_test)
```

In [39]:
```
print('predicted response :',Y_pred,sep='\n')
```

```
predicted response :
[[2.3252874 ]
 [2.29655741]
 [2.25825077]
 [2.31571074]
 [2.26782743]
 [2.33007573]
 [2.29655741]
 [2.24867411]
 [2.31571074]
 [2.21515579]
 [2.34444072]
 [2.35880571]
 [2.30613407]
 [2.33486406]
 [2.2630391 ]
 [2.30134574]
 [2.19121414]
 [2.33007573]
 [2.23430912]
 [2.31092241]
 [2.25825077]
 [2.35401738]
 [2.28698075]
 [2.23909745]
 [2.22473245]
 [2.21036746]
 [2.30613407]
 [2.22473245]
 [2.27261576]
 [2.28698075]
 [2.19600247]
 [2.2630391 ]
 [2.28219242]
 [2.30134574]
 [2.26782743]
 [2.27261576]
 [2.34444072]
 [2.34922905]
 [2.22473245]
 [2.28219242]
 [2.33007573]
 [2.30134574]
 [2.30134574]
 [2.29655741]
 [2.33007573]
 [2.32049907]
 [2.29176908]
 [2.30134574]
 [2.21994412]
 [2.33486406]
 [2.27740409]
 [2.26782743]
 [2.24388578]
 [2.30613407]
 [2.30613407]
 [2.28698075]
 [2.33007573]
 [2.30134574]
 [2.21036746]
```

```
[2.30134574]
[2.33486406]]
```

In [ ]: