In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib.figure import Figure
```

In [2]:
```
df=pd.read_csv("Heart.csv")
```

In [3]:
```
df
```

Out[3]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak
0	1	63	1	typical	145	233	1	2	150	0	2.3
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5
4	5	41	0	nontypical	130	204	0	2	172	0	1.4
...	...	...	...	...	...	...	...	...	...	...	...
298	299	45	1	typical	110	264	0	0	132	0	1.2
299	300	68	1	asymptomatic	144	193	1	0	141	0	3.4
300	301	57	1	asymptomatic	130	131	0	0	115	1	1.2
301	302	57	0	nontypical	130	236	0	2	174	0	0.0
302	303	38	1	nonanginal	138	175	0	0	173	0	0.0

303 rows × 15 columns

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 15 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  303 non-null    int64  
 1   Age         303 non-null    int64  
 2   Sex         303 non-null    int64  
 3   ChestPain   303 non-null    object 
 4   RestBP      303 non-null    int64  
 5   Chol        303 non-null    int64  
 6   Fbs         303 non-null    int64  
 7   RestECG     303 non-null    int64  
 8   MaxHR       303 non-null    int64  
 9   ExAng       303 non-null    int64  
 10  Oldpeak     303 non-null    float64
 11  Slope       303 non-null    int64  
 12  Ca          299 non-null    float64
 13  Thal        301 non-null    object 
 14  AHD         303 non-null    object 
dtypes: float64(2), int64(10), object(3)
memory usage: 35.6+ KB
```

In [5]:
```
df.describe()
```

Out[5]:

	Unnamed: 0	Age	Sex	RestBP	Chol	Fbs	RestECG	
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.
mean	152.000000	54.438944	0.679868	131.689769	246.693069	0.148515	0.990099	149.
std	87.612784	9.038662	0.467299	17.599748	51.776918	0.356198	0.994971	22.
min	1.000000	29.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.
25%	76.500000	48.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.
50%	152.000000	56.000000	1.000000	130.000000	241.000000	0.000000	1.000000	153.
75%	227.500000	61.000000	1.000000	140.000000	275.000000	0.000000	2.000000	166.
max	303.000000	77.000000	1.000000	200.000000	564.000000	1.000000	2.000000	202.

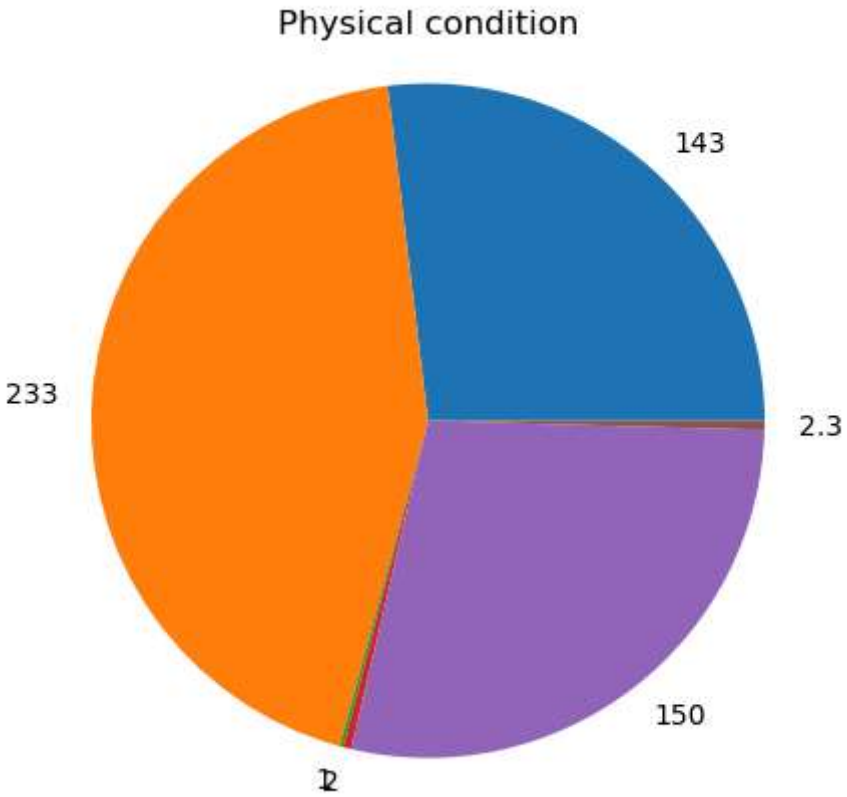
In [6]:
```
df.head()
```

Out[6]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak
0	1	63	1	typical	145	233	1	2	150	0	2.3
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5
4	5	41	0	nontypical	130	204	0	2	172	0	1.4

In [7]:
```
#pie chart ploting
```

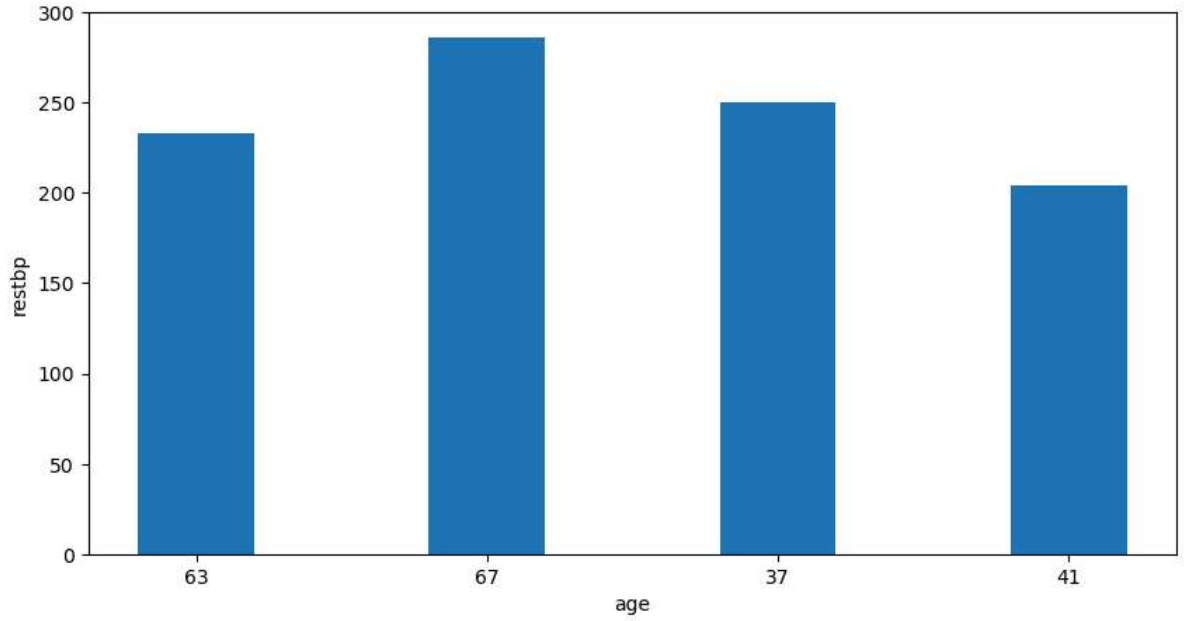
In [8]:
```
size=[143,233,1,2,150,2.3]
labels='RestBP','Chol','Fbs','RestECG','oldpeak'
plt.pie(size,labels=size)
plt.title("Physical condition")
plt.axis("equal")
plt.show()
```



In [9]:
```
#Bar plot
```

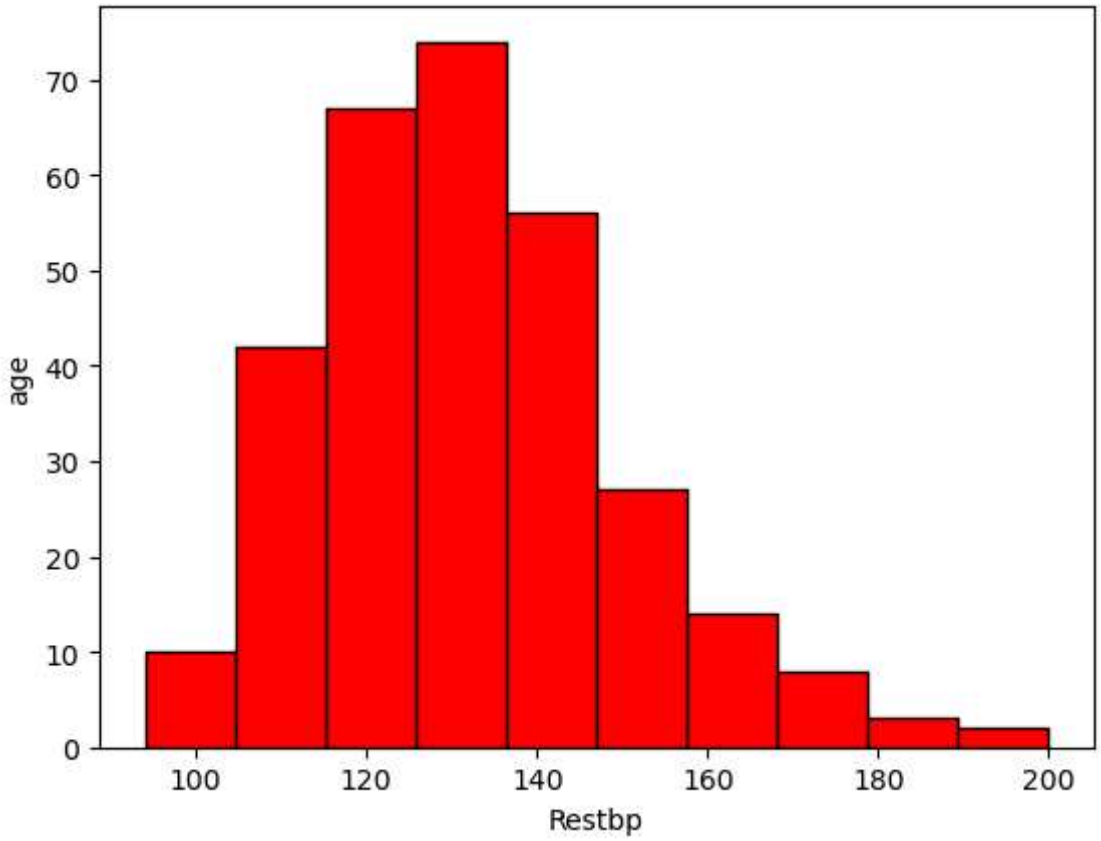
In [10]:
```
data={'63':233,'67':286,'37':250,'41':204}
age=list(data.keys())
values=list(data.values())
fig=plt.figure(figsize=(10,5))
plt.bar(age,values,width=0.4)
plt.xlabel("age")
plt.ylabel("restbp")
```

Out[10]:
```
Text(0, 0.5, 'restbp')
```



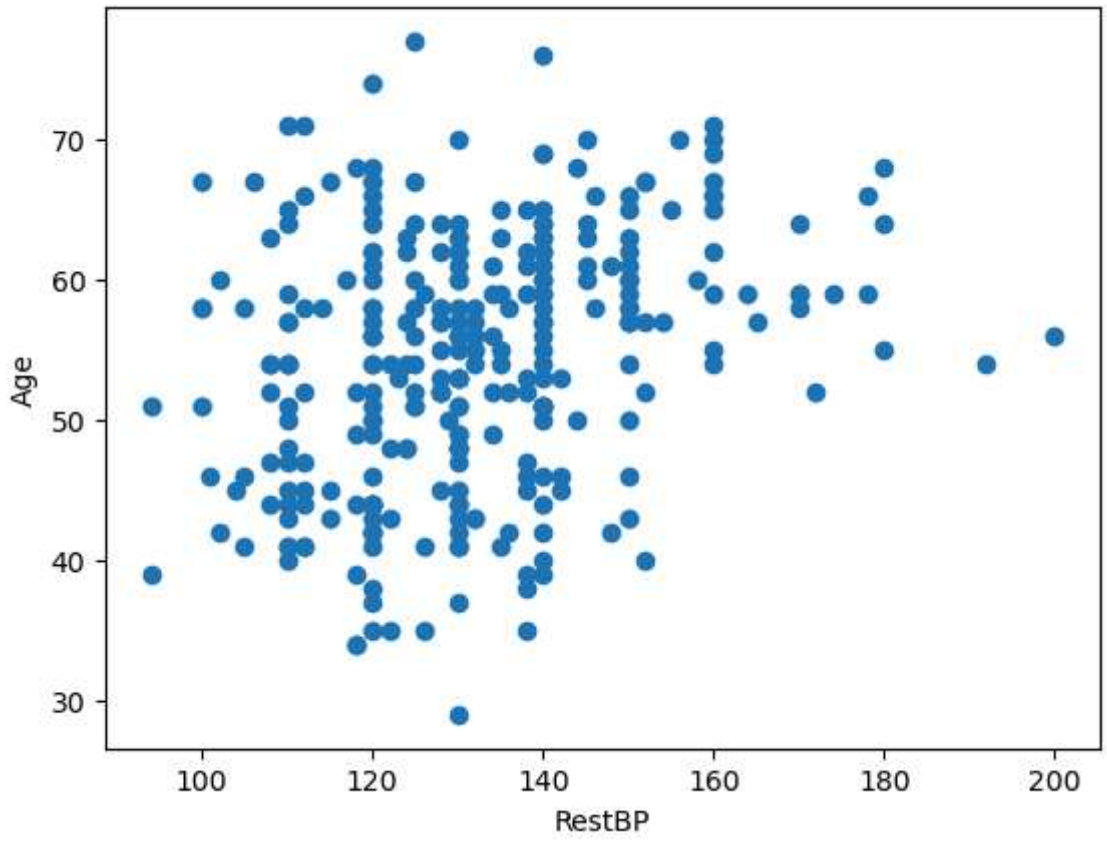
In [11]:
```
#histogram
```

In [12]:
```
data=df.RestBP
plt.hist(data,edgecolor='black',color='red')
plt.xlabel("Restbp")
plt.ylabel("age")
plt.show()
```



In [13]:
```
data=df.RestBP
y=df.Age
plt.scatter(data,y)
plt.xlabel("RestBP")
plt.ylabel("Age")
```

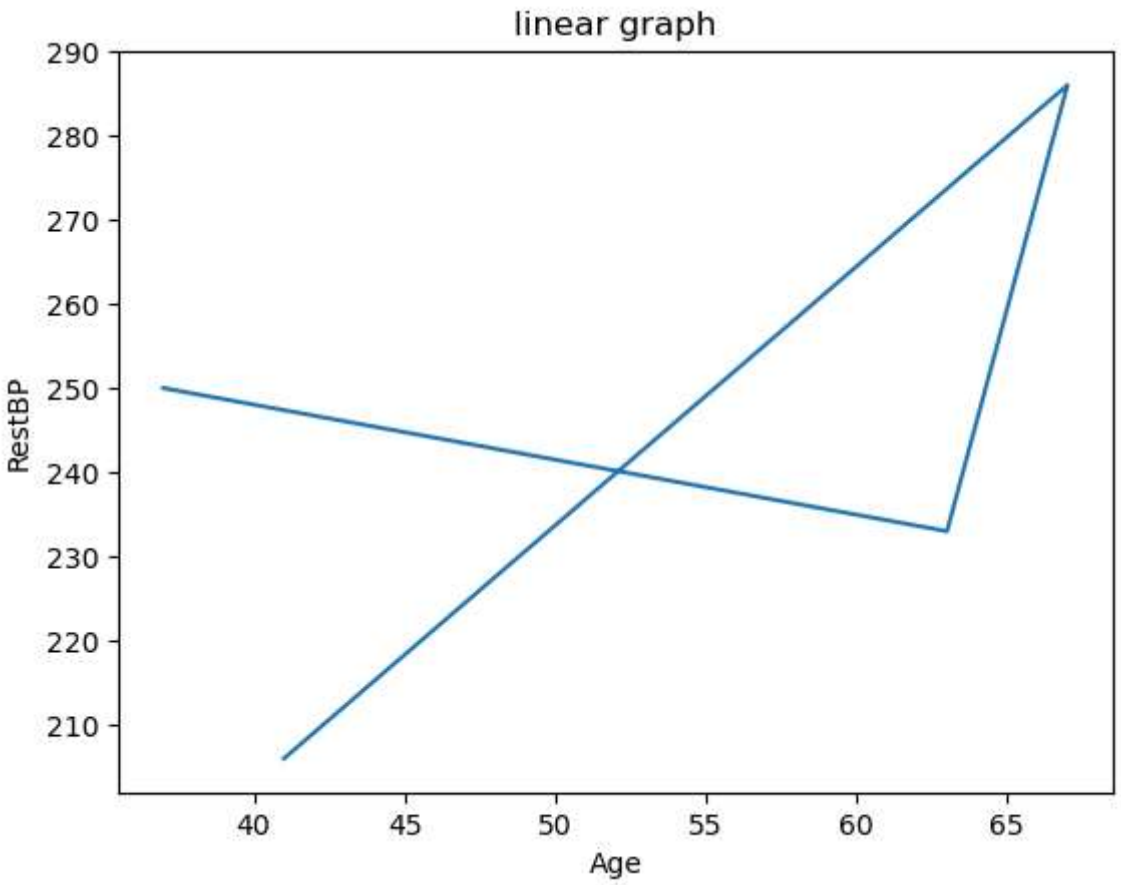
Out[13]:
```
Text(0, 0.5, 'Age')
```



In [14]:
```
#linear graph
```

In [15]:
```
x=[37,63,67,41]
y=[250,233,286,206]
plt.plot(x,y)
plt.title("linear graph")
plt.xlabel("Age")
plt.ylabel("RestBP")
```

Out[15]:
```
Text(0, 0.5, 'RestBP')
```

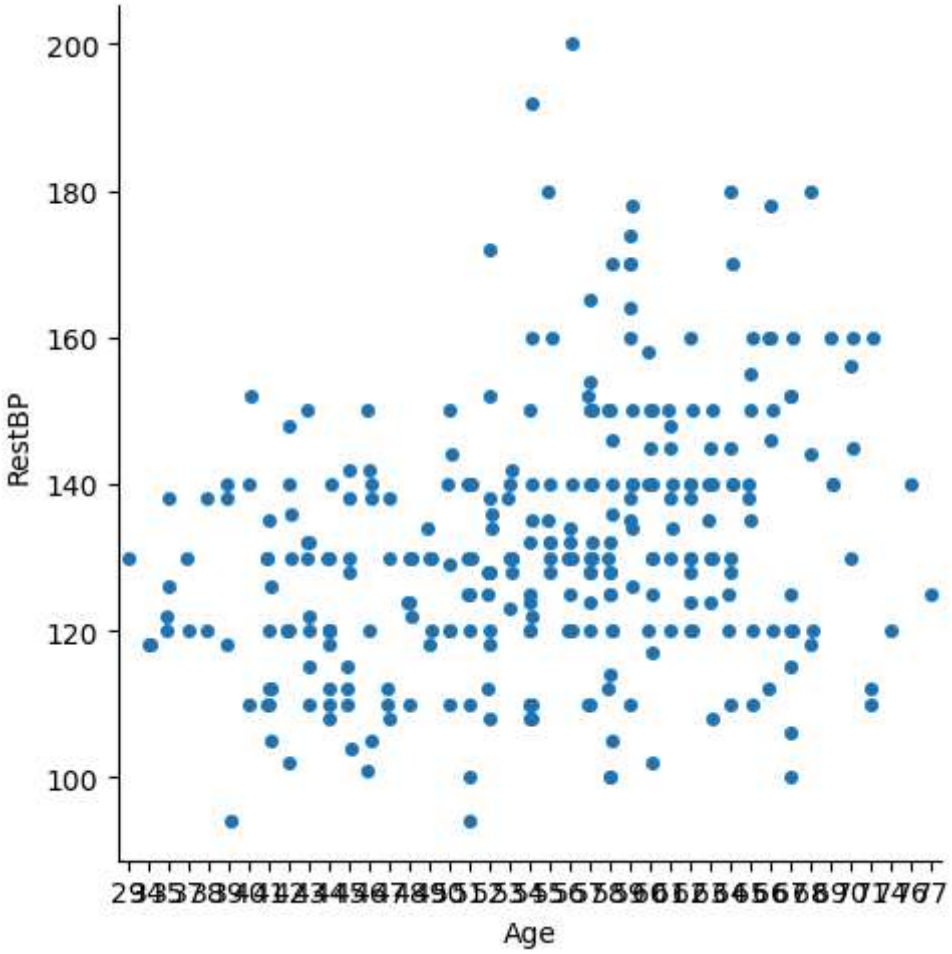


In [16]:
```
heart=pd.read_csv("Heart.csv")
```

In [17]:
```
sns.catplot(x='Age',y='RestBP',data=heart)
```

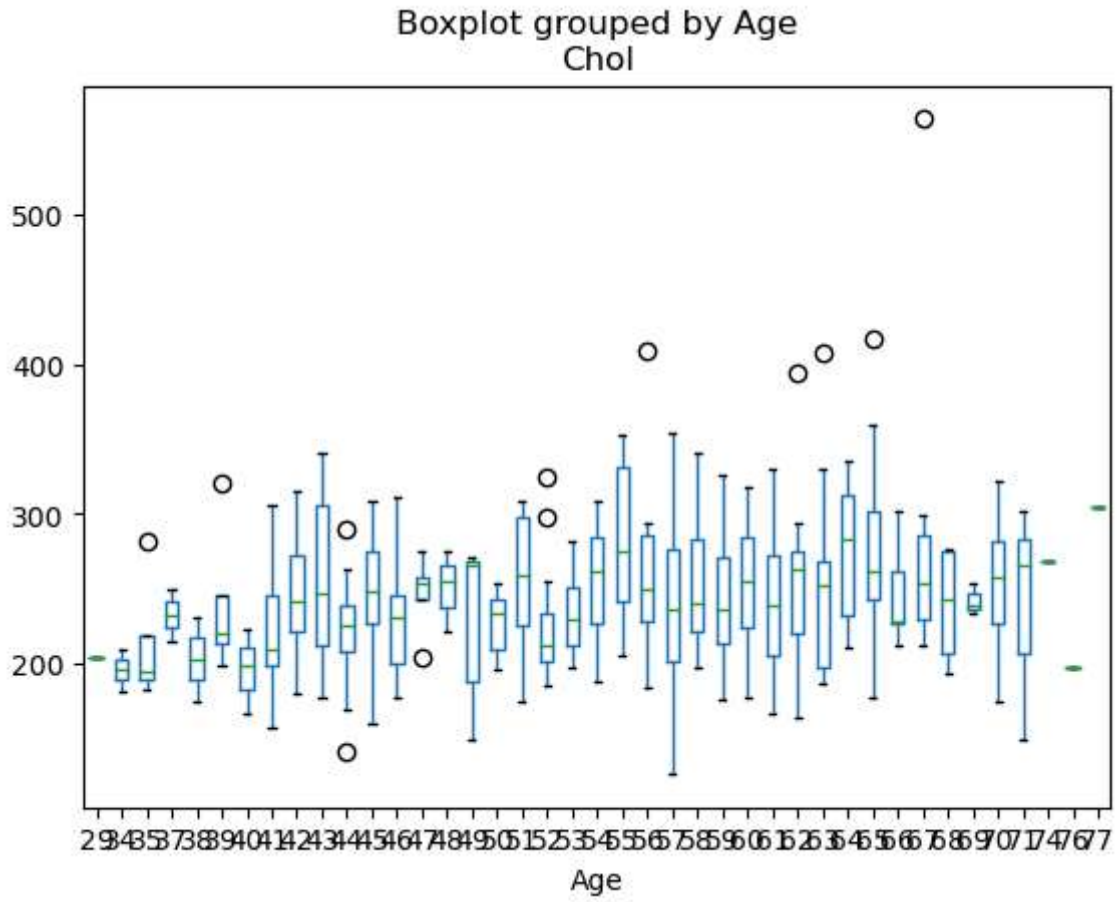
```
C:\Users\Tushar\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarn
ing: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

Out[17]:
```
<seaborn.axisgrid.FacetGrid at 0x1b51e072010>
```

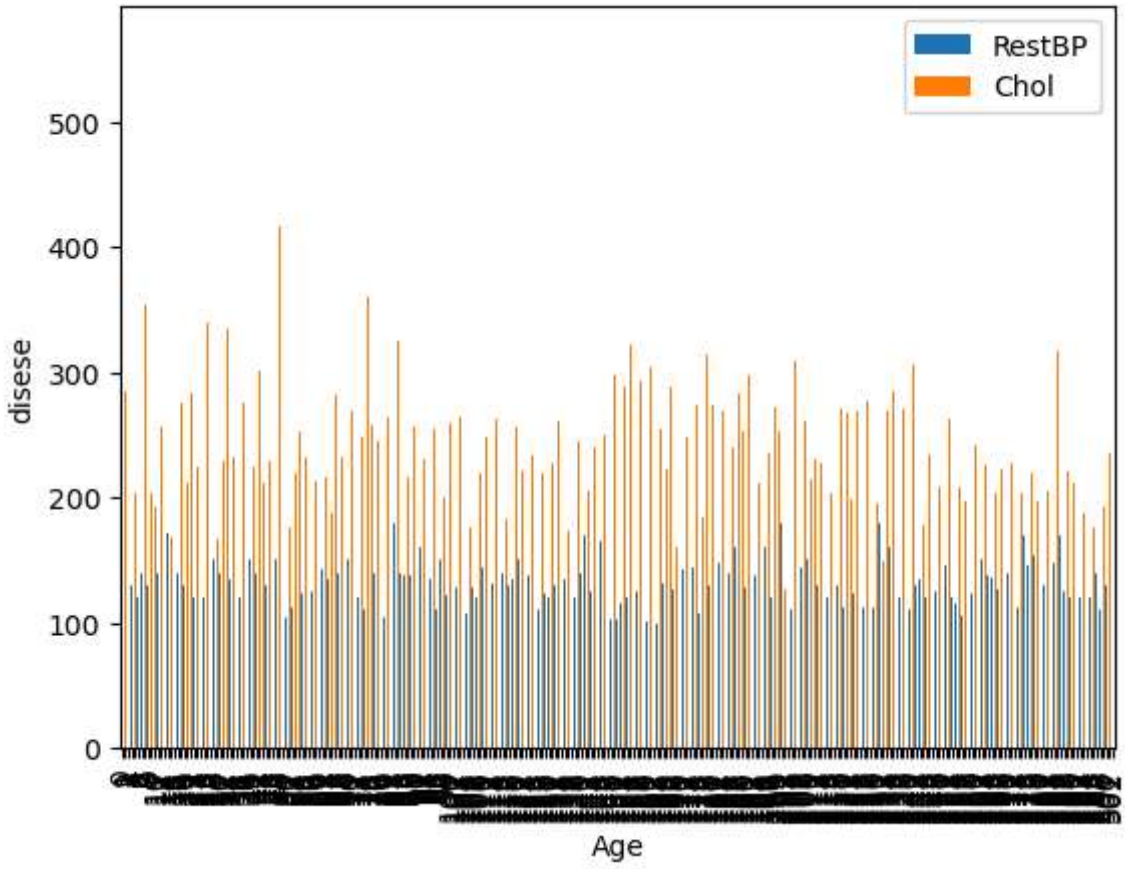


In [18]:
```
df.boxplot(by='Age',column=['Chol'],grid=False)
```

Out[18]:
```
<Axes: title={'center': 'Chol'}, xlabel='Age'>
```



In [19]:
```
data=df[["RestBP","Chol"]]
data.plot(kind='bar')
plt.xlabel("Age")
plt.ylabel("disese")
plt.show()
```



In [ ]:
```

```

In [20]:
```
#Visualization for Facebook dataset
```

In [21]:
```
df1=pd.read_csv("dataset_Facebook.csv",sep=';')
```

In [22]:
```
df1
```

Out[22]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	C
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 [23]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 15 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  303 non-null    int64  
 1   Age         303 non-null    int64  
 2   Sex         303 non-null    int64  
 3   ChestPain   303 non-null    object 
 4   RestBP      303 non-null    int64  
 5   Chol        303 non-null    int64  
 6   Fbs         303 non-null    int64  
 7   RestECG     303 non-null    int64  
 8   MaxHR       303 non-null    int64  
 9   ExAng       303 non-null    int64  
 10  Oldpeak     303 non-null    float64
 11  Slope       303 non-null    int64  
 12  Ca          299 non-null    float64
 13  Thal        301 non-null    object 
 14  AHD         303 non-null    object 
dtypes: float64(2), int64(10), object(3)
memory usage: 35.6+ KB
```

In [24]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 19 columns):
 #   Column                                                  Non
-Null Count  Dtype
---  ------                                                  ---
-----------  -----
 0   Page total likes                                        500
non-null    int64
 1   Type                                                    500
non-null    object
 2   Category                                                500
non-null    int64
 3   Post Month                                              500
non-null    int64
 4   Post Weekday                                            500
non-null    int64
 5   Post Hour                                               500
non-null    int64
 6   Paid                                                    499
non-null    float64
 7   Lifetime Post Total Reach                               500
non-null    int64
 8   Lifetime Post Total Impressions                         500
non-null    int64
 9   Lifetime Engaged Users                                  500
non-null    int64
 10  Lifetime Post Consumers                                 500
non-null    int64
 11  Lifetime Post Consumptions                              500
non-null    int64
 12  Lifetime Post Impressions by people who have liked your Page  500
non-null    int64
 13  Lifetime Post reach by people who like your Page        500
non-null    int64
 14  Lifetime People who have liked your Page and engaged with your post  500
non-null    int64
 15  comment                                                 500
non-null    int64
 16  like                                                    499
non-null    float64
 17  share                                                   496
non-null    float64
 18  Total Interactions                                      500
non-null    int64
dtypes: float64(3), int64(15), object(1)
memory usage: 74.3+ KB
```

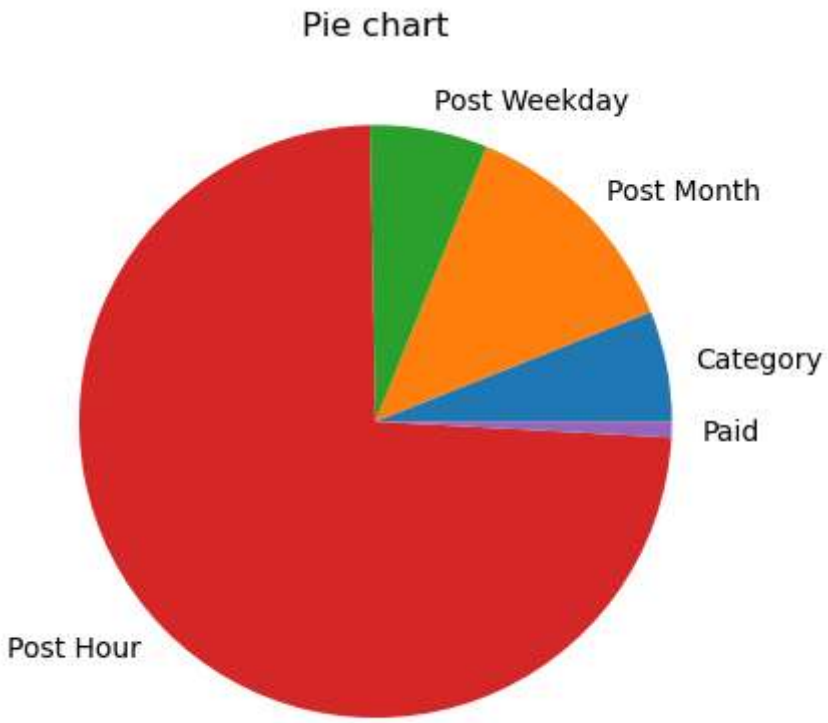
In [25]:
```
df1.describe()
```

Out[25]:

	Page total likes	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach
count	500.000000	500.000000	500.000000	500.000000	500.000000	499.000000	500.00000
mean	123194.176000	1.880000	7.038000	4.150000	7.840000	0.278557	13903.36000
std	16272.813214	0.852675	3.307936	2.030701	4.368589	0.448739	22740.78789
min	81370.000000	1.000000	1.000000	1.000000	1.000000	0.000000	238.00000
25%	112676.000000	1.000000	4.000000	2.000000	3.000000	0.000000	3315.00000
50%	129600.000000	2.000000	7.000000	4.000000	9.000000	0.000000	5281.00000
75%	136393.000000	3.000000	10.000000	6.000000	11.000000	1.000000	13168.00000
max	139441.000000	3.000000	12.000000	7.000000	23.000000	1.000000	180480.00000

In [26]:
```
#piechart plotting
```

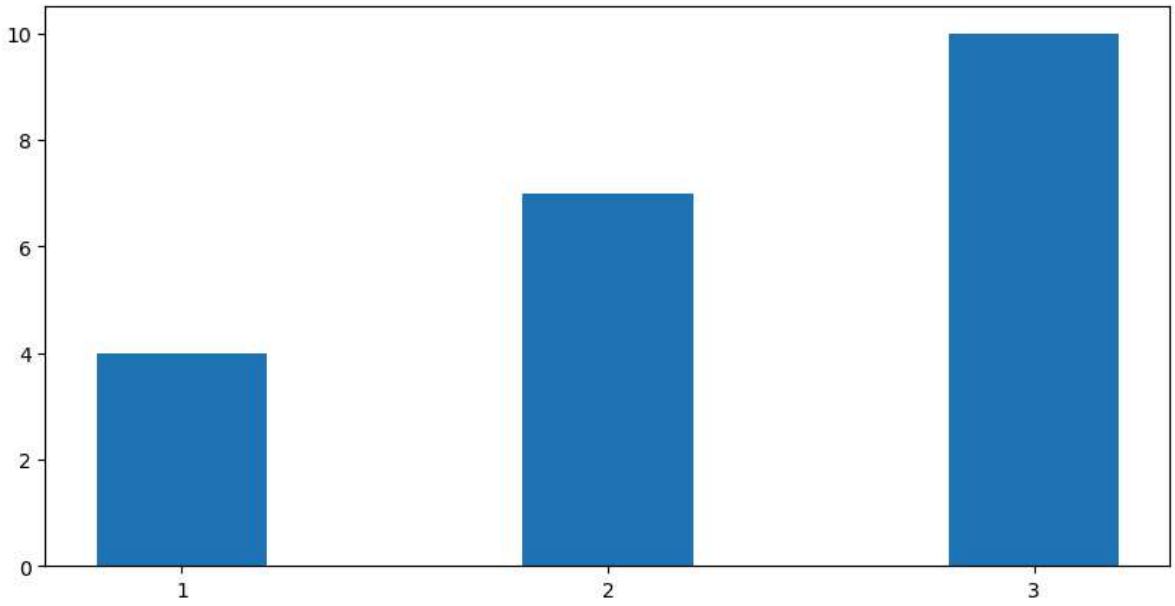
In [27]:
```
size=[1.88,4,2,23,0.27]
labels='Category','Post Month','Post Weekday','Post Hour','Paid'
plt.pie(size,labels=labels)
plt.title("Pie chart")
plt.show()
```



In [64]:
```
fig=plt.figure(figsize=(10,5))
data={'1':4,'2':7,'3':10}
Category=list(data.keys())
values=list(data.values())
plt.bar(Category,values,width=0.4)
plt.xlabel("Category")
plt.ylabel("Post Month")
plt.show()
```

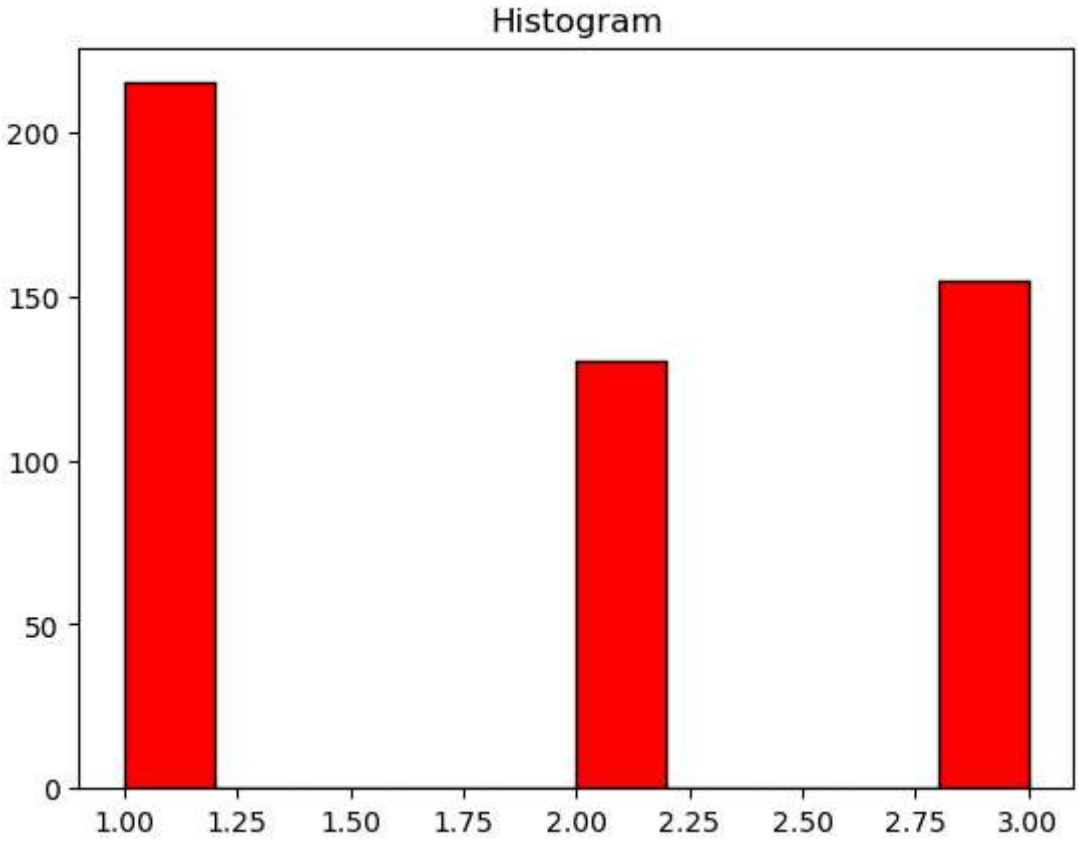
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[64], line 6
      4 values=list(data.values())
      5 plt.bar(Category,values,width=0.4)
----> 6 plt.xlabel("Category")
      7 plt.ylabel("Post Month")
      8 plt.show()

TypeError: 'str' object is not callable
```



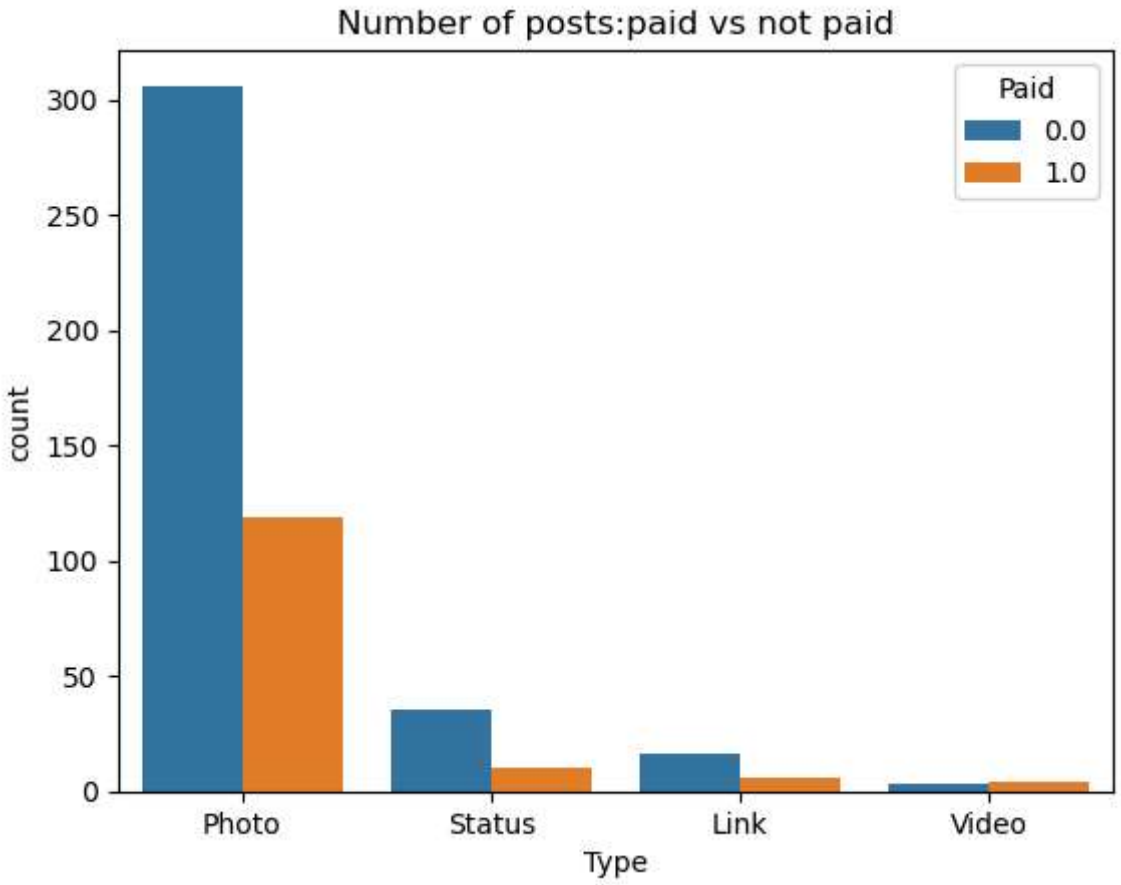
In [39]:
```
data=df1.Category
plt.hist(data,color='red',edgecolor='black')

plt.title("Histogram")
plt.show()
```



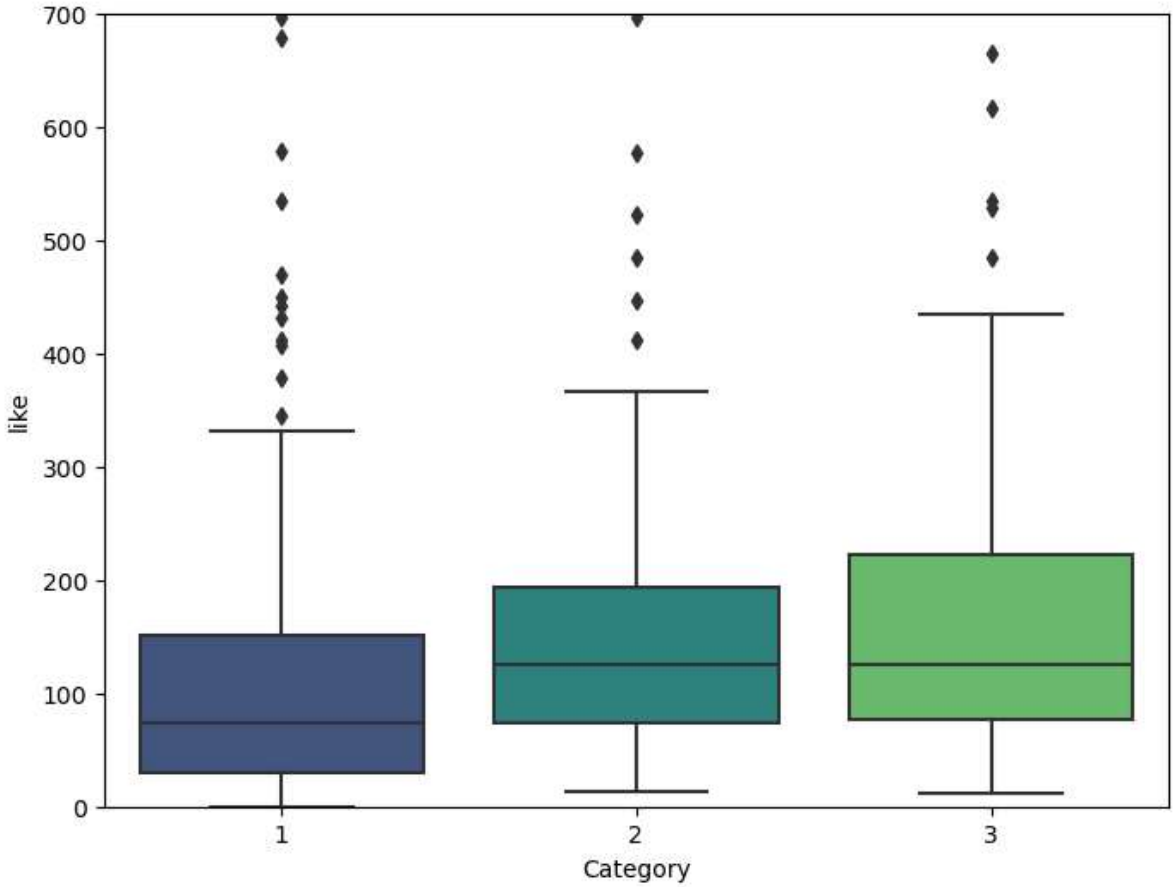
In [38]:
```
sns.countplot(x='Type',hue='Paid',data=df1)
plt.title("Number of posts:paid vs not paid")
```

Out[38]: Text(0.5, 1.0, 'Number of posts:paid vs not paid')



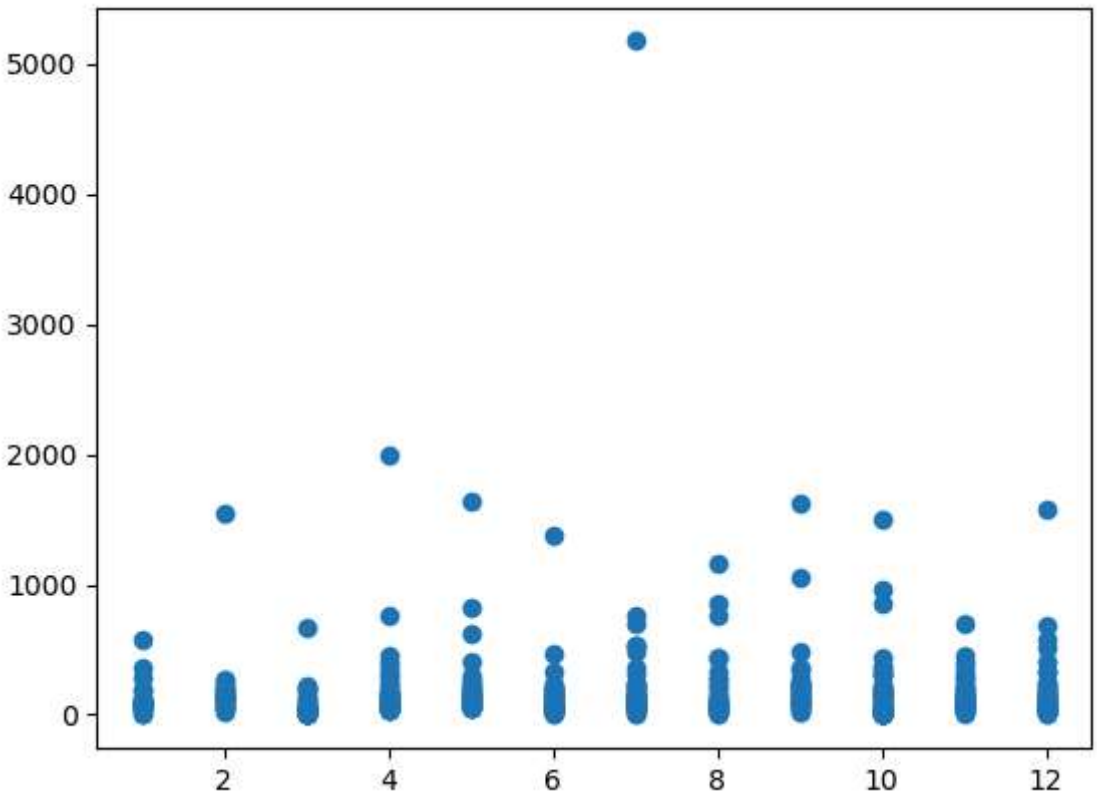
In [46]:
```
plt.figure(figsize=(8,6))
sns.boxplot(x=df1['Category'],y=df1['like'],data=df1,palette='viridis')
plt.ylim(0,700)
```

Out[46]: (0.0, 700.0)



In [54]:
```
x=df1['Post Month']
y=df1['like']
plt.scatter(x,y)
```

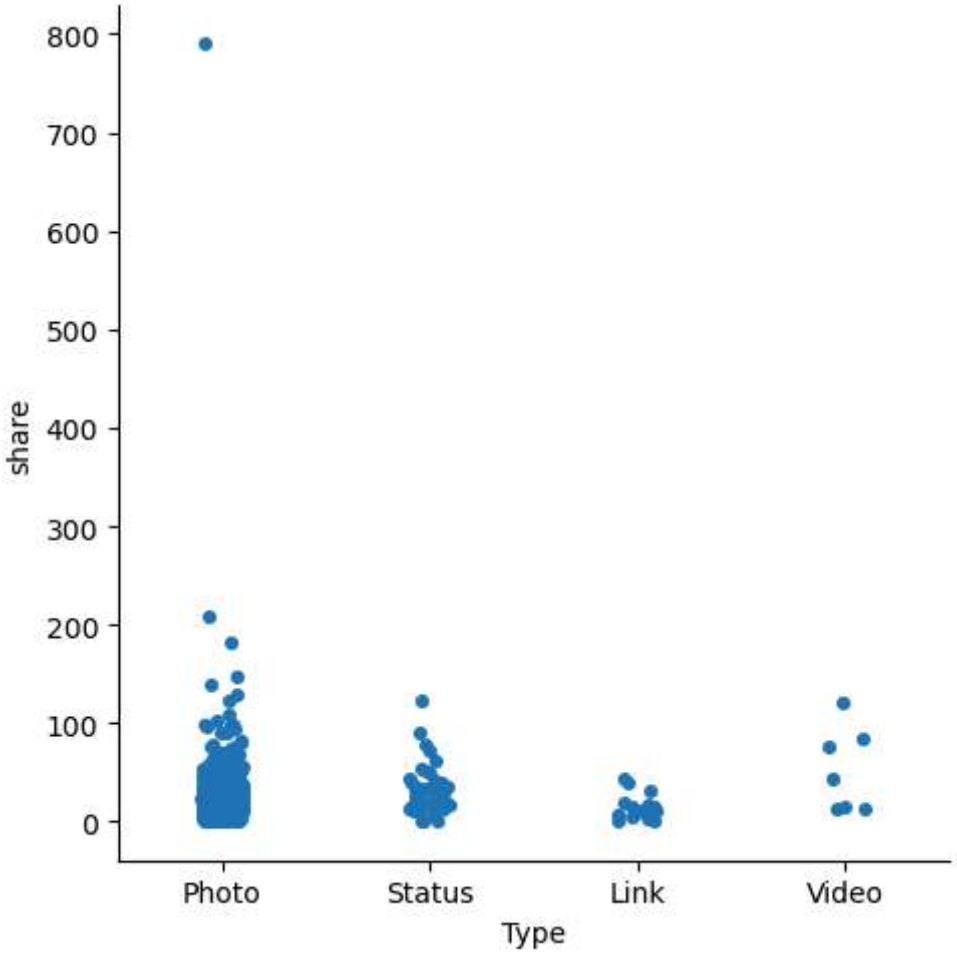
Out[54]: <matplotlib.collections.PathCollection at 0x1b5223c9b10>



In [58]:
```
sns.catplot(x='Type',y='share',data=df1)
```

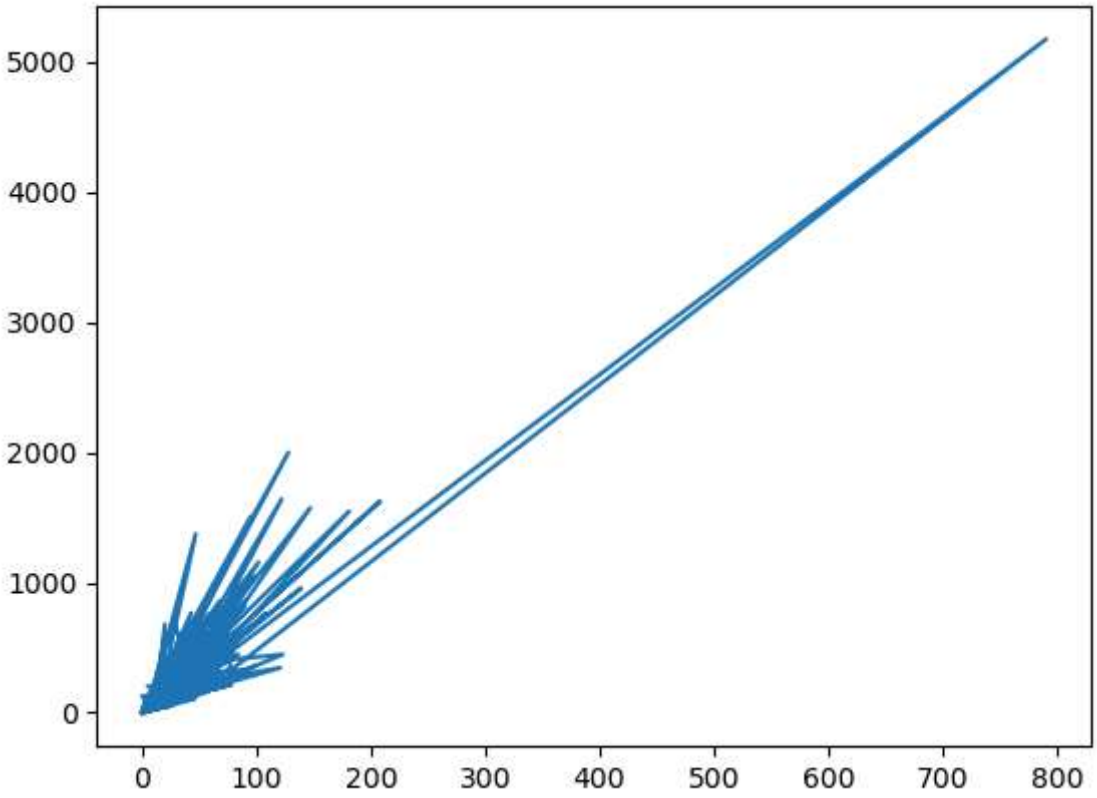
```
C:\Users\Tushar\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarn
ing: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

Out[58]:
```
<seaborn.axisgrid.FacetGrid at 0x1b522b3c390>
```



In [61]:
```
x=df1['share']
y=df1['like']
plt.plot(x,y)
```

Out[61]: [<matplotlib.lines.Line2D at 0x1b524fe1390>]



In [ ]: