

In [68]:

```
print("Name: Lakshya Singh Chauhan")
print("We will be cleaning heart disease data, and conclude which age group has high risk of heart stroke as per diabetes and high blood pressure level")
print("We will also find which gender has the most not normal platelets count in blood, and plot a pie chart around it")
```

Name: Lakshya Singh Chauhan

We will be cleaning heart disease data, and conclude which age group has high risk of heart stroke as per diabetes and high blood pressure level

We will also find which gender has the most not normal platelets count in blood, and plot a pie chart around it

Task 1 - Find the diabetic and high blood pressure patients across all age groups, and conclude the risk heart stroke is higher in which age group

In [69]:

```
#Import Libraries
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

#read the csv
df = pd.read_csv('heart_disease.csv')
df
```

Out[69]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure
0	75.0	0	582	0	20	
1	55.0	0	7861	0	38	
2	65.0	0	146	0	20	
3	50.0	1	111	0	20	
4	65.0	1	160	1	20	
...
294	62.0	0	61	1	38	
295	55.0	0	1820	0	38	
296	45.0	0	2060	1	60	
297	45.0	0	2413	0	38	
298	50.0	0	196	0	45	

299 rows × 13 columns

In [70]:

```
#Filter by diabetes(condition will be who has diabetes) and create new dataframe
filter_by_diabetes = df.loc[df["diabetes"] == 1]
filter_by_diabetes
```

Out[70]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressu
4	65.0	1	160	1	20	
7	60.0	1	315	1	60	
19	48.0	1	582	1	55	
21	65.0	1	128	1	30	
23	53.0	0	63	1	60	
...	
290	45.0	0	615	1	55	
292	52.0	0	190	1	38	
293	63.0	1	103	1	35	
294	62.0	0	61	1	38	
296	45.0	0	2060	1	60	

125 rows × 13 columns



In [71]:

```
#On this new data frame perform group operation as per age and create new dataframe  
diabetes_age_group = filter_by_diabetes.groupby("age")["diabetes"].count().reset_index()  
diabetes_age_group
```

Out[71]:

	age	diabetes
0	40.000	2
1	42.000	2
2	44.000	2
3	45.000	11
4	46.000	1
5	48.000	2
6	49.000	1
7	50.000	12
8	51.000	2
9	52.000	2
10	53.000	3
11	54.000	1
12	55.000	4
13	56.000	1
14	58.000	5
15	59.000	3
16	60.000	19
17	60.667	2
18	61.000	2
19	62.000	3
20	63.000	4
21	65.000	15
22	66.000	1
23	68.000	2
24	69.000	1
25	70.000	9
26	72.000	2
27	73.000	1
28	75.000	3
29	80.000	3
30	82.000	2
31	90.000	1
32	94.000	1

In [72]:

```
#Filter by high_blood_pressure(condition will be who has high_blood_pressure) and create  
high_blood_pressure_true = df.loc[df["high_blood_pressure"] == 1]  
high_blood_pressure_true
```

Out[72]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressu
0	75.0	0	582	0	20	
5	90.0	1	47	0	40	
9	80.0	1	123	0	35	
10	75.0	1	81	0	38	
11	62.0	0	231	0	25	
...	
255	52.0	1	191	1	30	
257	58.0	0	132	1	38	
270	44.0	0	582	1	30	
275	45.0	0	582	0	38	
294	62.0	0	61	1	38	

105 rows × 13 columns



In [73]:

```
#On this new data frame perform group operation as per age and create new dataframe  
groupby_high_blood_pressure = high_blood_pressure_true.groupby("age")["high_blood_pressu  
groupby_high_blood_pressure
```

Out[73]:

	age	high_blood_pressure
0	40.000	1
1	44.000	2
2	45.000	5
3	46.000	2
4	48.000	1
5	49.000	3
6	50.000	10
7	51.000	1
8	52.000	1
9	53.000	3
10	54.000	1
11	55.000	4
12	57.000	1
13	58.000	3
14	59.000	2
15	60.000	9
16	60.667	1
17	61.000	2
18	62.000	3
19	63.000	2
20	65.000	8
21	66.000	2
22	68.000	2
23	70.000	13
24	72.000	5
25	73.000	2
26	75.000	7
27	77.000	1
28	79.000	1
29	80.000	3
30	82.000	1
31	90.000	1
32	94.000	1
33	95.000	1

In [74]:

```
#plot the scatter graph to show which age group is more prone to diabetes
plt.figure(figsize=(12, 8))
plt.xticks(rotation='vertical')

rgb = np.random.rand(3)
print(rgb)

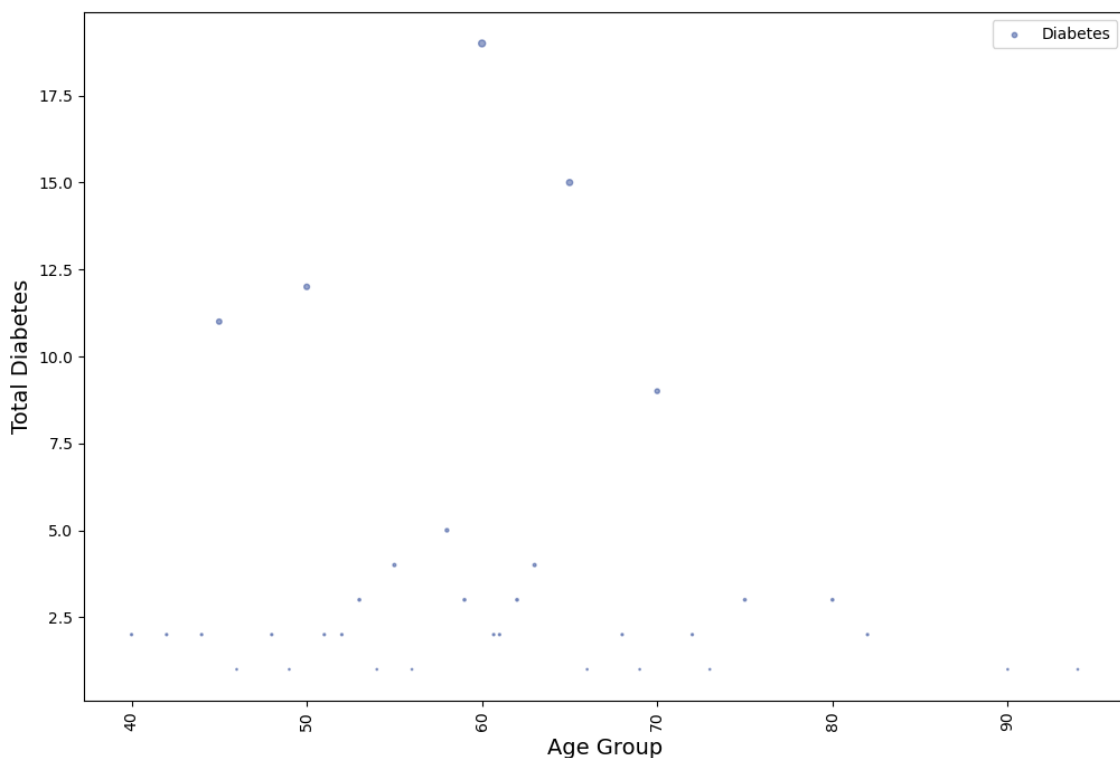
plt.scatter(diabetes_age_group['age'], diabetes_age_group['diabetes'],
            c=[rgb], label = 'Diabetes',
            alpha=0.5,
            s = diabetes_age_group['diabetes'])

plt.legend()
plt.xlabel("Age Group", size=14)
plt.ylabel("Total Diabetes", size=14)
```

```
[0.1906108  0.30054176 0.60894825]
```

Out[74]:

```
Text(0, 0.5, 'Total Diabetes')
```



Conclusion -

Task 2 - Find as per gender who has not normal platelets level in blood

In [75]:

```
#Filter by platelets(condition lesser then 150000 OR greater then 450000) and create new  
filter_by_platelets = df.loc[(df["platelets" ] < 150000) | (df["platelets"] > 450000)]  
filter_by_platelets
```

Out[75]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressu
6	75.0	1	246	0	15	
7	60.0	1	315	1	60	
12	45.0	1	981	0	30	
15	82.0	1	379	0	50	
19	48.0	1	582	1	55	
25	80.0	0	148	1	38	
47	60.0	0	582	1	38	
48	80.0	1	553	0	20	
55	95.0	1	371	0	30	
65	60.0	0	68	0	20	
69	65.0	0	113	1	25	
71	58.0	0	582	1	35	
73	65.0	0	224	1	50	
102	80.0	0	898	0	25	
105	72.0	1	328	0	30	
109	45.0	0	292	1	35	
117	85.0	1	102	0	60	
123	60.0	1	582	0	30	
163	50.0	1	2334	1	35	
167	59.0	0	66	1	20	
176	69.0	0	1419	0	40	
177	49.0	1	69	0	50	
194	45.0	0	582	0	20	
200	63.0	1	1767	0	45	
211	50.0	0	582	0	62	
212	78.0	0	224	0	50	
223	47.0	0	582	0	25	
224	58.0	0	582	1	25	
227	55.0	1	2794	0	35	
230	60.0	0	166	0	30	
240	70.0	0	81	1	35	
264	61.0	0	582	1	38	
267	56.0	1	135	1	38	
277	70.0	0	582	1	38	
279	55.0	0	84	1	38	
281	70.0	0	582	0	40	
287	45.0	0	582	1	55	

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressu
291	60.0	0	320	0	35	
296	45.0	0	2060	1	60	
297	45.0	0	2413	0	38	

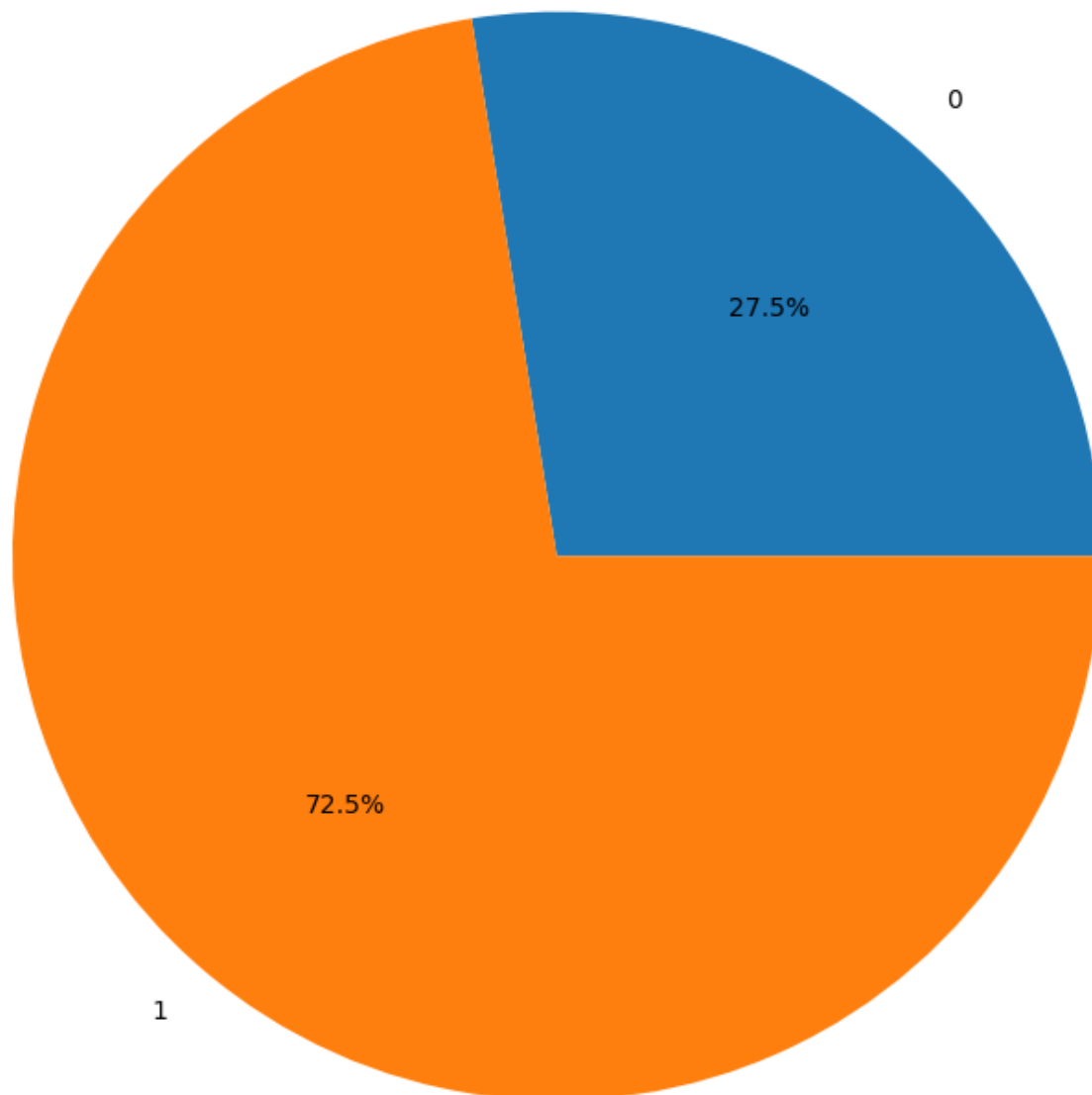
```
#On this new dataframe perform group operation as per gender and create new dataframe  
groupby_gender = filter_by_platelets.groupby("gender")["platelets"].count().reset_index(  
groupby_gender
```

Out[76]:

	gender	platelets
0	0	11
1	1	29

In [77]:

```
#Plot a pie chart as per the gender to show the percentage of male and female who has no  
value = groupby_gender["platelets"]  
label = groupby_gender["gender"]  
  
plt.pie(value, labels=label, autopct="%0.1f%%", radius=2)  
plt.show()
```



Conclusion -

In [78]:

I understood