In [139]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

In [140]:
```
fb=pd.read_csv("Facebook.csv")
```

In [141]:
```
fb.shape
```

Out[141]: (500, 19)

In [142]:
```
fb.head()
```

Out[142]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Post Impressions by people who have liked your Page	Lifetime Post reach by people who like your Page	L	e
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109	159	3078	1640		
1	139441	Status	2	12	3	10	0.0	10460	19057	1457	1361	1674	11710	6112		
2	139441	Photo	3	12	3	3	0.0	2413	4373	177	113	154	2812	1503		
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211	790	1119	61027	32048		
4	139441	Photo	2	12	2	3	0.0	7244	13594	671	410	580	6228	3200		

In [143]:
```
fb.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 [144]:
```
fb.isnull().values.any()
```

Out[144]:
```
True
```

In [145]:
```
fb.describe()
```

Out[145]:

	Page total likes	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Life Cons
count	500.000000	500.000000	500.000000	500.000000	500.000000	499.000000	500.00000	5.000000e+02	500.000000	500.000000	5(
mean	123194.176000	1.880000	7.038000	4.150000	7.840000	0.278557	13903.36000	2.958595e+04	920.344000	798.772000	14
std	16272.813214	0.852675	3.307936	2.030701	4.368589	0.448739	22740.78789	7.680325e+04	985.016636	882.505013	20
min	81370.000000	1.000000	1.000000	1.000000	1.000000	0.000000	238.00000	5.700000e+02	9.000000	9.000000	
25%	112676.000000	1.000000	4.000000	2.000000	3.000000	0.000000	3315.00000	5.694750e+03	393.750000	332.500000	5(
50%	129600.000000	2.000000	7.000000	4.000000	9.000000	0.000000	5281.00000	9.051000e+03	625.500000	551.500000	8!
75%	136393.000000	3.000000	10.000000	6.000000	11.000000	1.000000	13168.00000	2.208550e+04	1062.000000	955.500000	14(
max	139441.000000	3.000000	12.000000	7.000000	23.000000	1.000000	180480.00000	1.110282e+06	11452.000000	11328.000000	197

In [146]:
```
fb.mean()
```

Out[146]:
```
Page total likes                                                  123194.176000
Category                                                               1.880000
Post Month                                                             7.038000
Post Weekday                                                           4.150000
Post Hour                                                              7.840000
Paid                                                                   0.278557
Lifetime Post Total Reach                                          13903.360000
Lifetime Post Total Impressions                                    29585.948000
Lifetime Engaged Users                                               920.344000
Lifetime Post Consumers                                              798.772000
Lifetime Post Consumptions                                          1415.130000
Lifetime Post Impressions by people who have liked your Page       16766.376000
Lifetime Post reach by people who like your Page                    6585.488000
Lifetime People who have liked your Page and engaged with your post  609.986000
comment                                                                7.482000
like                                                                 177.945892
share                                                                 27.266129
Total Interactions                                                   212.120000
dtype: float64
```

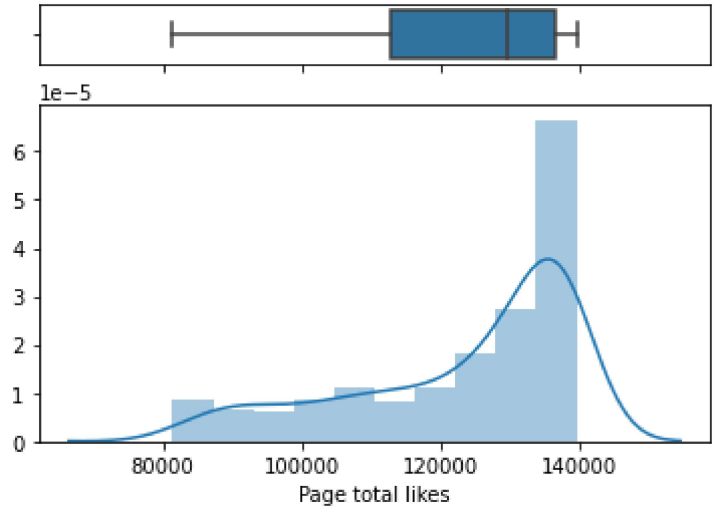
Filling missing values

In [147]:
```
fb['Paid'].fillna(value=fb['Paid'].mean(), inplace=True)
fb['like'].fillna(value=fb['like'].mean(), inplace=True)
fb['share'].fillna(value=fb['share'].mean(), inplace=True)
```

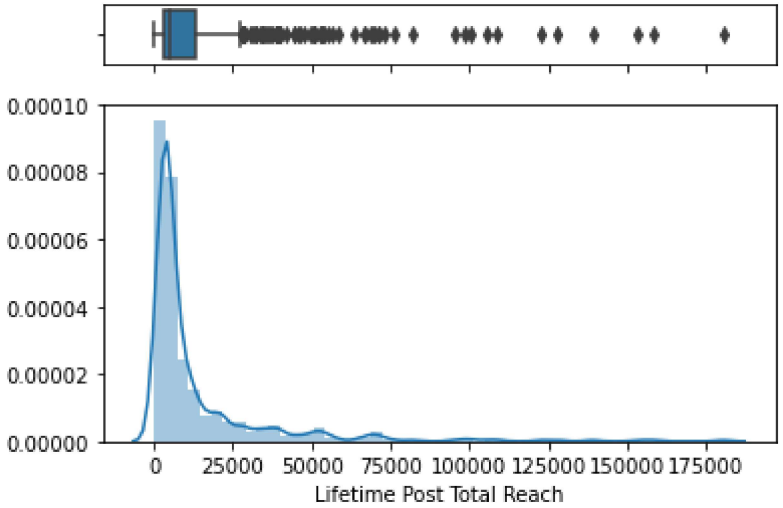
In [148]:
```
fb.info()
```

```
<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                                                    500 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                                                    500 non-null    float64
 17  share                                                   500 non-null    float64
 18  Total Interactions                                      500 non-null    int64  
dtypes: float64(3), int64(15), object(1)
memory usage: 74.3+ KB
```

In [149]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Page total likes"], ax=ax_box)
sns.distplot(fb["Page total likes"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Page total likes.png")
```



In [150]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Total Reach"], ax=ax_box)
sns.distplot(fb["Lifetime Post Total Reach"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Total Reach.png")
```

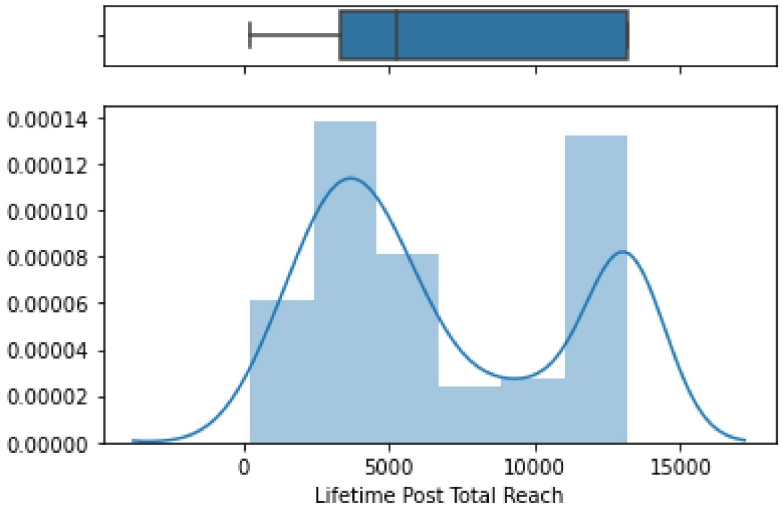


In [151]:
```
print(fb['Lifetime Post Total Reach'].quantile(0.10))
print(fb['Lifetime Post Total Reach'].quantile(0.75))
```

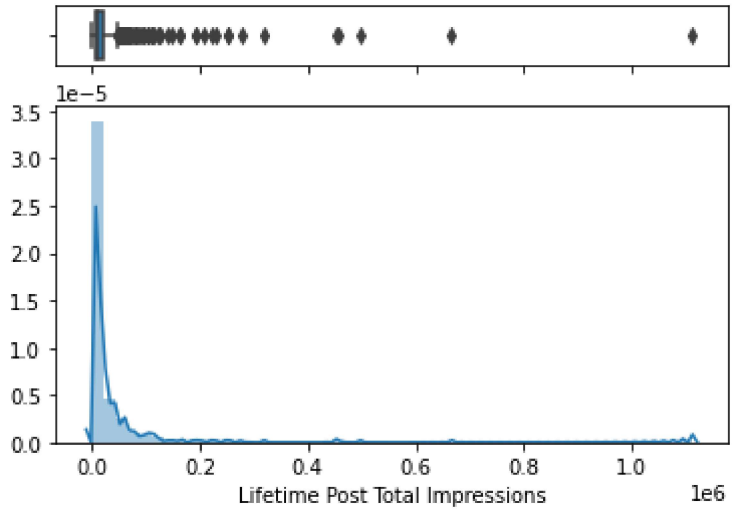
```
2044.9000000000005
13168.0
```

In [152]:
```
fb['Lifetime Post Total Reach'] = np.where(fb['Lifetime Post Total Reach'] >13168 ,13168,fb['Lifetime Post Total Reac
h'])
```

In [153]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Total Reach"], ax=ax_box)
sns.distplot(fb["Lifetime Post Total Reach"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Total Reach(1).png")
```



In [154]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Total Impressions"], ax=ax_box)
sns.distplot(fb["Lifetime Post Total Impressions"], ax=ax_hist)


ax_box.set(xlabel='')
plt.savefig("Lifetime Post Total Impressions.png")
```

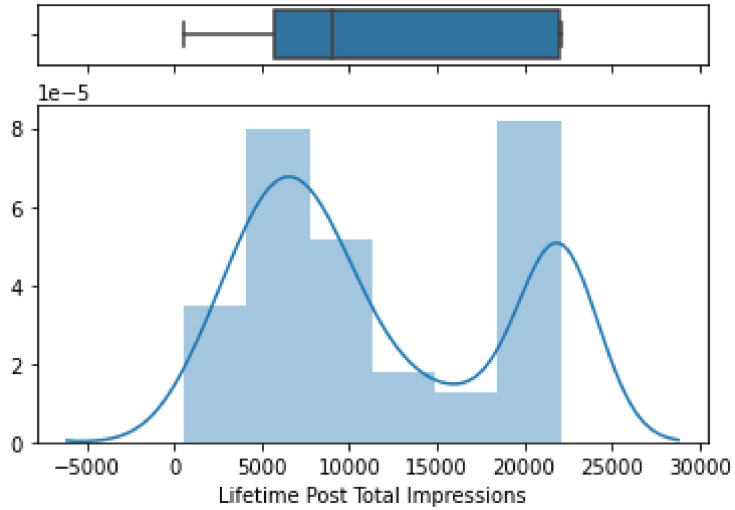


In [155]:
```
print(fb['Lifetime Post Total Impressions'].quantile(0.10))
print(fb['Lifetime Post Total Impressions'].quantile(0.75))
```

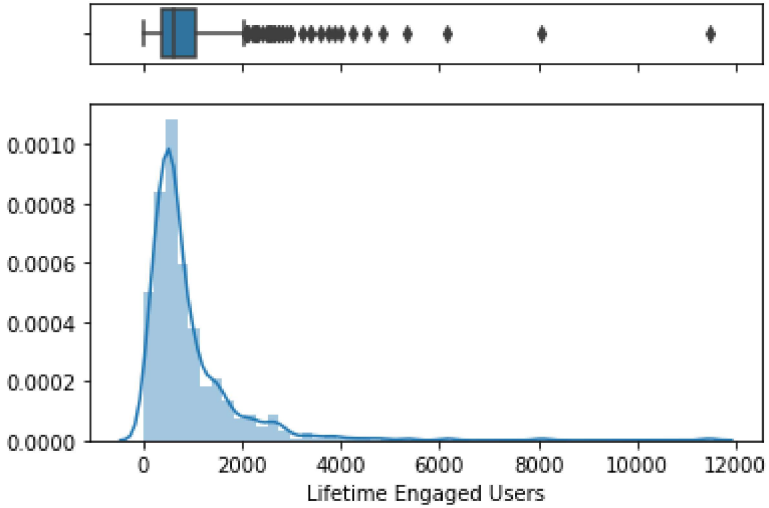
```
3650.6000000000004
22085.5
```

In [156]:
```
fb['Lifetime Post Total Impressions'] = np.where(fb['Lifetime Post Total Impressions'] >22085.5 ,22085.5,fb['Lifetime
 Post Total Impressions'])
```

In [157]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Total Impressions"], ax=ax_box)
sns.distplot(fb["Lifetime Post Total Impressions"], ax=ax_hist)


ax_box.set(xlabel='')
plt.savefig("Lifetime Post Total Impressions(1).png")
```



In [158]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Engaged Users"], ax=ax_box)
sns.distplot(fb["Lifetime Engaged Users"], ax=ax_hist)


ax_box.set(xlabel='')
plt.savefig("Lifetime Engaged Users.png")
```

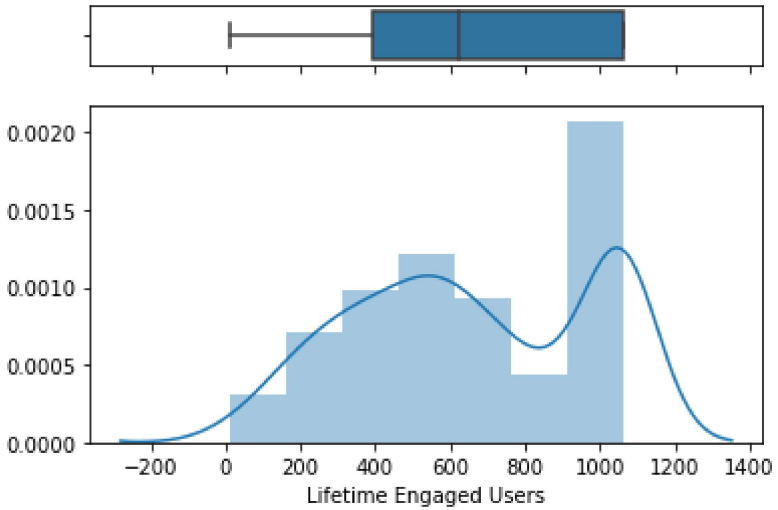


In [159]:
```
print(fb['Lifetime Engaged Users'].quantile(0.10))
print(fb['Lifetime Engaged Users'].quantile(0.75))
```

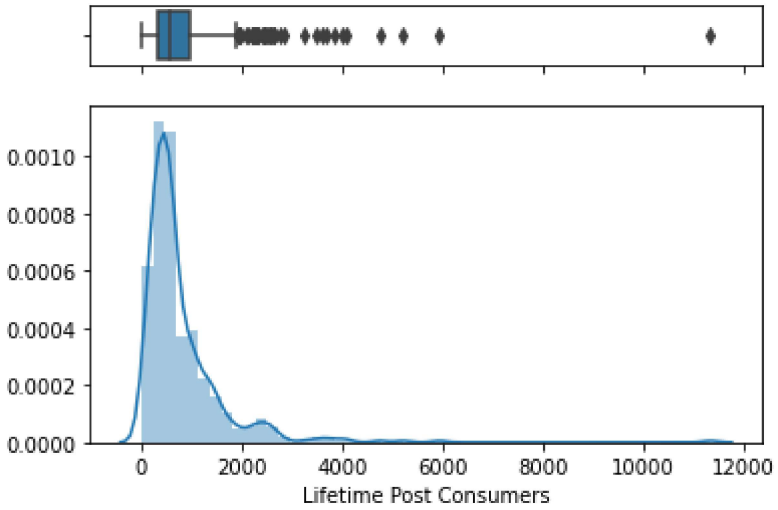
```
223.30000000000004
1062.0
```

In [160]:
```
fb['Lifetime Engaged Users'] = np.where(fb['Lifetime Engaged Users'] >1062 ,1062,fb['Lifetime Engaged Users'])
```

In [161]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["Lifetime Engaged Users"], ax=ax_box)
sns.distplot(fb["Lifetime Engaged Users"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Engaged Users(1).png")
```



In [162]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Consumers"], ax=ax_box)
sns.distplot(fb["Lifetime Post Consumers"], ax=ax_hist)


ax_box.set(xlabel='')
plt.savefig("Lifetime Post Consumers.png")
```

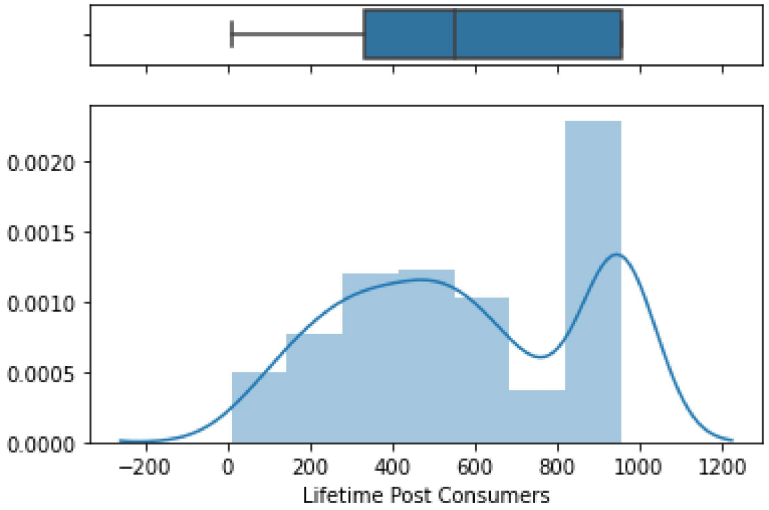


In [163]:
```
print(fb['Lifetime Post Consumers'].quantile(0.10))
print(fb['Lifetime Post Consumers'].quantile(0.75))
```

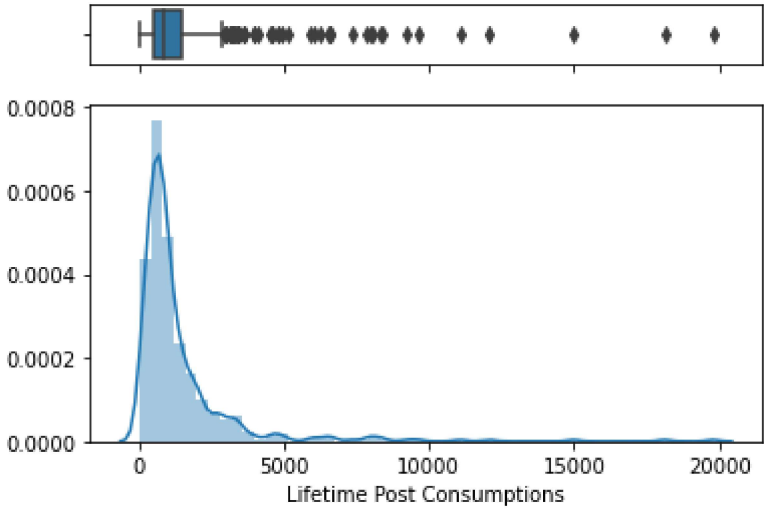
```
189.0
955.5
```

In [164]:
```
fb['Lifetime Post Consumers'] = np.where(fb['Lifetime Post Consumers'] >955.5 ,955.5,fb['Lifetime Post Consumers'])
```

In [165]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Consumers"], ax=ax_box)
sns.distplot(fb["Lifetime Post Consumers"], ax=ax_hist)


ax_box.set(xlabel='')
plt.savefig("Lifetime Post Consumers(1).png")
```



In [166]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Consumptions"], ax=ax_box)
sns.distplot(fb["Lifetime Post Consumptions"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Consumptions.png")
```

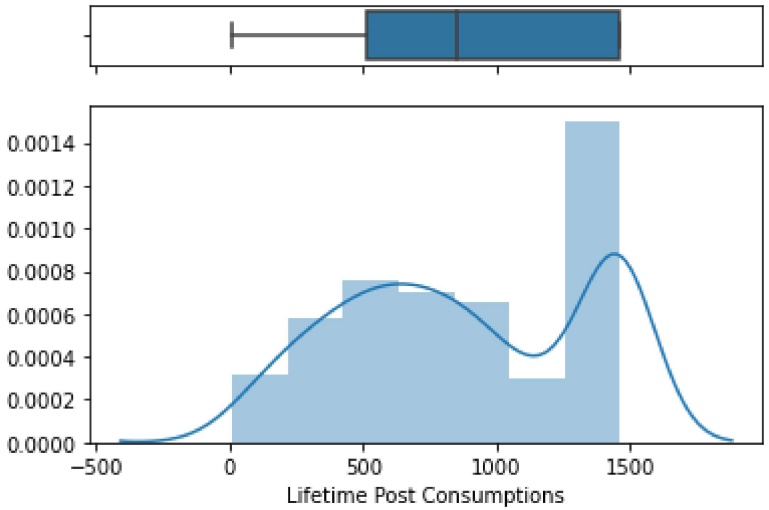


In [167]:
```
print(fb['Lifetime Post Consumptions'].quantile(0.10))
print(fb['Lifetime Post Consumptions'].quantile(0.75))
```

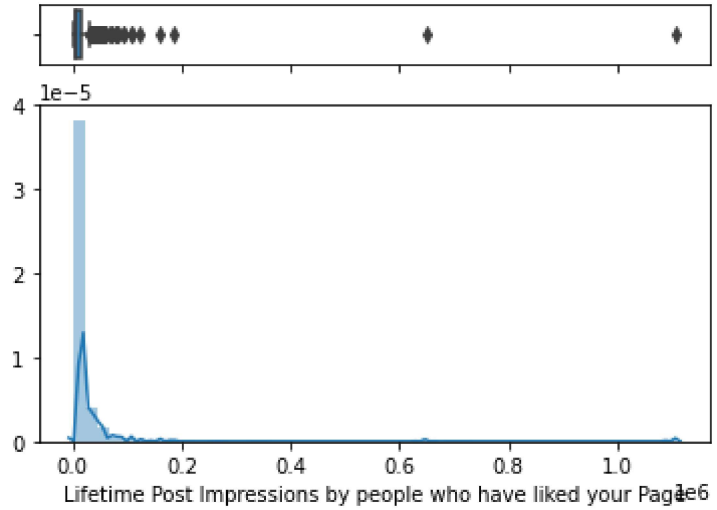
```
264.9
1463.0
```

In [168]:
```
fb['Lifetime Post Consumptions'] = np.where(fb['Lifetime Post Consumptions'] >1463 ,1463,fb['Lifetime Post Consumption
s'])
```

In [169]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Consumptions"], ax=ax_box)
sns.distplot(fb["Lifetime Post Consumptions"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Consumptions(1).png")
```



In [170]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Impressions by people who have liked your Page"], ax=ax_box)
sns.distplot(fb["Lifetime Post Impressions by people who have liked your Page"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Impressions by people who have liked your Page.png")
```

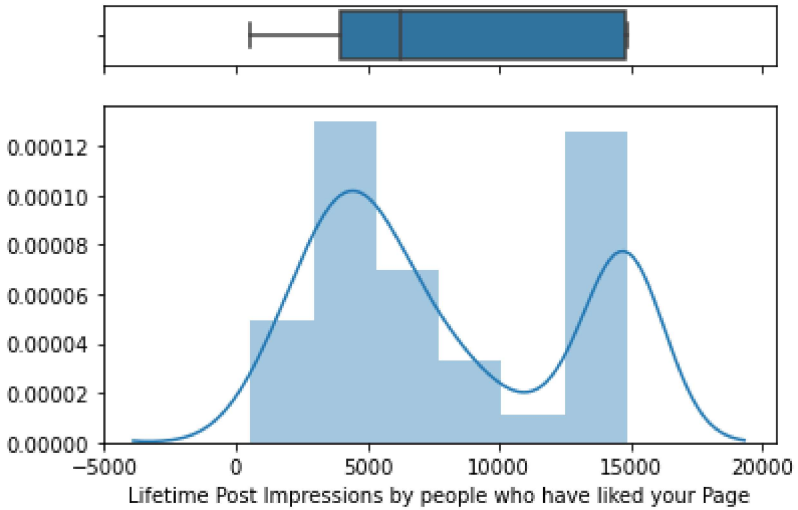


In [171]:
```
print(fb['Lifetime Post Impressions by people who have liked your Page'].quantile(0.10))
print(fb['Lifetime Post Impressions by people who have liked your Page'].quantile(0.75))
```

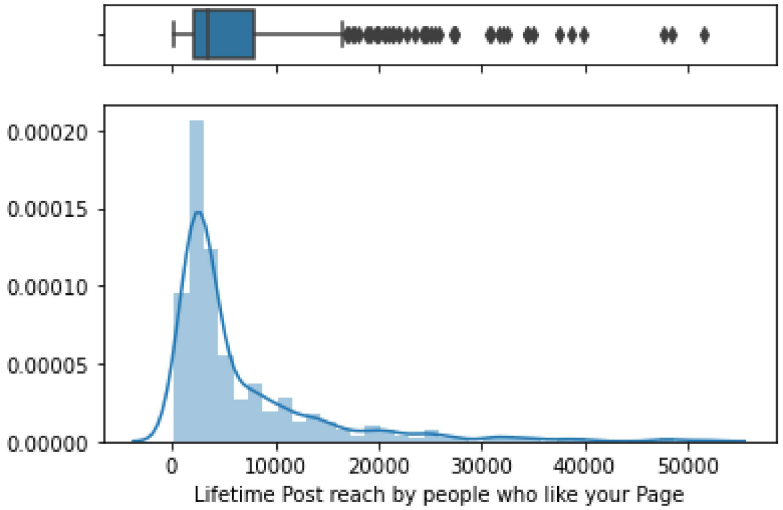
```
2568.1
14860.5
```

In [172]:
```
fb['Lifetime Post Impressions by people who have liked your Page'] = np.where(fb['Lifetime Post Impressions by people
 who have liked your Page'] >14860.5 ,14860.5,fb['Lifetime Post Impressions by people who have liked your Page'])
```

In [173]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post Impressions by people who have liked your Page"], ax=ax_box)
sns.distplot(fb["Lifetime Post Impressions by people who have liked your Page"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post Impressions by people who have liked your Page(1).png")
```



In [174]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post reach by people who like your Page"], ax=ax_box)
sns.distplot(fb["Lifetime Post reach by people who like your Page"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post reach by people who like your Page.png")
```

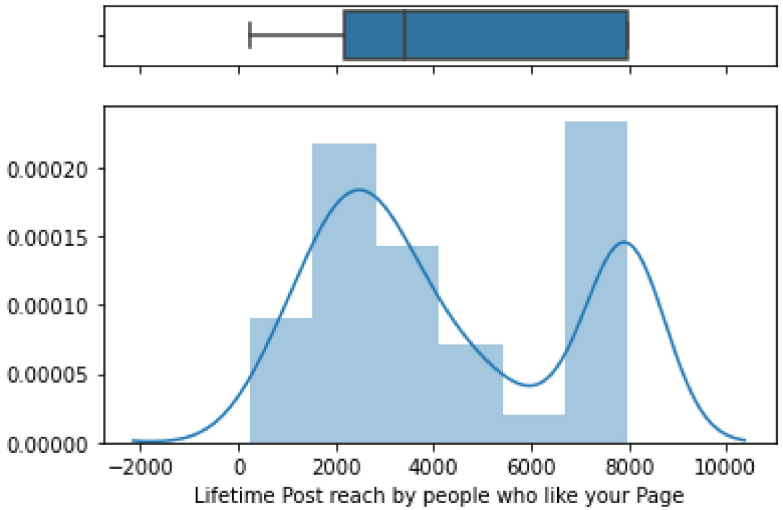


In [175]:
```
print(fb['Lifetime Post reach by people who like your Page'].quantile(0.10))
print(fb['Lifetime Post reach by people who like your Page'].quantile(0.75))
```

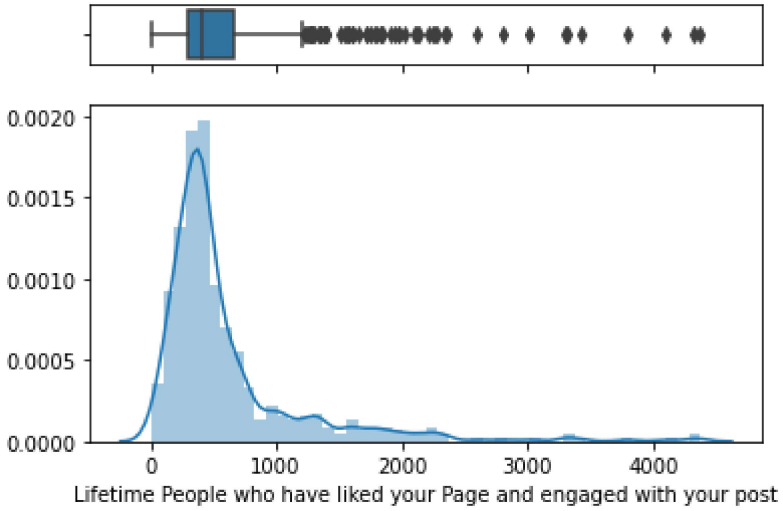
```
1442.2
7989.0
```

In [176]:
```
fb['Lifetime Post reach by people who like your Page'] = np.where(fb['Lifetime Post reach by people who like your Pag
e'] >7989 ,7989,fb['Lifetime Post reach by people who like your Page'])
```

In [177]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})


sns.boxplot(fb["Lifetime Post reach by people who like your Page"], ax=ax_box)
sns.distplot(fb["Lifetime Post reach by people who like your Page"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime Post reach by people who like your Page(1).png")
```



In [178]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["Lifetime People who have liked your Page and engaged with your post"], ax=ax_box)
sns.distplot(fb["Lifetime People who have liked your Page and engaged with your post"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime People who have liked your Page and engaged with your post.png")
```

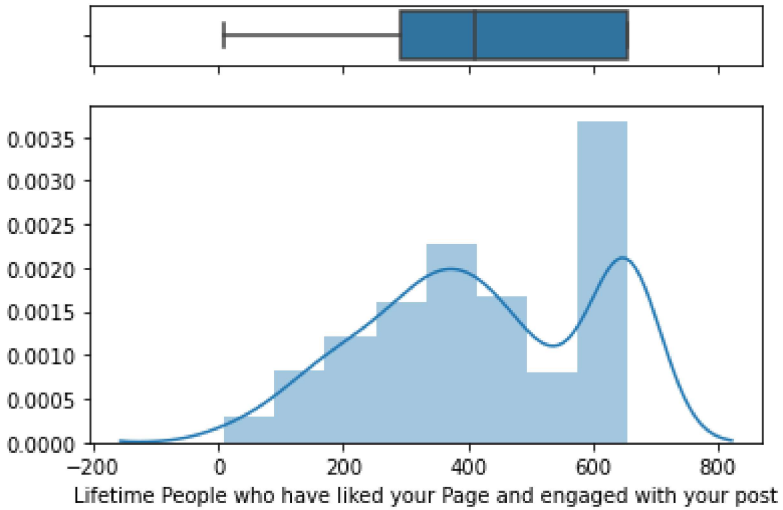


In [179]:
```
print(fb['Lifetime People who have liked your Page and engaged with your post'].quantile(0.10))
print(fb['Lifetime People who have liked your Page and engaged with your post'].quantile(0.75))
```

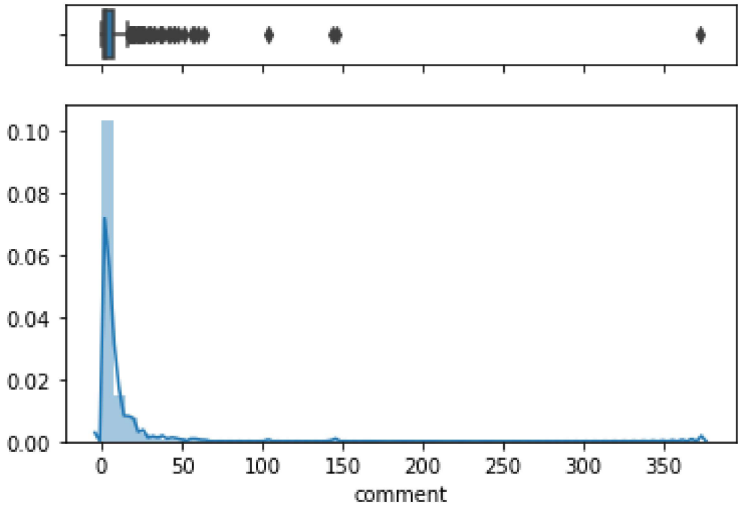
```
175.9
656.25
```

In [180]:
```
fb['Lifetime People who have liked your Page and engaged with your post'] = np.where(fb['Lifetime People who have like
d your Page and engaged with your post'] >656.25 ,656.25,fb['Lifetime People who have liked your Page and engaged with
your post'])
```

In [181]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["Lifetime People who have liked your Page and engaged with your post"], ax=ax_box)
sns.distplot(fb["Lifetime People who have liked your Page and engaged with your post"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("Lifetime People who have liked your Page and engaged with your post(1).png")
```



In [182]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["comment"], ax=ax_box)
sns.distplot(fb["comment"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("comment.png")
```

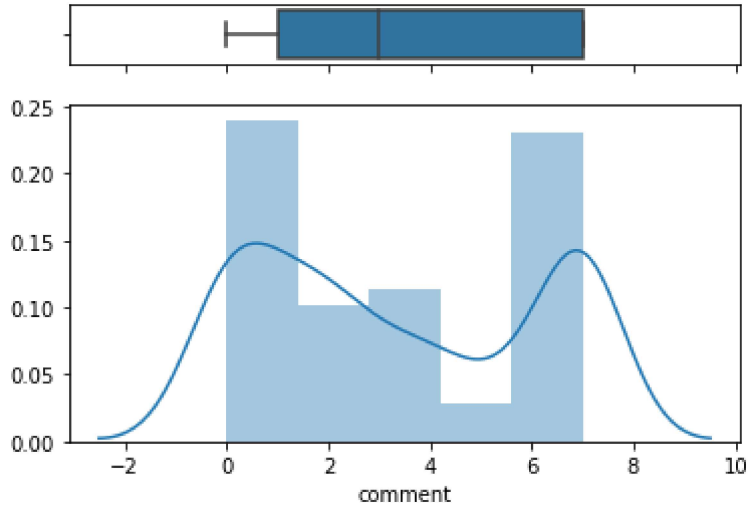


In [183]:
```
print(fb['comment'].quantile(0.10))
print(fb['comment'].quantile(0.75))
```

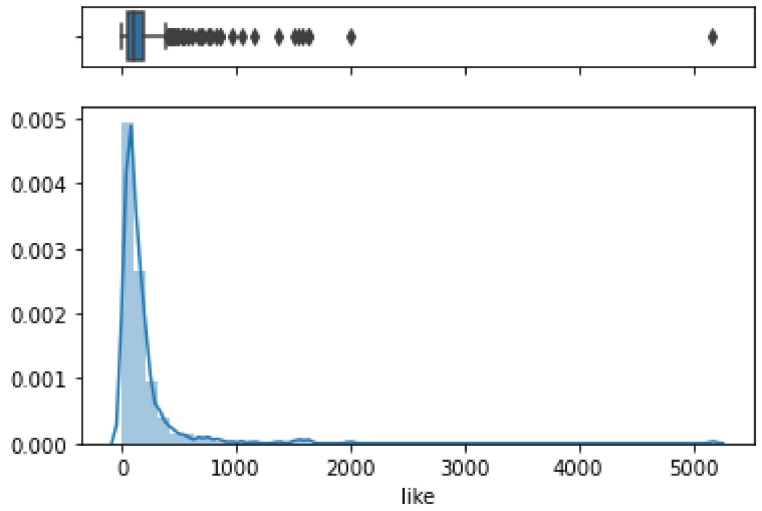
```
0.0
7.0
```

In [184]:
```
fb['comment'] = np.where(fb['comment'] >7 ,7,fb['comment'])
```

In [185]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["comment"], ax=ax_box)
sns.distplot(fb["comment"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("comment(1).png")
```



In [186]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})
sns.boxplot(fb["like"], ax=ax_box)
sns.distplot(fb["like"], ax=ax_hist)
ax_box.set(xlabel='')
plt.savefig("like.png")
```

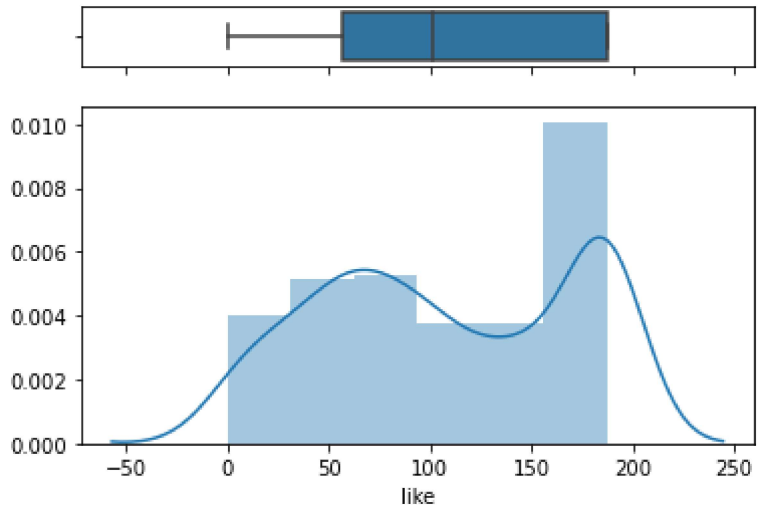


In [187]:
```
print(fb['like'].quantile(0.10))
print(fb['like'].quantile(0.75))
```

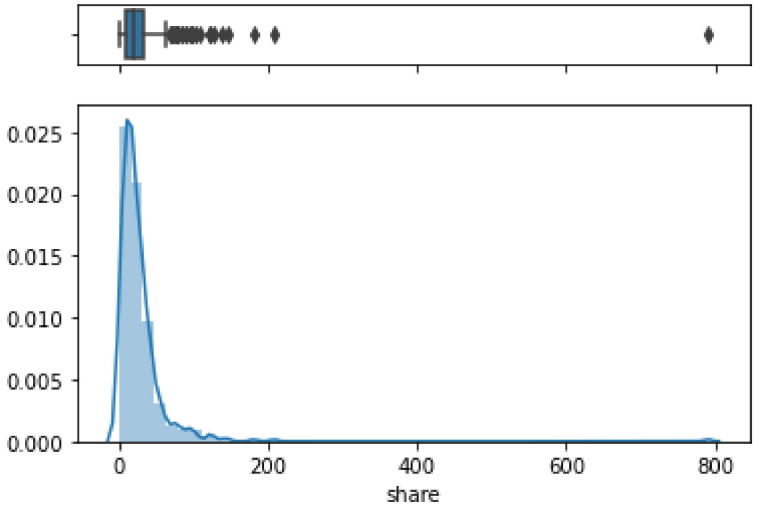
```
24.900000000000006
187.25
```

In [188]:
```
fb['like'] = np.where(fb['like'] >187.25 ,187.25,fb['like'])
```

In [189]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})
sns.boxplot(fb["like"], ax=ax_box)
sns.distplot(fb["like"], ax=ax_hist)
ax_box.set(xlabel='')
plt.savefig("like(1).png")
```



In [190]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["share"], ax=ax_box)
sns.distplot(fb["share"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("share.png")
```

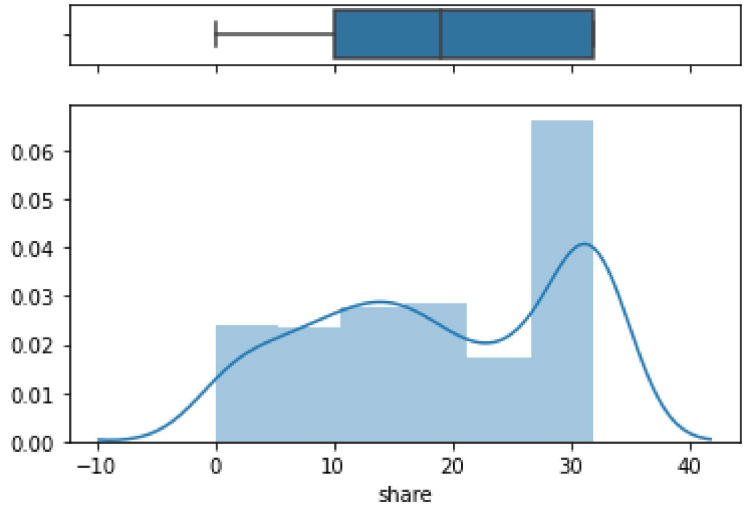


In [191]:
```
print(fb['share'].quantile(0.10))
print(fb['share'].quantile(0.75))
```

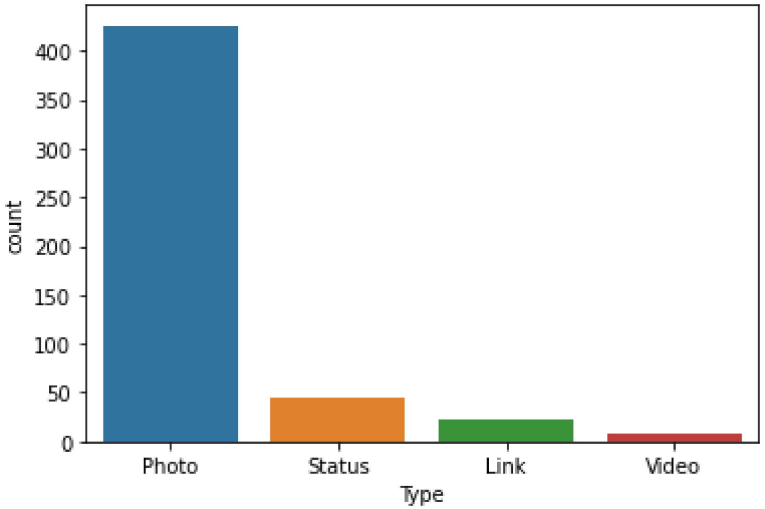
```
4.0
32.0
```

In [192]:
```
fb['share'] = np.where(fb['share'] >32 ,32,fb['share'])
```

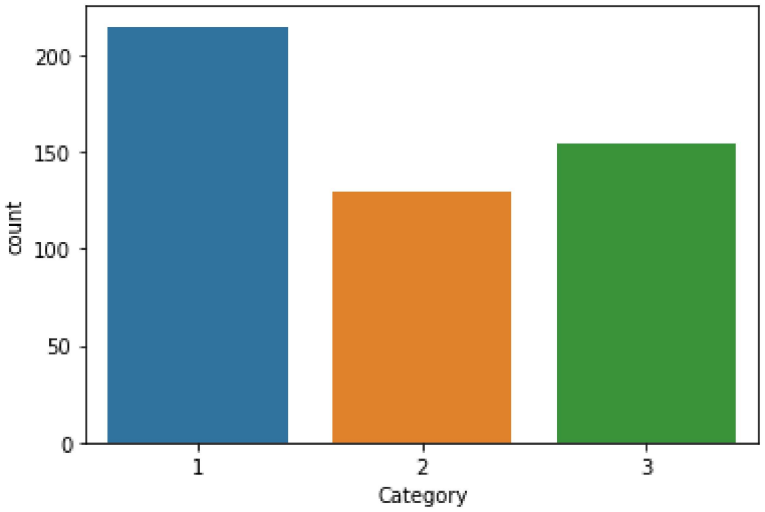
In [193]:
```
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.15, .85)})

sns.boxplot(fb["share"], ax=ax_box)
sns.distplot(fb["share"], ax=ax_hist)

ax_box.set(xlabel='')
plt.savefig("share(1).png")
```



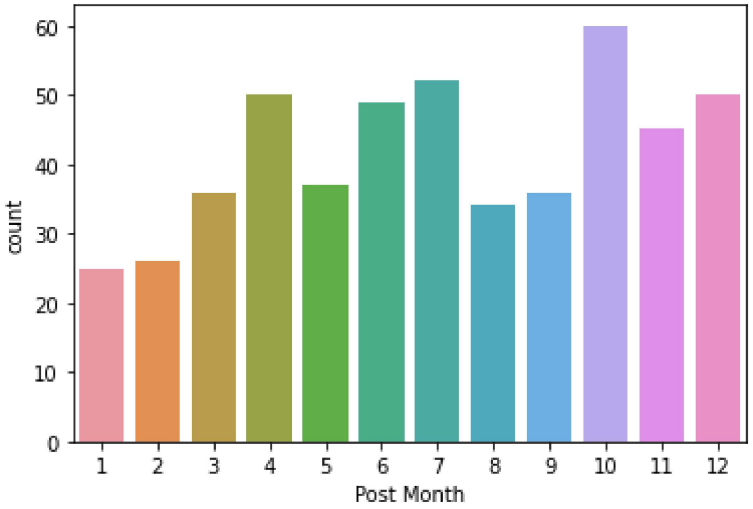
In [208]:
```
sns.countplot(fb["Type"])
plt.savefig("Type.png")
```



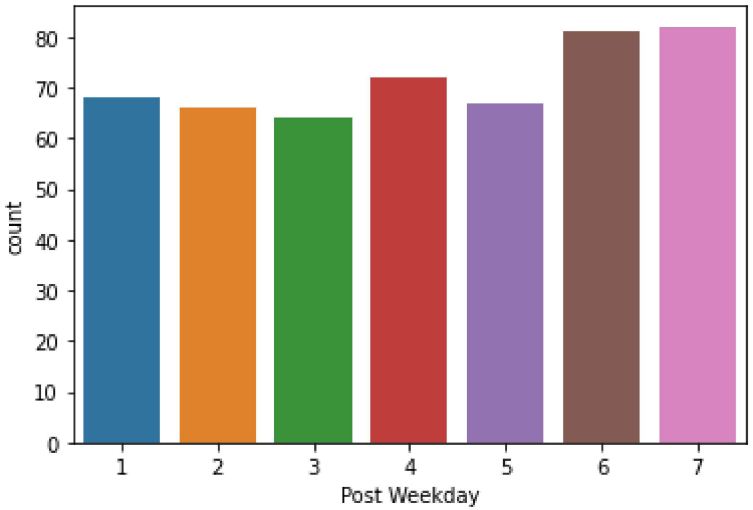
In [209]:
```
sns.countplot(fb["Category"])
plt.savefig("Category.png")
```



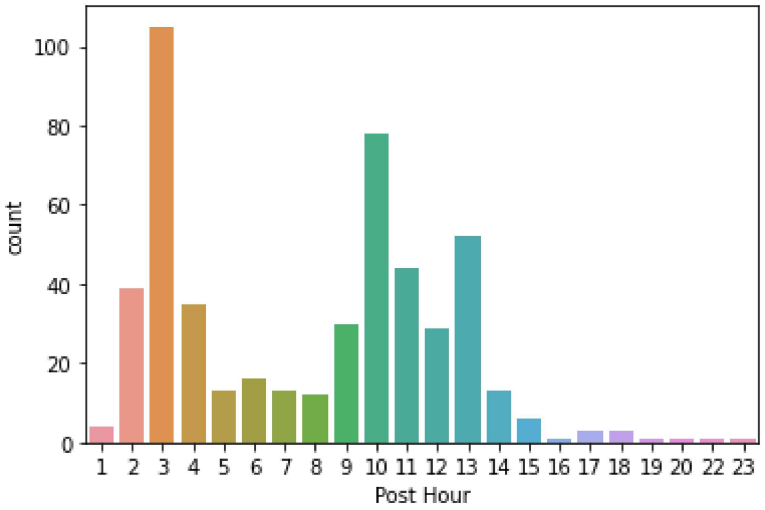
In [210]:
```
sns.countplot(fb["Post Month"])
plt.savefig("Post Month.png")
```



In [211]:
```
sns.countplot(fb["Post Weekday"])
plt.savefig("Post Weekday.png")
```

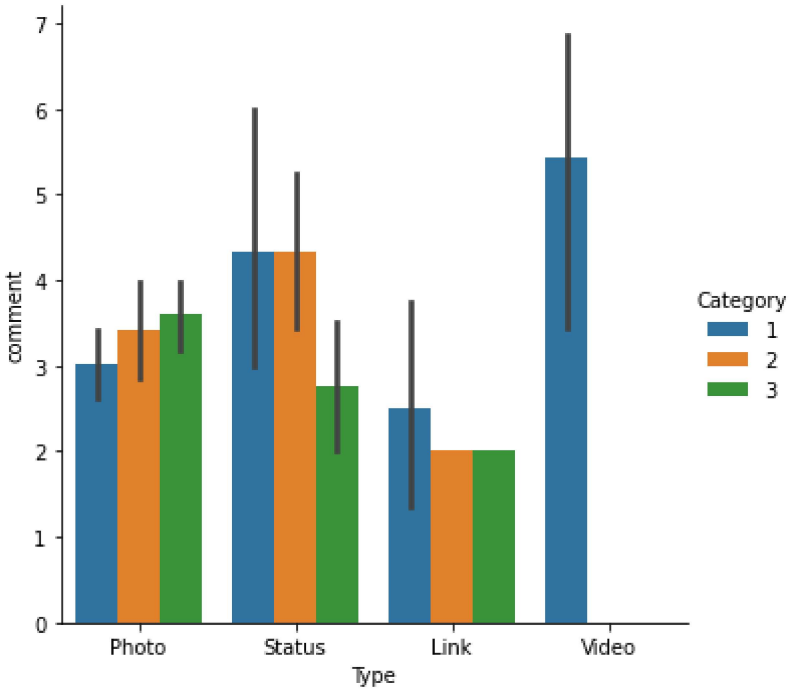


In [212]:
```
sns.countplot(fb["Post Hour"])
plt.savefig("Post Hour.png")
```

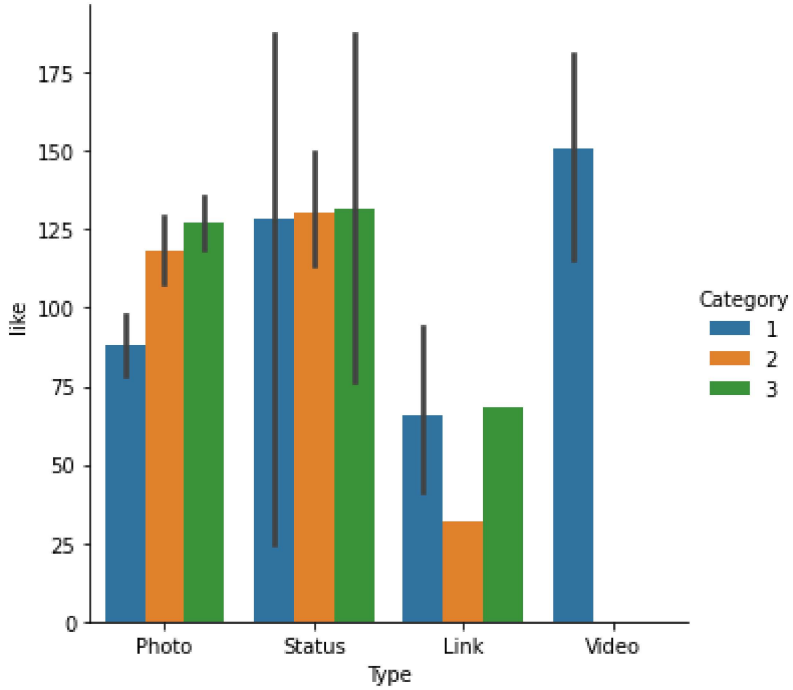


Bivariate analysis

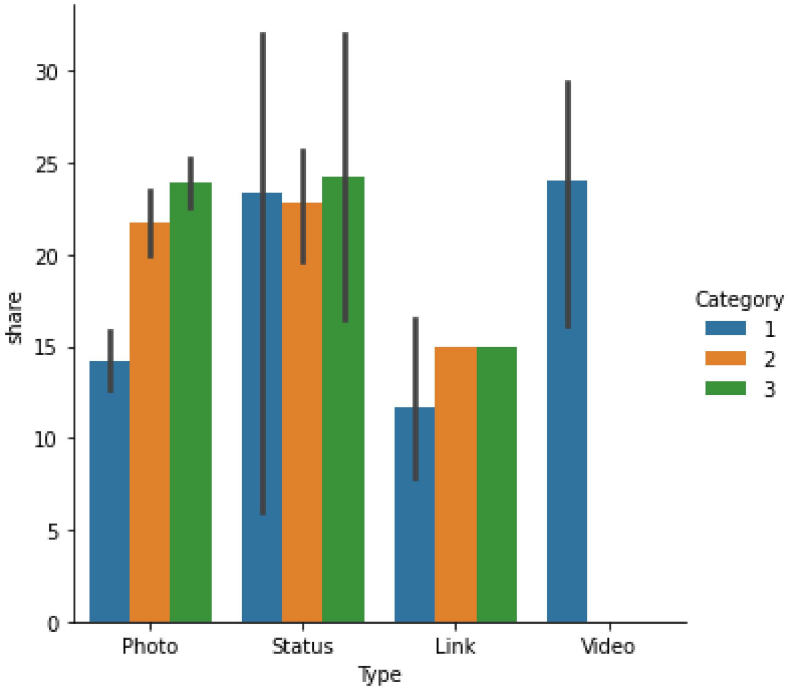
In [213]:
```
sns.catplot(x="Type", y="comment", hue="Category", kind="bar", data=fb)
plt.savefig("comment vs Typer.png")
```



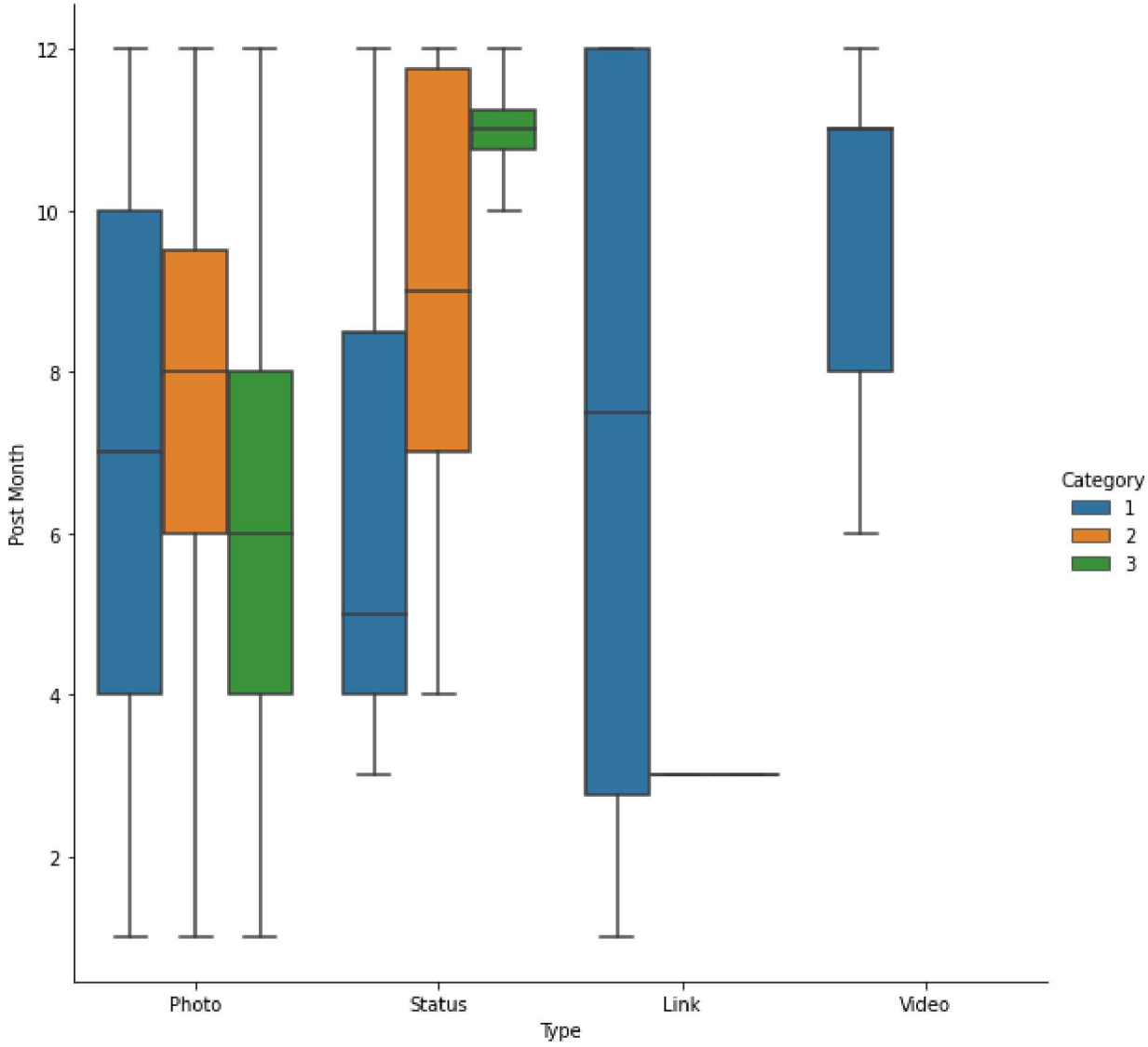
In [214]:
```
sns.catplot(x="Type", y="like", hue="Category", kind="bar", data=fb)
plt.savefig("like vs Type.png")
```



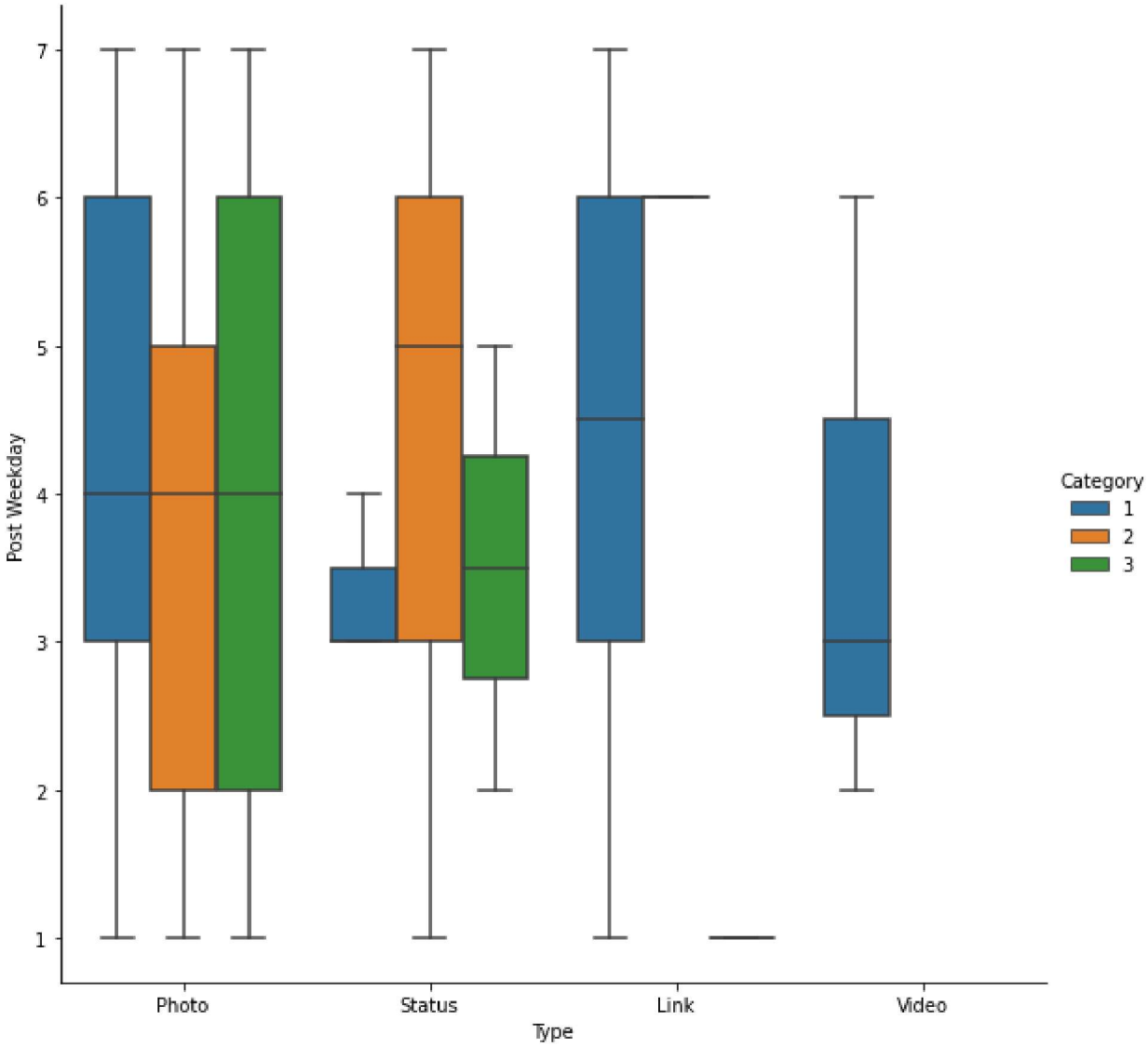
In [217]:
```
sns.catplot(x="Type", y="share", hue="Category", kind="bar", data=fb)
plt.savefig("Share vs Type.png")
```



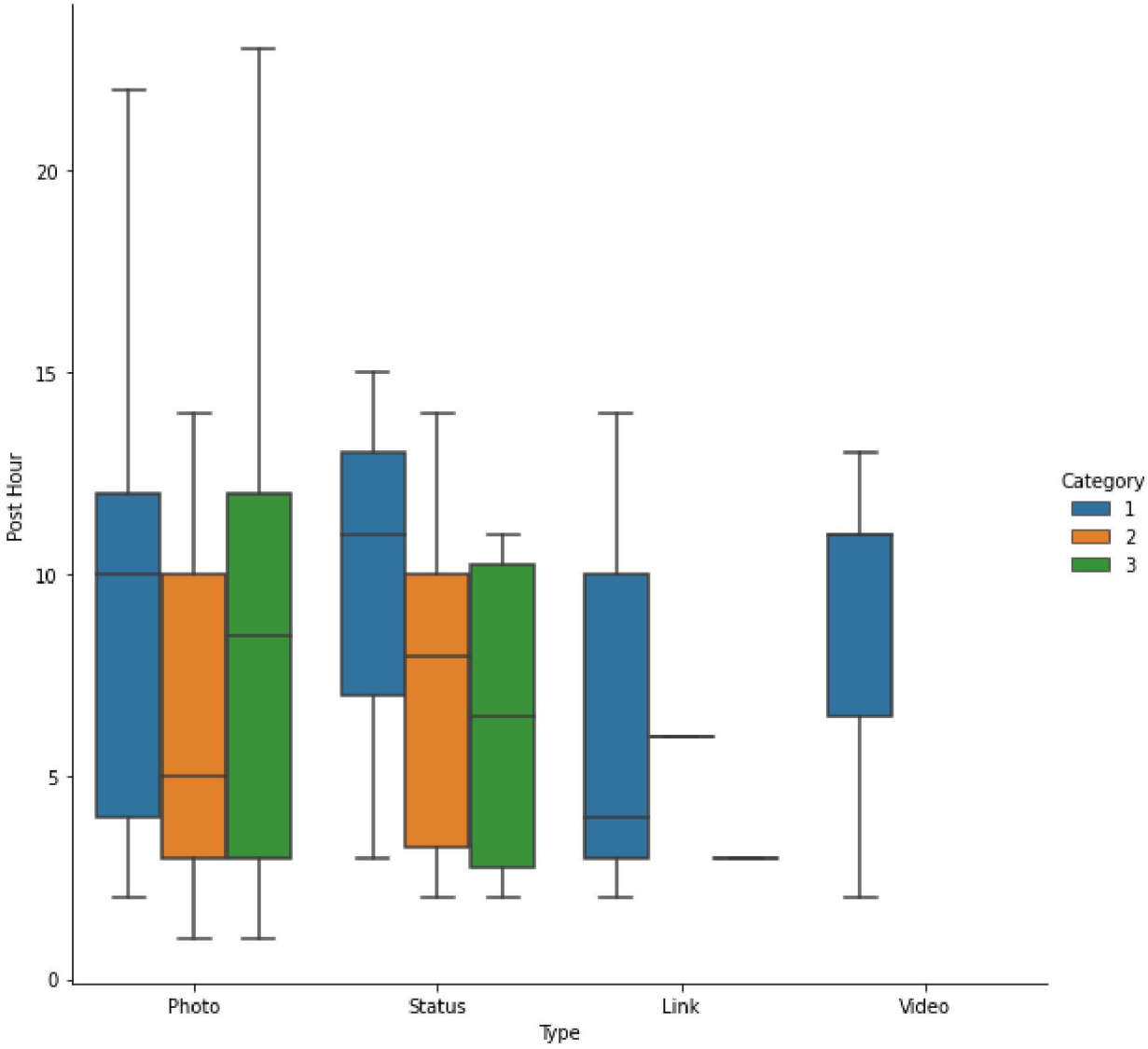
In [218]:
```
sns.catplot(x="Type", y="Post Month", hue="Category",
            kind="box", dodge=True, data=fb,height=8)
plt.savefig("post month vs type.png")
```



In [219]:
```
sns.catplot(x="Type", y="Post Weekday", hue="Category",
            kind="box", dodge=True, data=fb,height=8)
plt.savefig("post weekday vs Type.png")
```



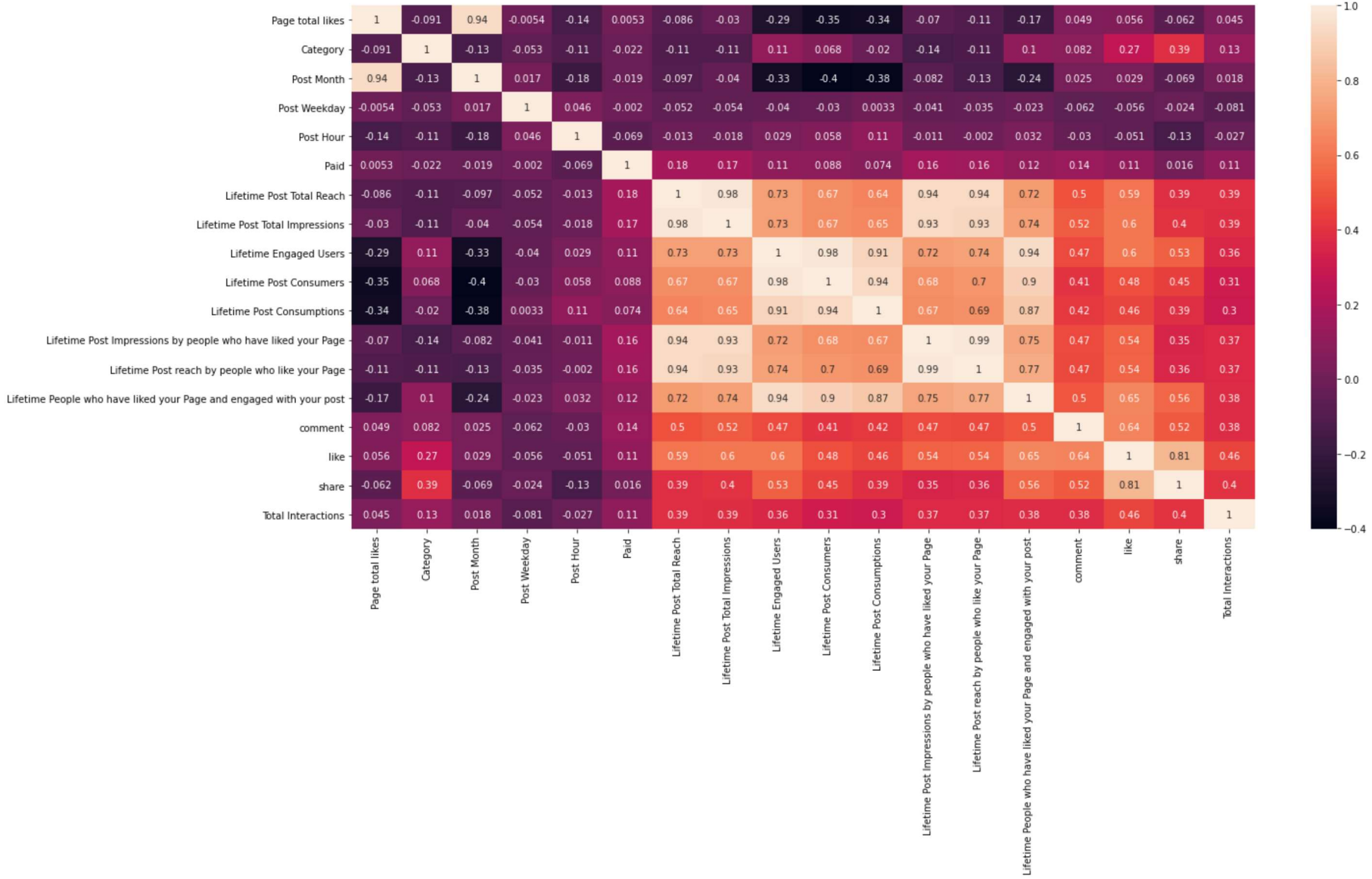
In [220]:
```
sns.catplot(x="Type", y="Post Hour", hue="Category",
            kind="box", dodge=True, data=fb,height=8)
plt.savefig("Post Hour vs type.png")
```



Mutlivariate analysis

In [221]:

```
conc=fb.corr()
plt.figure(figsize = (21,10))
sns.heatmap(conc,annot=True)
plt.savefig("heatmap.png")
```



Null hypothessis = The average likes of people who like a post on saturday is equal to people who like post on sunday.

Alternate hypothesis = The average likes of people who like a post on saturday is not equal to people who like post on sund
ay.

For the hypothesis testing we use paired t-test method.

In [206]:
```
a=fb[fb['Post Weekday']==6].like
b=fb[fb['Post Weekday']==7].like
a_mean=np.mean(a)
b_mean=np.mean(b)
print(a_mean,b_mean)
```

```
102.60118384917985 95.53048780487805
```

In [207]:
```
from scipy.stats import skew,ttest_ind
t_statistic, p_value = ttest_ind(a , b)

print("t statistic = ",t_statistic," p value = ", p_value)
if p_value<0.05:
    print("The p value of",p_value,"is significant. Hence reject null hypothesis")
else:
    print("The p value of",p_value,"is not significant. Hence accept null hypothesis")
```

```
t statistic =  0.6833989048166412  p value =  0.4953371233774665
The p value of 0.4953371233774665 is not significant. Hence accept null hypothesis
```

In [ ]: