In [15]:

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


import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
%matplotlib inline

df = pd.read_csv(r"C:\Users\ladsh\OneDrive\Desktop\DATASET/dataset_Facebook.csv", sep=";
df
```

Out[15]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users
0	139441	Photo	2	12	4	3	0.0	2752	5091	178
1	139441	Status	2	12	3	10	0.0	10460	19057	1457
2	139441	Photo	3	12	3	3	0.0	2413	4373	177
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211
4	139441	Photo	2	12	2	3	0.0	7244	13594	671
...	...	...	...	...	...	...	...	...	...	...
495	85093	Photo	3	1	7	2	0.0	4684	7536	733
496	81370	Photo	2	1	5	8	0.0	3480	6229	537
497	81370	Photo	1	1	5	2	0.0	3778	7216	625
498	81370	Photo	3	1	4	11	0.0	4156	7564	626
499	81370	Photo	2	1	4	4	NaN	4188	7292	564

500 rows × 19 columns

In [16]:

```
df.head(5)
```

Out[16]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users
0	139441	Photo	2	12	4	3	0.0	2752	5091	178
1	139441	Status	2	12	3	10	0.0	10460	19057	1457
2	139441	Photo	3	12	3	3	0.0	2413	4373	177
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211
4	139441	Photo	2	12	2	3	0.0	7244	13594	671

In [17]:

```
post_types = df.Type.unique()
post_types
```

Out[17]:

```
array(['Photo', 'Status', 'Link', 'Video'], dtype=object)
```

In [18]:

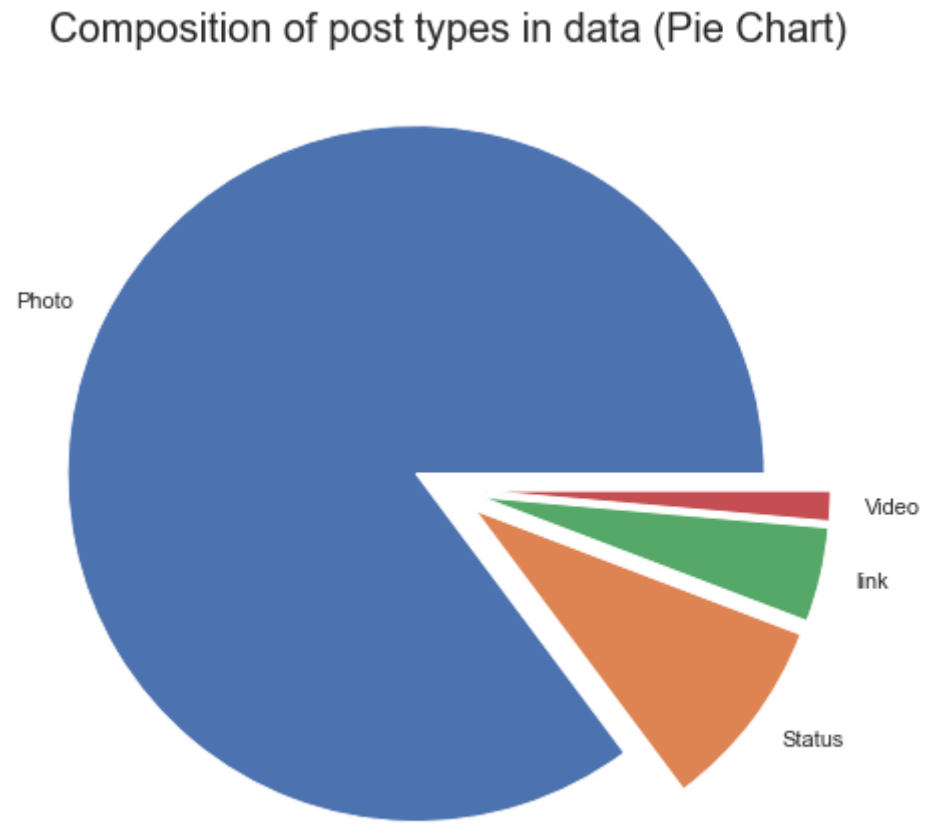
```
frequency_data = {}
for post in post_types:
    subset = df[df.Type == post]
    frequency_data[post] = subset.shape[0]

frequency_data
```

Out[18]:

```
{'Photo': 426, 'Status': 45, 'Link': 22, 'Video': 7}
```

In [19]:

```
fig = plt.figure(figsize=(8, 8))

# Adds subplot on position 1
ax = fig.add_subplot(111)

# Generating legend for pie chart
legend = [
    "Photo",
    "Status",
    "link",
    "Video"
]

# Defining explode values
explode = [0.1, 0.1, 0.1, 0.1]

# Generating and displaying piechart
plt.pie(
    x=frequency_data.values(),
    labels=legend,
    explode=explode,
)
plt.title("Composition of post types in data (Pie Chart)", fontsize=20)
plt.show()
```



In [20]:

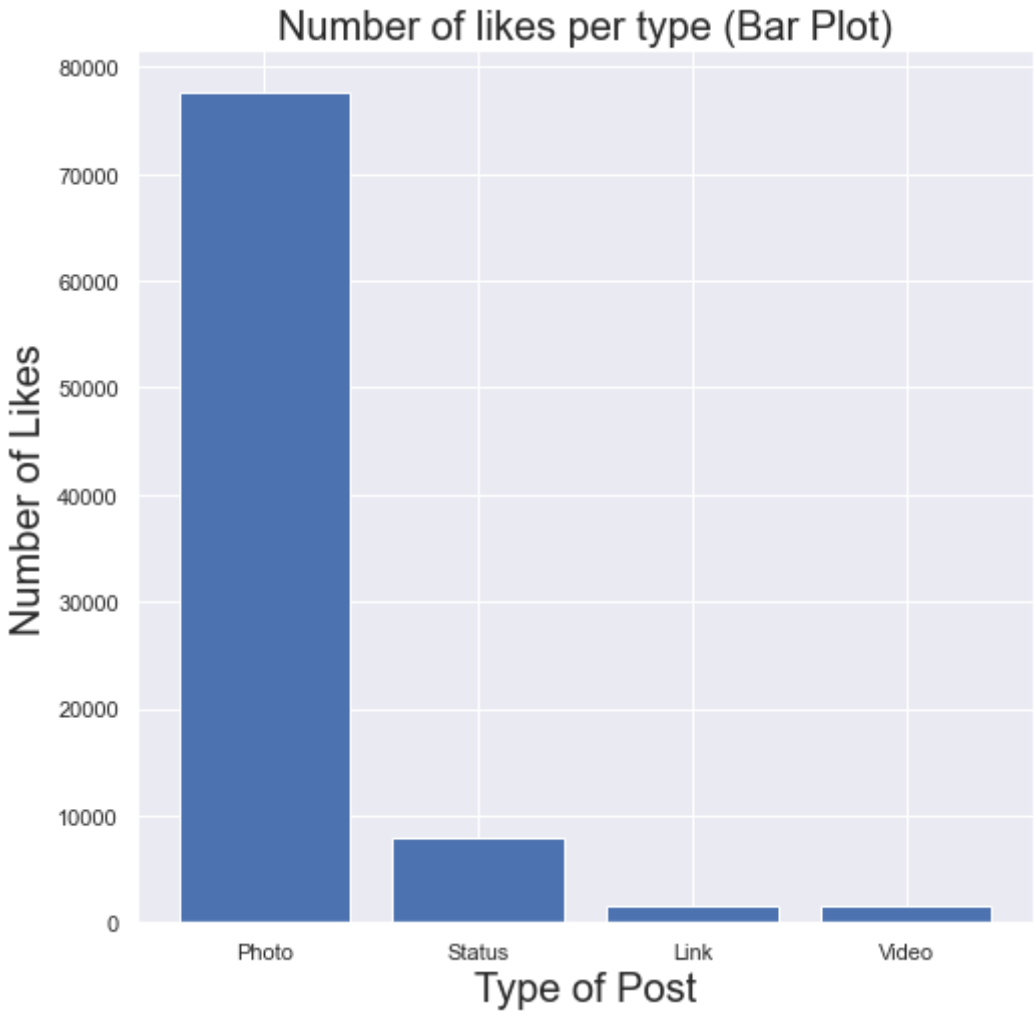
```
likes_per_type = {}

for post in post_types:
    subset = df[df.Type == post]
    likes_per_type[post] = subset.like.sum()

likes_per_type
```

Out[20]:

```
{'Photo': 77610.0, 'Status': 7952.0, 'Link': 1613.0, 'Video': 1620.0}
```

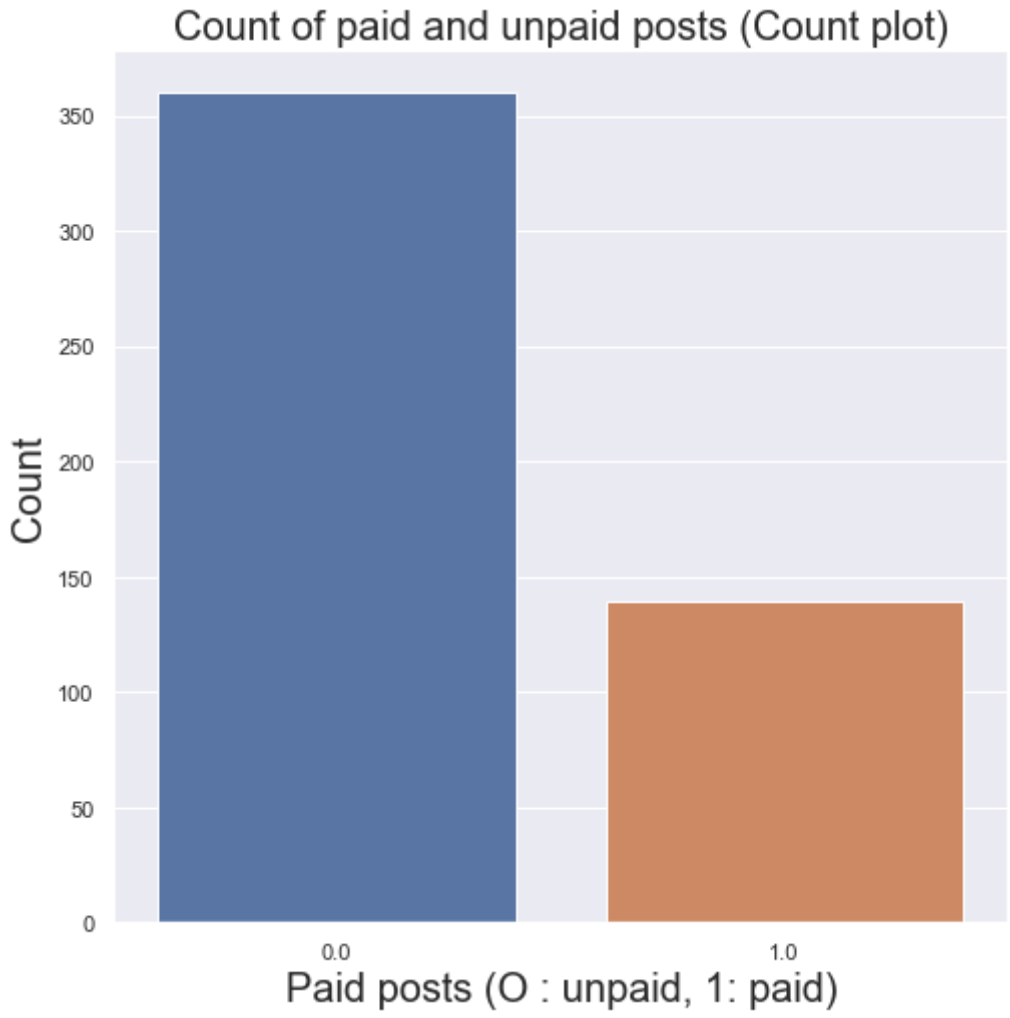
In [21]:

```
fig = plt.figure(figsize=(8, 8))

# Adds subplot on position 1
ax = fig.add_subplot(111)

# Generating and displaying bar chart
plt.bar(
    x=likes_per_type.keys(),
    height=likes_per_type.values()
)
plt.xlabel("Type of Post", fontsize=20)
plt.ylabel("Number of Likes", fontsize=20)
plt.title("Number of likes per type (Bar Plot)", fontsize=20)
plt.show()
```



In [22]:

```
# Generating bar graph
fig = plt.figure(figsize=(8, 8))

# Adds subplot on position 1
ax = fig.add_subplot(111)

sns.countplot(x=df.Paid)

plt.xlabel("Paid posts (O : unpaid, 1: paid)", fontsize=20)
plt.ylabel("Count", fontsize=20)
plt.title("Count of paid and unpaid posts (Count plot)", fontsize=20)

plt.show()
```

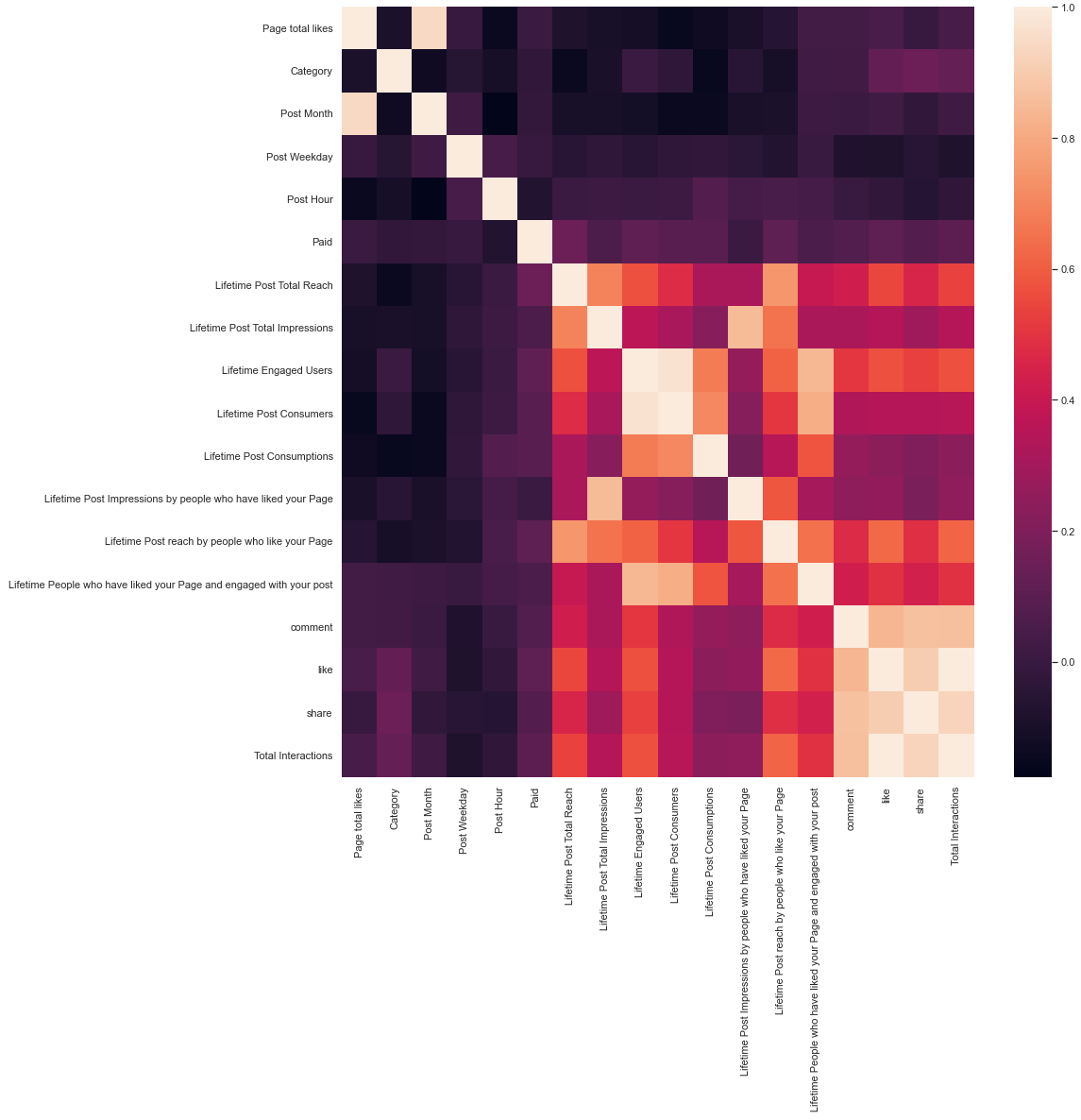


In [23]:

```
# Generating bar graph
fig = plt.figure(figsize=(15, 15))

# Adds subplot on position 1
ax = fig.add_subplot(111)

sns.heatmap(df.corr())

plt.plot()
```

Out[23]:

```
[]
```

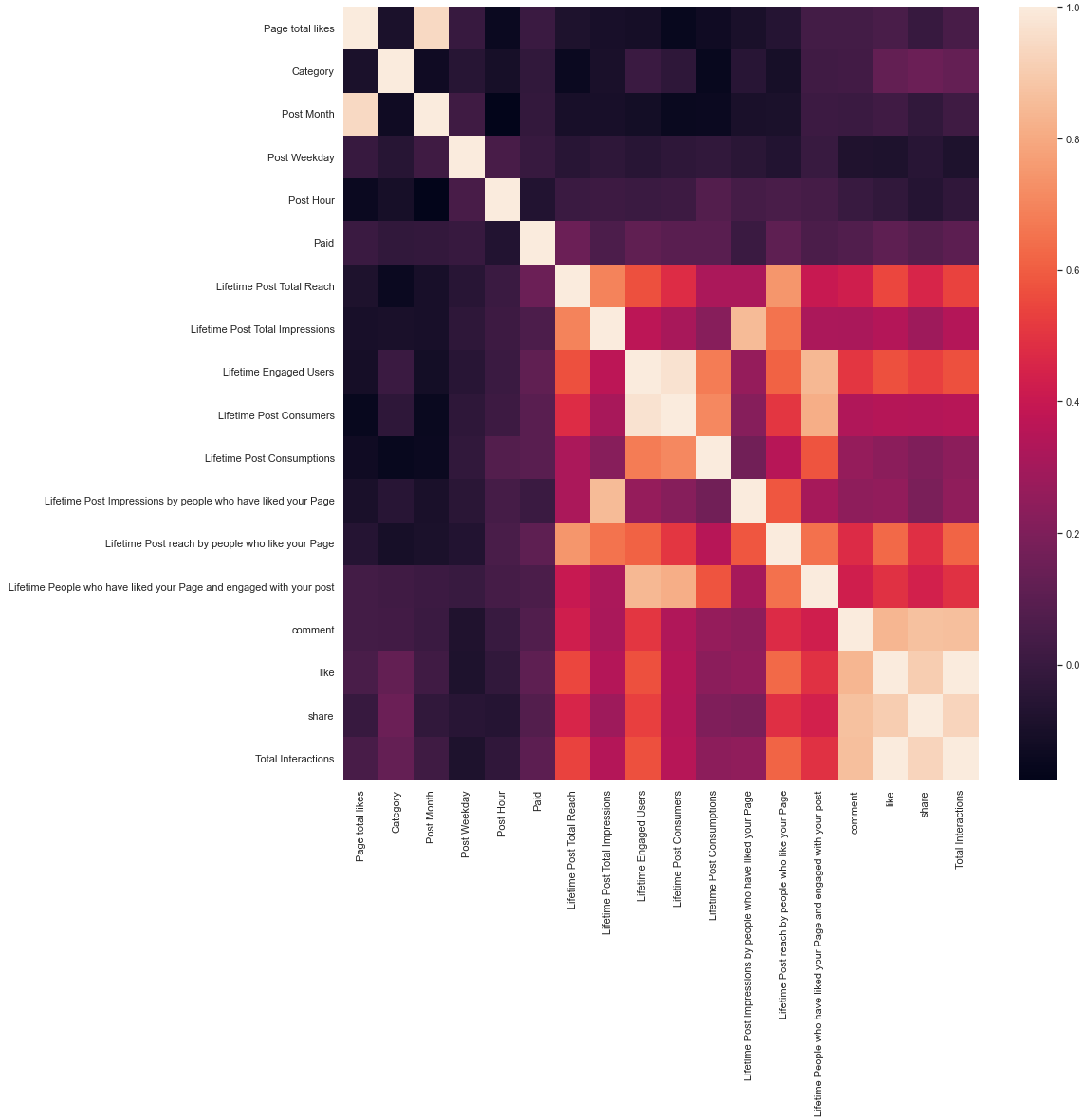


In [55]:

```
fig=plt.figure(figsize=(15,15))
sns.heatmap(df.corr())
```

Out[55]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c3568340>
```



In [24]:

```
df1 = pd.read_csv("C:\SEM 5\Dataset\Heart.csv")
df1
```

Out[24]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Old
0	1	63	1	typical	145	233	1	2	150	0	
1	2	67	1	asymptomatic	160	286	0	2	108	1	
2	3	67	1	asymptomatic	120	229	0	2	129	1	
3	4	37	1	nonanginal	130	250	0	0	187	0	
4	5	41	0	nontypical	130	204	0	2	172	0	
...	...	...	...	...	...	...	...	...	...	...	
298	299	45	1	typical	110	264	0	0	132	0	
299	300	68	1	asymptomatic	144	193	1	0	141	0	
300	301	57	1	asymptomatic	130	131	0	0	115	1	
301	302	57	0	nontypical	130	236	0	2	174	0	
302	303	38	1	nonanginal	138	175	0	0	173	0	

303 rows × 15 columns

In [25]:

```
df1.head()
```

Out[25]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpe
0	1	63	1	typical	145	233	1	2	150	0	2
1	2	67	1	asymptomatic	160	286	0	2	108	1	1
2	3	67	1	asymptomatic	120	229	0	2	129	1	2
3	4	37	1	nonanginal	130	250	0	0	187	0	3
4	5	41	0	nontypical	130	204	0	2	172	0	1

In [26]:

```
x=df1.Age
x
```

Out[26]:

```
0      63
1      67
2      67
3      37
4      41
       ..
298    45
299    68
300    57
301    57
302    38
Name: Age, Length: 303, dtype: int64
```

In [27]:

```
y=df1.Chol
y
```

Out[27]:

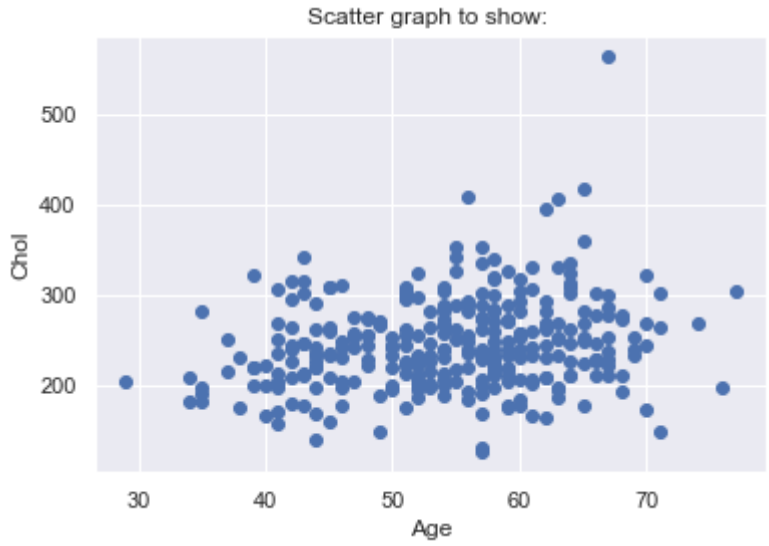
```
0      233
1      286
2      229
3      250
4      204
      ...
298    264
299    193
300    131
301    236
302    175
Name: Chol, Length: 303, dtype: int64
```

In [28]:

```
plt.title("Scatter graph to show: ")
plt.xlabel("Age")
plt.ylabel("Chol")
plt.scatter(x,y)
```

Out[28]:

```
<matplotlib.collections.PathCollection at 0x216c39122b0>
```



In [29]:

```
x=df1.Sex
x
```

Out[29]:

```
0      1
1      1
2      1
3      1
4      0
      ..
298    1
299    1
300    1
301    0
302    1
Name: Sex, Length: 303, dtype: int64
```

In [30]:

```
y=df1.Ca
y
```

Out[30]:

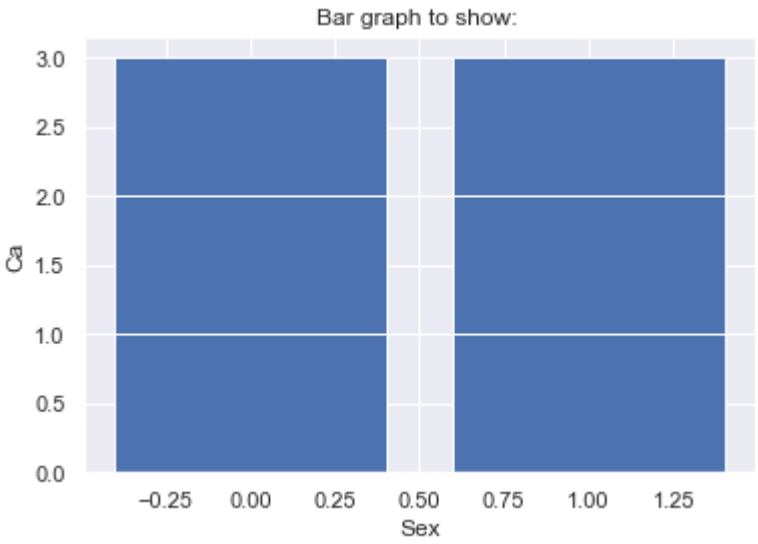
```
0      0.0
1      3.0
2      2.0
3      0.0
4      0.0
      ...
298    0.0
299    2.0
300    1.0
301    1.0
302    NaN
Name: Ca, Length: 303, dtype: float64
```

In [31]:

```
plt.title("Bar graph to show: ")
plt.xlabel("Sex")
plt.ylabel("Ca")
plt.bar(x,y)
```

Out[31]:

```
<BarContainer object of 303 artists>
```



In [32]:

```
x
```

Out[32]:

```
0      1
1      1
2      1
3      1
4      0
      ..
298    1
299    1
300    1
301    0
302    1
Name: Sex, Length: 303, dtype: int64
```

In [33]:

```
y=df1.Slope
y
```

Out[33]:

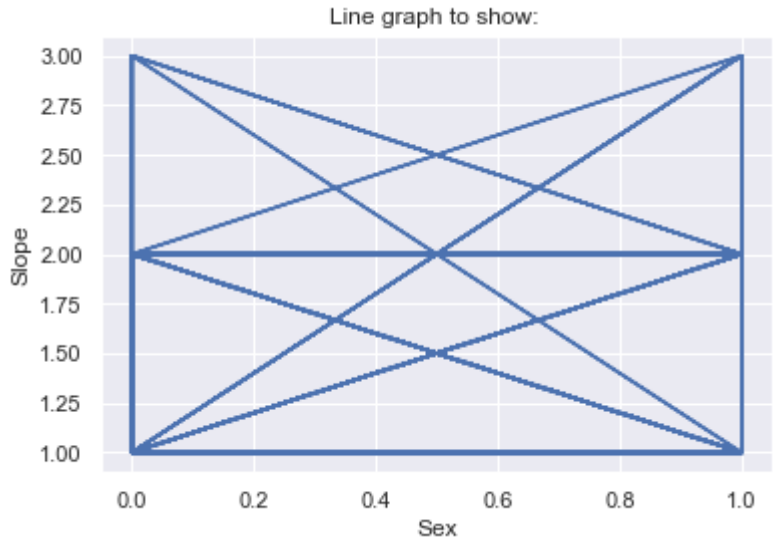
```
0      3
1      2
2      2
3      3
4      1
      ..
298    2
299    2
300    2
301    2
302    1
Name: Slope, Length: 303, dtype: int64
```

In [34]:

```
plt.title("Line graph to show: ")
plt.xlabel("Sex")
plt.ylabel("Slope")
plt.plot(x,y)
```

Out[34]:

```
[<matplotlib.lines.Line2D at 0x216c39880d0>]
```



In [35]:

```
x=df1.Chol
x
```

Out[35]:

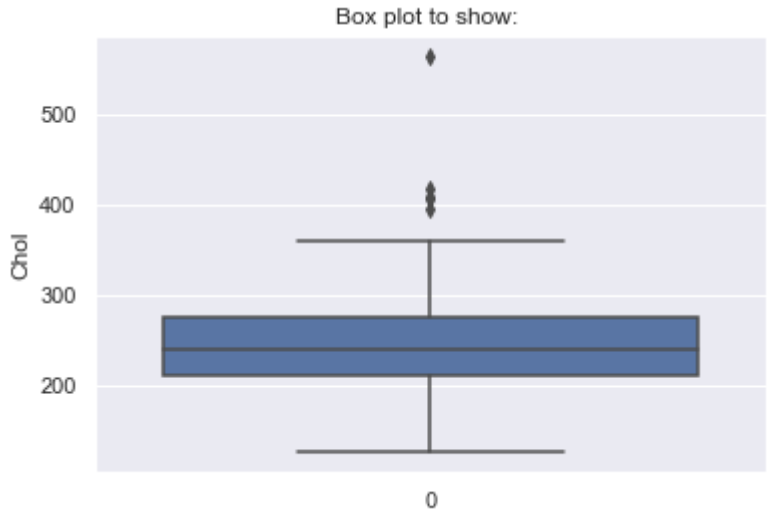
```
0      233
1      286
2      229
3      250
4      204
      ...
298    264
299    193
300    131
301    236
302    175
Name: Chol, Length: 303, dtype: int64
```

In [36]:

```
plt.title("Box plot to show: ")
plt.ylabel("Chol")
sns.boxplot(x)
```

Out[36]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c3925370>
```



In [37]:

```
x=df1.Ca
x
```

Out[37]:

```
0      0.0
1      3.0
2      2.0
3      0.0
4      0.0
      ...
298    0.0
299    2.0
300    1.0
301    1.0
302    NaN
Name: Ca, Length: 303, dtype: float64
```

In [38]:

```
y=df1.Slope
y
```

Out[38]:

```
0      3
1      2
2      2
3      3
4      1
      ..
298    2
299    2
300    2
301    2
302    1
Name: Slope, Length: 303, dtype: int64
```

In [39]:

```
df1.fillna(method="ffill",inplace=True)
```

In [40]:

```
df1.isnull().sum()
```

Out[40]:

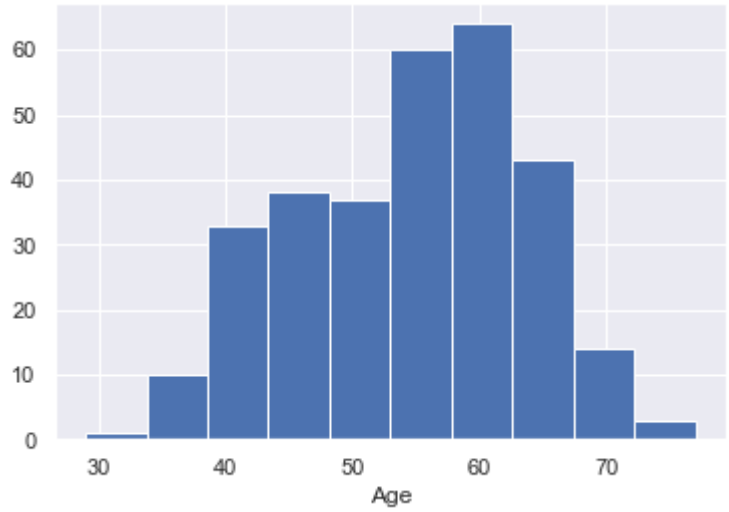
```
Unnamed: 0    0
Age           0
Sex           0
ChestPain     0
RestBP        0
Chol          0
Fbs           0
RestECG       0
MaxHR         0
ExAng         0
Oldpeak       0
Slope         0
Ca            0
Thal          0
AHD           0
dtype: int64
```

In [41]:

```
plt.hist(df1['Age'])
plt.xlabel('Age')
plt.plot()
```

Out[41]:

```
[]
```

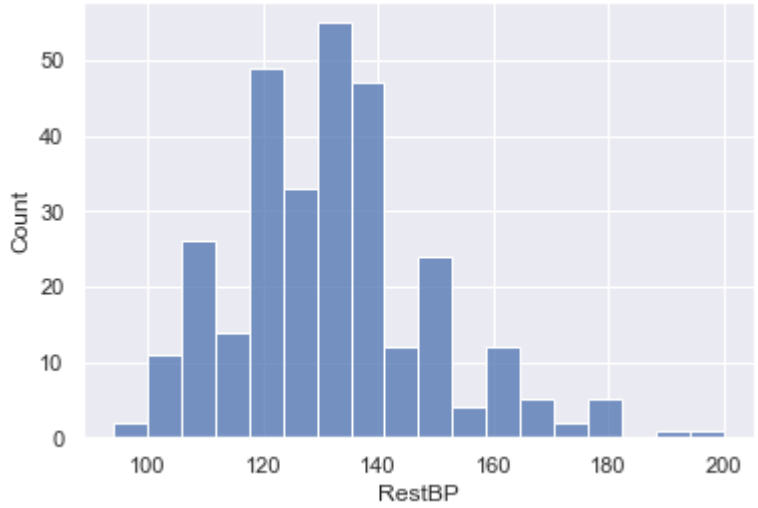


In [50]:

```
sns.histplot(df1['RestBP'])
```

Out[50]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c34cbd90>
```

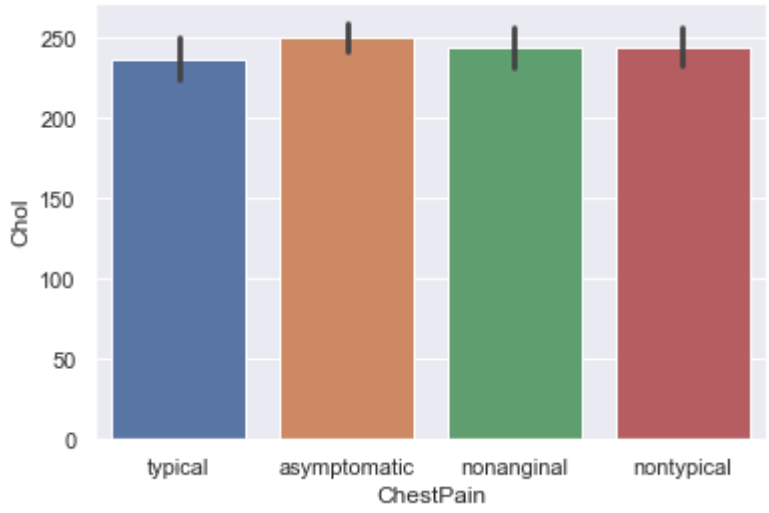


In [43]:

```
sns.barplot(y='Chol', x="ChestPain", data=df1)
```

Out[43]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c3b224c0>
```



In [44]:

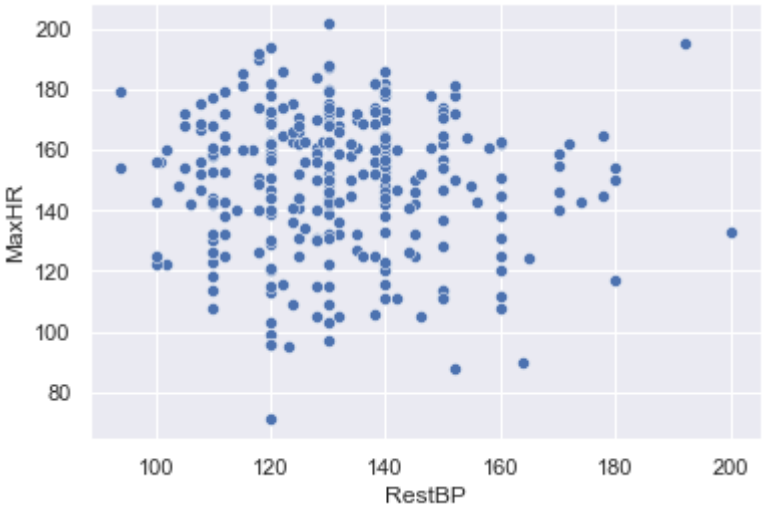
```
sns.lineplot(x="Age", y="Chol", data=df1)
plt.show()
```

In [45]:

```
sns.scatterplot(x="RestBP", y="MaxHR", data=df1)
```

Out[45]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c3bc4d00>
```

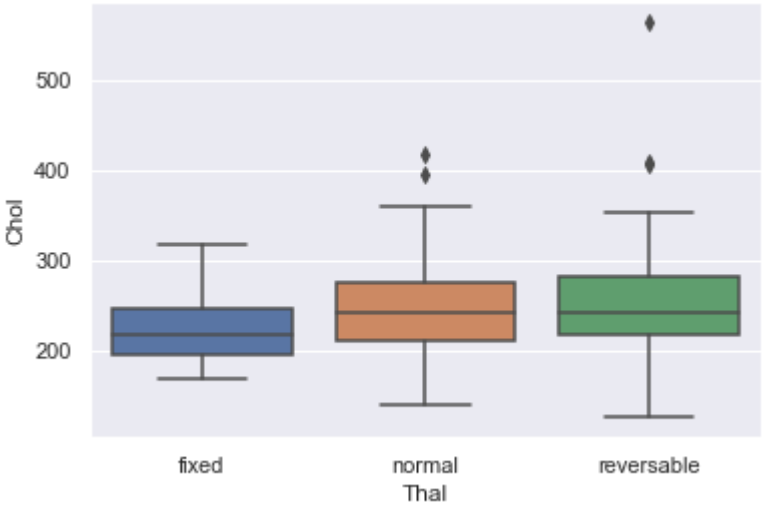


In [46]:

```
sns.boxplot(y='Chol', x="Thal", data=df1)
```

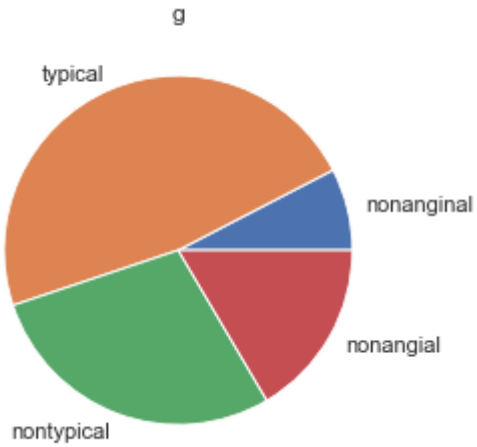
Out[46]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c3c119a0>
```



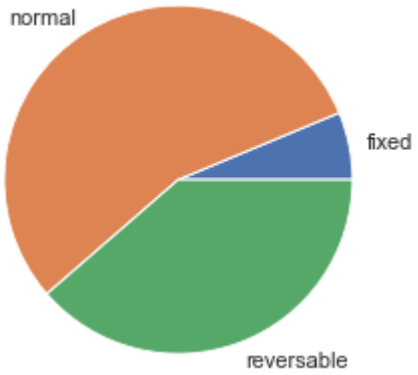
In [47]:

```
data = [df1.ChestPain[df1.ChestPain == "typical"].count(),df1.ChestPain[df1.ChestPain ==
       df1.ChestPain[df1.ChestPain == "nonanginal"].count(),df1.ChestPain[df1.ChestPain
labels = ["nonanginal","typical","nontypical","nonangial"]
plt.pie(data,labels=labels)
plt.title("g")
plt.show()
```



In [48]:

```
data=[df1.Thal[df1.Thal=="fixed"].count(),df1.Thal[df1.Thal=="normal"].count(),df1.Thal[
labels=["fixed","normal","reversable"]
plt.pie(data,labels=labels)
plt.show()
```

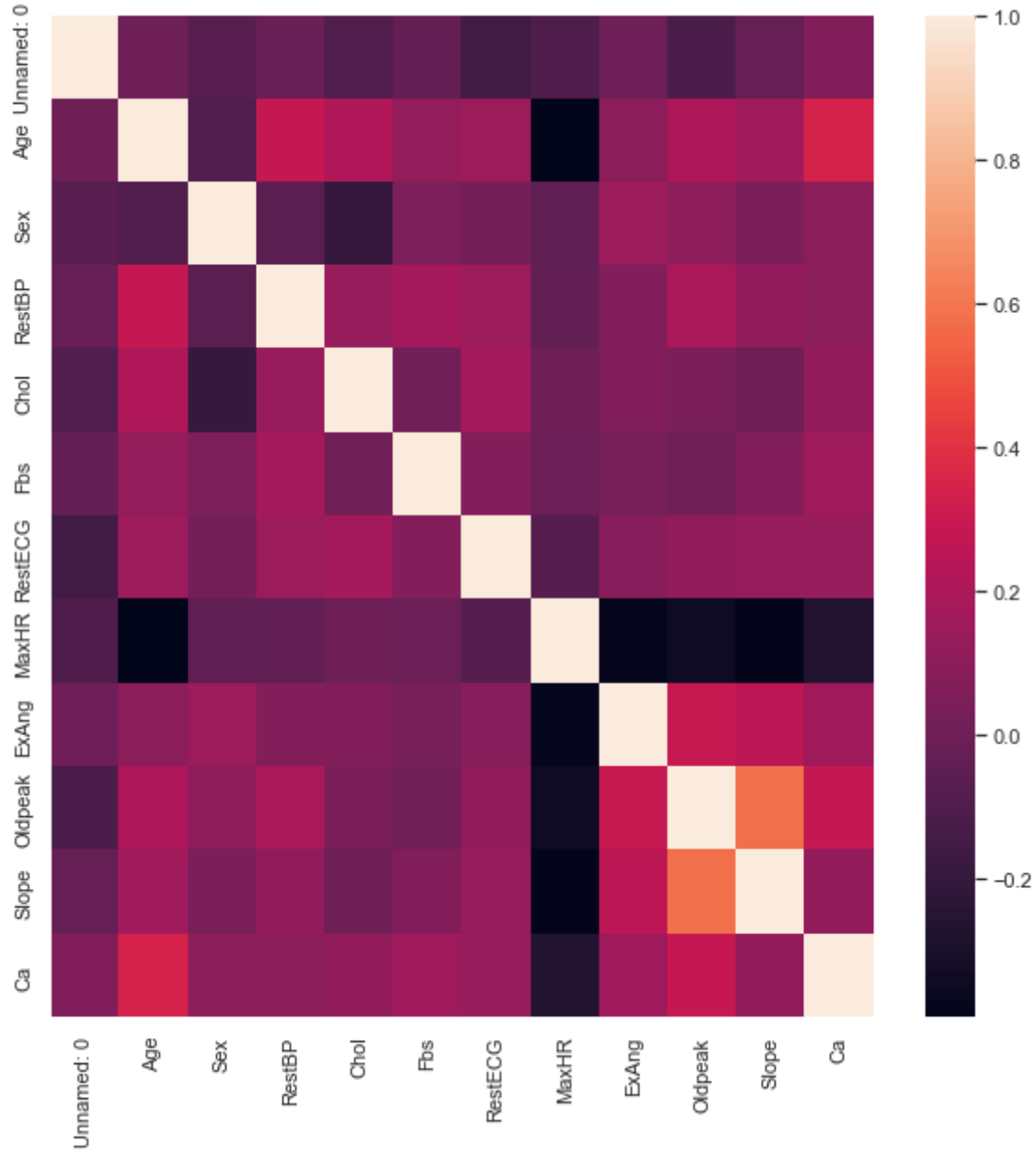


In [62]:

```
fig=plt.figure(figsize=(10,10))
sns.heatmap(df1.corr())
```

Out[62]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x216c562d610>
```



In [ ]:

In [ ]:

In [ ]: