In [9]:
```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```python
social=pd.read_csv("train.csv")
social
```

Out[3]:

	User_ID	Age	Gender	Platform	Daily_Usage_Time (minutes)	Posts_Per_Day	Likes_Received_Per_Day	Comments_Received_Per_Day	Messages_Sent_Per_Day	Dominant_Emotion
0	1	25	Female	Instagram	120.0	3.0	45.0	10.0	12.0	Happiness
1	2	30	Male	Twitter	90.0	5.0	20.0	25.0	30.0	Anger
2	3	22	Non-binary	Facebook	60.0	2.0	15.0	5.0	20.0	Neutral
3	4	28	Female	Instagram	200.0	8.0	100.0	30.0	50.0	Anxiety
4	5	33	Male	LinkedIn	45.0	1.0	5.0	2.0	10.0	Boredom
...	...	...	...	...	...	...	...	...	...	...
996	996	33	Non-binary	Twitter	85.0	4.0	35.0	18.0	18.0	Boredom
997	997	22	Female	Facebook	70.0	1.0	14.0	6.0	10.0	Neutral
998	998	35	Male	Whatsapp	110.0	3.0	50.0	25.0	25.0	Happiness
999	999	28	Non-binary	Telegram	60.0	2.0	18.0	8.0	18.0	Anger
1000	1000	27	Female	Snapchat	120.0	4.0	40.0	18.0	22.0	Neutral

1001 rows × 10 columns

In [5]:
```python
social.head()
```

Out[5]:

	User_ID	Age	Gender	Platform	Daily_Usage_Time (minutes)	Posts_Per_Day	Likes_Received_Per_Day	Comments_Received_Per_Day	Messages_Sent_Per_Day	Dominant_Emotion
0	1	25	Female	Instagram	120.0	3.0	45.0	10.0	12.0	Happiness
1	2	30	Male	Twitter	90.0	5.0	20.0	25.0	30.0	Anger
2	3	22	Non-binary	Facebook	60.0	2.0	15.0	5.0	20.0	Neutral
3	4	28	Female	Instagram	200.0	8.0	100.0	30.0	50.0	Anxiety
4	5	33	Male	LinkedIn	45.0	1.0	5.0	2.0	10.0	Boredom

In [6]:
```python
#1.DATA UNDERSTANDING
social.describe()
```

Out[6]:

	Daily_Usage_Time (minutes)	Posts_Per_Day	Likes_Received_Per_Day	Comments_Received_Per_Day	Messages_Sent_Per_Day
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000
mean	95.950000	3.321000	39.898000	15.611000	22.560000
std	38.850442	1.914582	26.393867	8.819493	8.516274
min	40.000000	1.000000	5.000000	2.000000	8.000000
25%	65.000000	2.000000	20.000000	8.000000	17.750000
50%	85.000000	3.000000	33.000000	14.000000	22.000000
75%	120.000000	4.000000	55.000000	22.000000	28.000000
max	200.000000	8.000000	110.000000	40.000000	50.000000

In [7]:
```python
#2.INTIAL CHECK UP
social.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1001 entries, 0 to 1000
Data columns (total 10 columns):
 #   Column                      Non-Null Count  Dtype  
---  ------                      --------------  -----  
 0   User_ID                     1001 non-null   object 
 1   Age                         1001 non-null   object 
 2   Gender                      1000 non-null   object 
 3   Platform                    1000 non-null   object 
 4   Daily_Usage_Time (minutes)  1000 non-null   float64
 5   Posts_Per_Day               1000 non-null   float64
 6   Likes_Received_Per_Day      1000 non-null   float64
 7   Comments_Received_Per_Day   1000 non-null   float64
 8   Messages_Sent_Per_Day       1000 non-null   float64
 9   Dominant_Emotion            1000 non-null   object 
dtypes: float64(5), object(5)
memory usage: 78.3+ KB
```

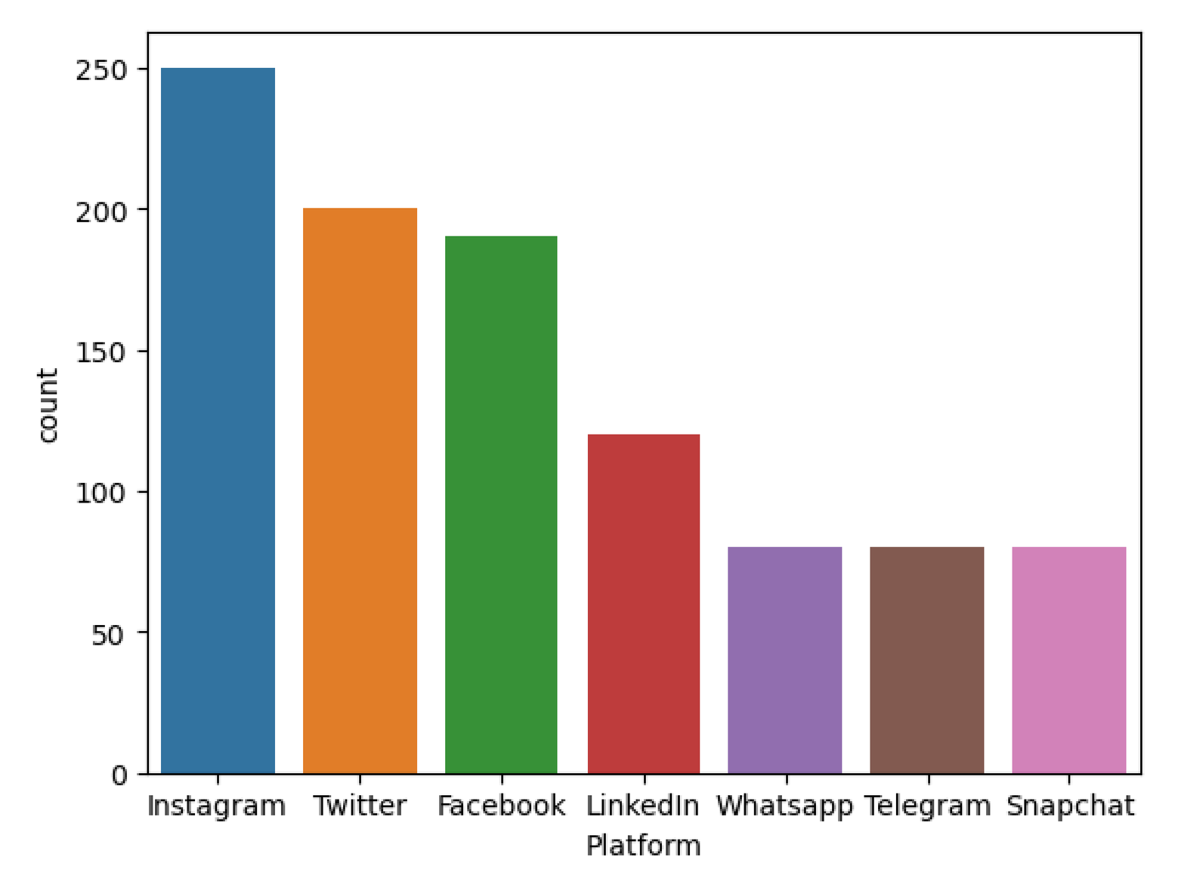
In [8]:
```python
#ASKING QUESTINS
###1.which platform used high
###2.highest posts per day
###3.platoform wise daily usage
###4.age wise post per day
###5.Age wise dominant emotion
###6.age wise likes per day
###7.platform wise message sent per day
###8.gender ,age wise comments received per day
```

In [10]:
```python
#UNIVARIENT ANALYSIS
###1.which platform used high
social["Platform"].value_counts(normalize=True)*100
```

Out[10]:
```
Platform
Instagram    25.0
Twitter      20.0
Facebook     19.0
LinkedIn     12.0
Whatsapp      8.0
Telegram      8.0
Snapchat      8.0
Name: proportion, dtype: float64
```

In [11]:
```python
sns.countplot(social,x='Platform')
```

Out[11]: <Axes: xlabel='Platform', ylabel='count'>



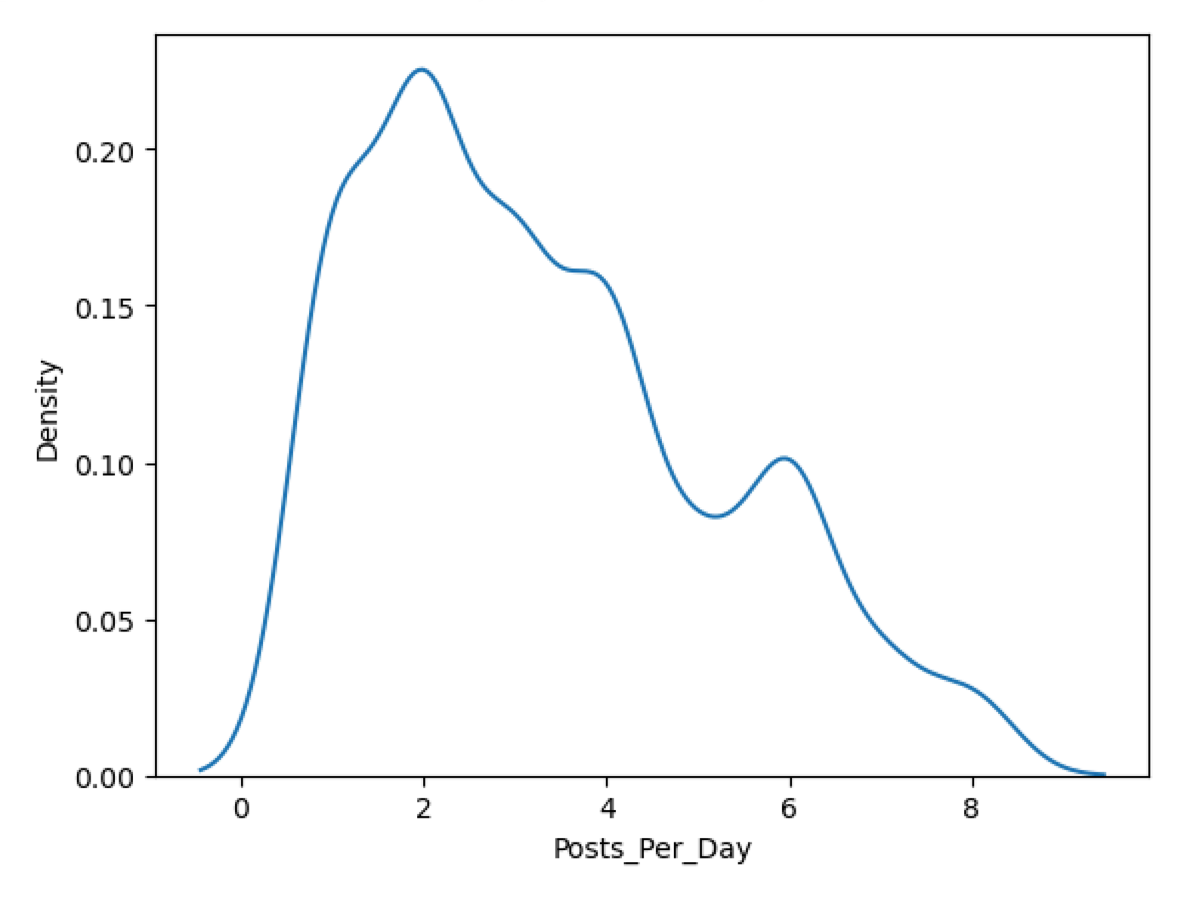
In [12]:
```python
###2.highest meassages per day
social["Posts_Per_Day"].value_counts(normalize=True)*100
```

Out[12]:
```
Posts_Per_Day
2.0    23.0
1.0    19.0
3.0    17.1
4.0    16.1
6.0    10.9
5.0     7.1
7.0     3.9
8.0     2.9
Name: proportion, dtype: float64
```

In [13]:
```python
sns.kdeplot(social,x='Posts_Per_Day')
```

```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

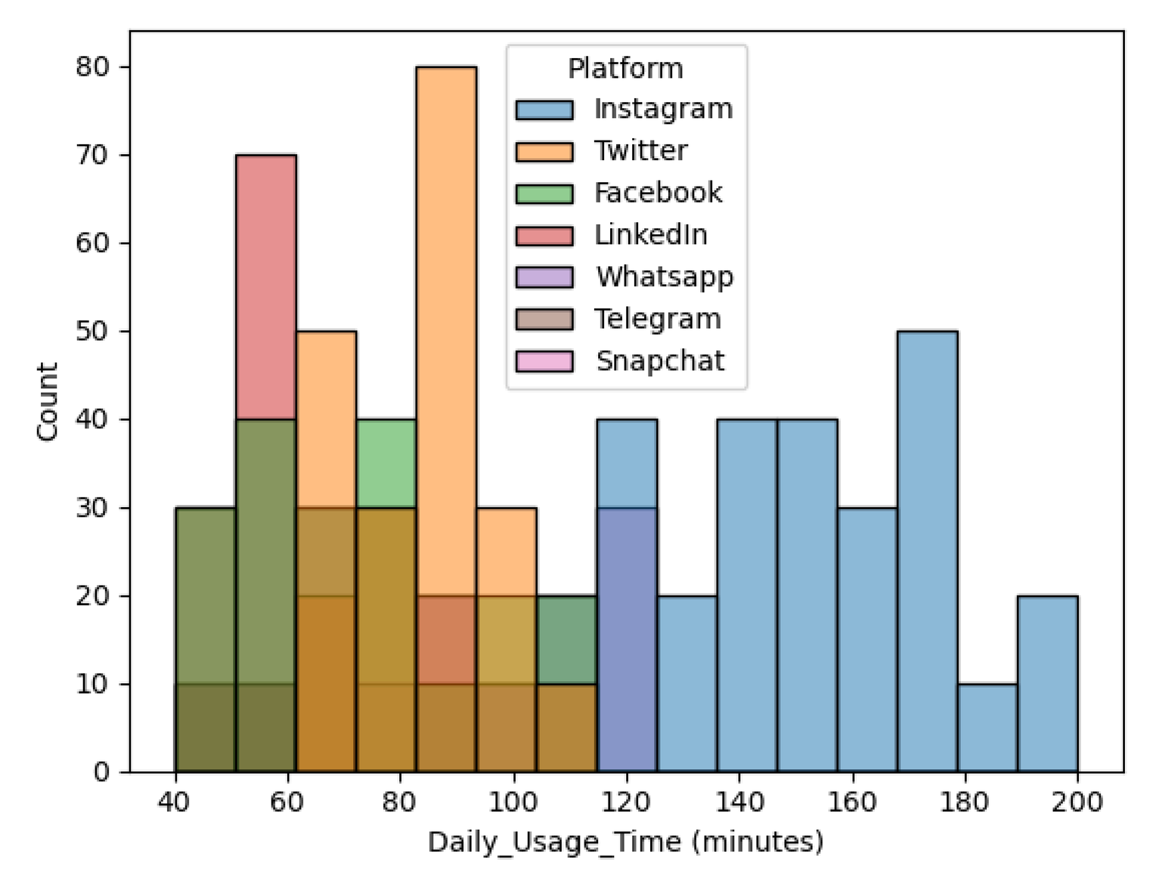
Out[13]: <Axes: xlabel='Posts_Per_Day', ylabel='Density'>



In [14]:
```python
#BIVARIENT ANALYSIS
###3.platoform wise daily usage
sns.histplot(social,x='Daily_Usage_Time (minutes)',hue='Platform')
```

```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

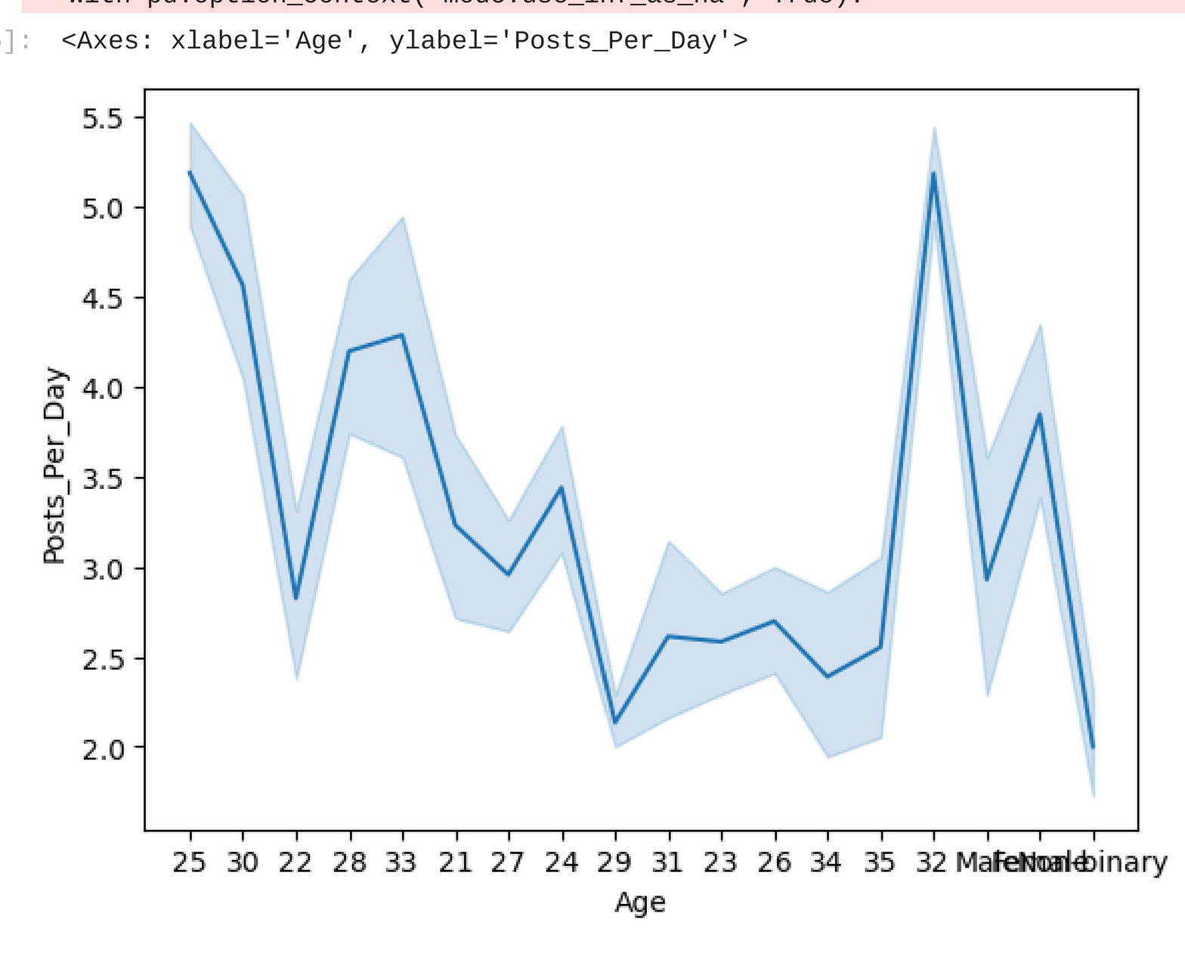
Out[14]: <Axes: xlabel='Daily_Usage_Time (minutes)', ylabel='Count'>



In [25]:
```python
###4.age wise posts per day
sns.lineplot(social,x='Age',y='Posts_Per_Day')
```

```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

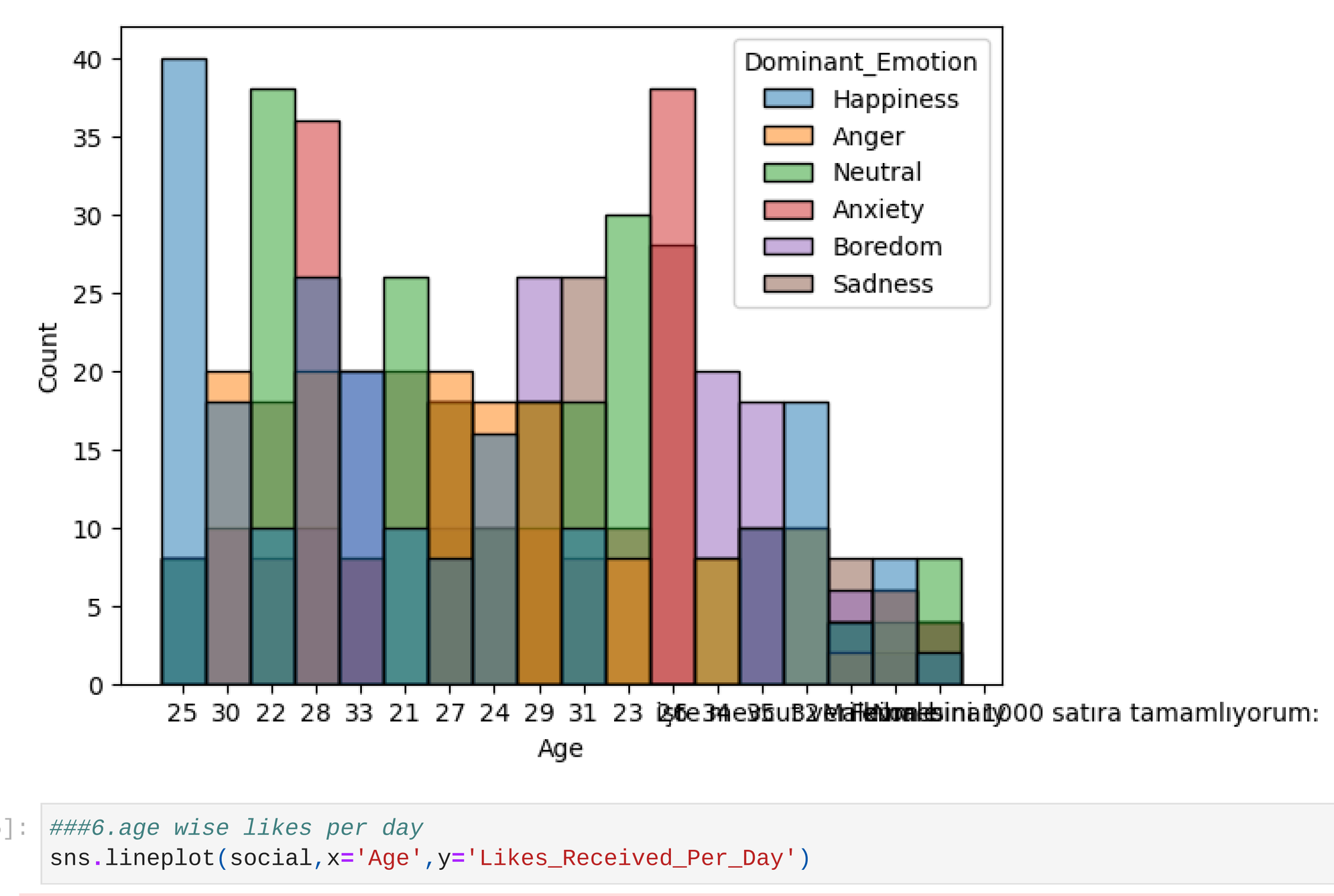
Out[25]: <Axes: xlabel='Age', ylabel='Posts_Per_Day'>



In [34]:
```python
###5.Age wise dominant emotion
sns.histplot(social,x='Age',hue='Dominant_Emotion')
```

```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

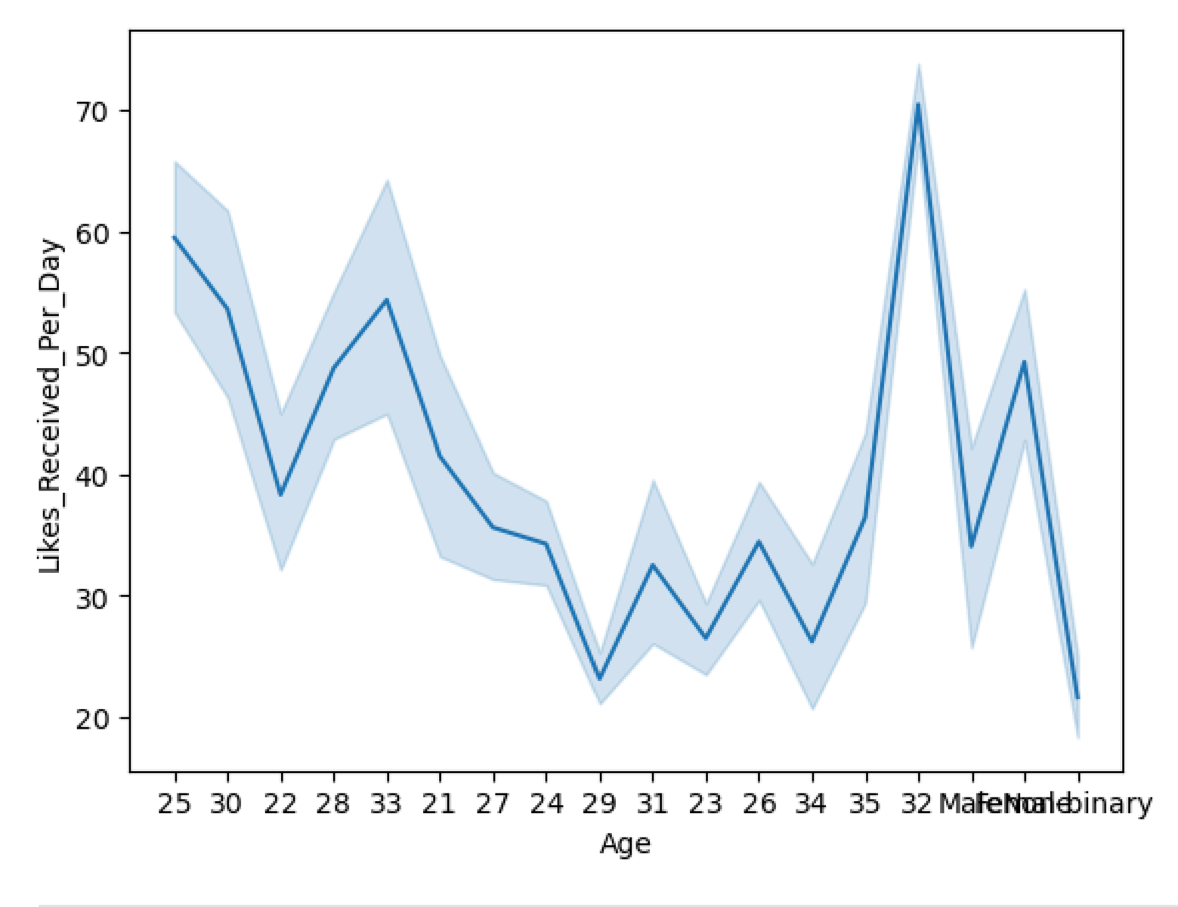
Out[34]: <Axes: xlabel='Age', ylabel='Count'>



In [35]:
```python
###6.age wise likes per day
sns.lineplot(social,x='Age',y='Likes_Received_Per_Day')
```

```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

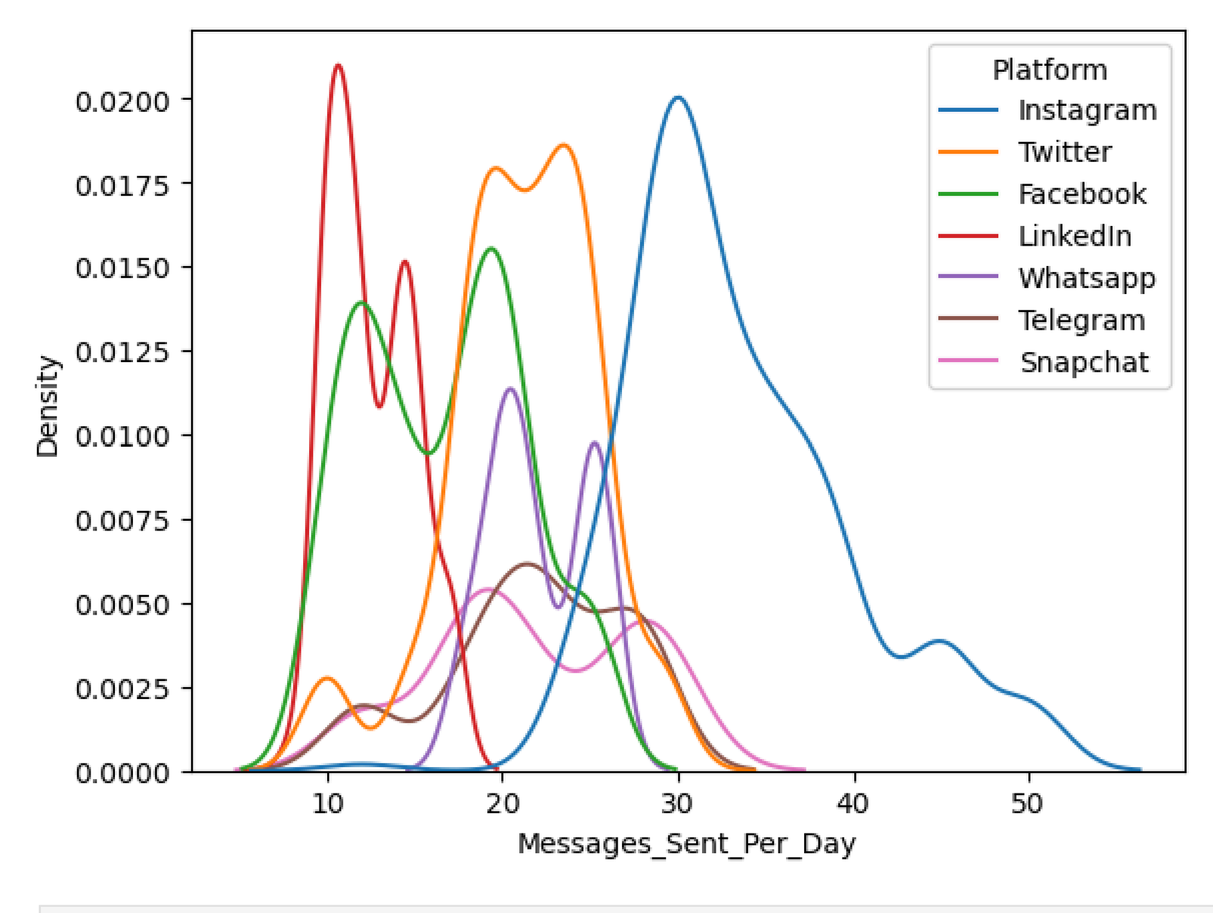
Out[35]: <Axes: xlabel='Age', ylabel='Likes_Received_Per_Day'>



In [42]:
```python
###7.platform wise message sent per day
sns.kdeplot(social,x='Messages_Sent_Per_Day',hue='Platform')
```

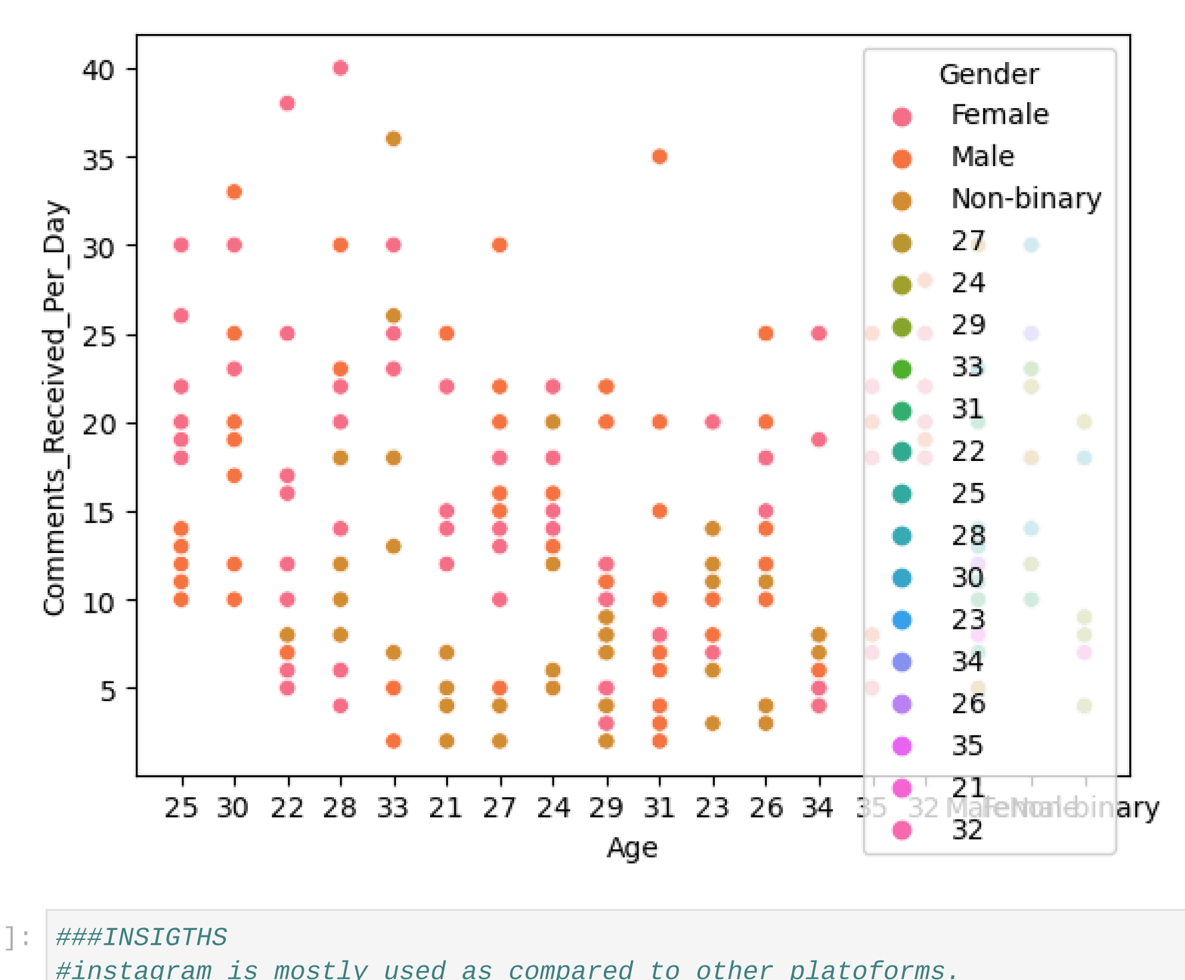
```
C:\Users\ajaykanth\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

Out[42]: <Axes: xlabel='Messages_Sent_Per_Day', ylabel='Density'>



In [44]:
```python
#MULTI VARIENT ANALYSIS
###8.gender ,age wise comments received per day
sns.scatterplot(social,x='Age',y='Comments_Received_Per_Day',hue='Gender')
```

Out[44]: <Axes: xlabel='Age', ylabel='Comments_Received_Per_Day'>



In [ ]:
```python
###INSIGTHS
#instagram is mostly used as compared to other platoforms.
#whatsapp,snapchat and telegram are used with same 8%.
# 2 posts liked per day percentage is 23.
#Atleast only 2% of 8 posts liked per day.
# Twitter platform is used 100 mins a day.
# Linkedin is used 60 mins per day with average 70 people.
# Snapchat is used between 120 to 400 mins per day.
# Snapchat is used by most of time.
# people age of 22 and 35 higly posts per day.
# happiness is the high Dominant emotion of age 25a
# Anxiety is more in people whose age is 22.
# most of mesages sent per day in the platform of instagram.
# female of age 28 receives 40 comments per day.
```