In [3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as pyplot
```

In [4]:
```
df= pd.read_csv("/content/Customer_support.csv")
```

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

Out[ ]:

	Ticket ID	Customer Name	Customer Email	Customer Age	Customer Gender	Produc Purchase
0	1	Marisa Obrien	carrollallison@example.com	32	Other	GoPro Her
1	2	Jessica Rios	clarkeashley@example.com	42	Female	LG Sma T
2	3	Christopher Robbins	gonzalestracy@example.com	48	Other	Dell XP
3	4	Christina Dillon	bradleyolson@example.org	27	Female	Microso Offic
4	5	Alexander Carroll	bradleymark@example.com	67	Female	Autodes AutoCA

In [ ]:
```
df.tail()
```

Out[ ]:

	Ticket ID	Customer Name	Customer Email	Customer Age	Customer Gender	Product Purchased
8464	8465	David Todd	adam28@example.net	22	Female	LG OLED
8465	8466	Lori Davis	russell68@example.com	27	Female	Bose SoundLink Speaker
8466	8467	Michelle Kelley	ashley83@example.org	57	Female	GoPro Action Camera
8467	8468	Steven Rodriguez	fpowell@example.org	54	Male	PlayStation
8468	8469	Steven Davis MD	lori20@example.net	53	Other	Philips Hue Lights

In [ ]:
```
df.shape[0]
```

Out[ ]:
```
8469
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8469 entries, 0 to 8468
Data columns (total 13 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Ticket ID                     8469 non-null   int64
 1   Customer Name                 8469 non-null   object
 2   Customer Email                8469 non-null   object
 3   Customer Age                  8469 non-null   int64
 4   Customer Gender               8469 non-null   object
 5   Product Purchased             8469 non-null   object
 6   Date of Purchase              8469 non-null   object
 7   Ticket Type                   8469 non-null   object
 8   Ticket Subject                8469 non-null   object
 9   Ticket Status                 8469 non-null   object
 10  Ticket Priority               8469 non-null   object
 11  Ticket Channel                8469 non-null   object
 12  Customer Satisfaction Rating  8469 non-null   object
dtypes: int64(2), object(11)
memory usage: 860.3+ KB
```

In [ ]:
```
df.groupby('Customer Gender' )['Customer Age'].mean()
```

Out[ ]:

Customer Gender	Customer Age
Female	43.778317
Male	43.975829
Other	44.348846

dtype: float64

In [ ]:
```
df.loc[df['Customer Satisfaction Rating'] == '5', 'Customer Email']
```

Out[ ]:

	Customer Email
19	jameslopez@example.com
28	garciastacy@example.com
33	darlenelee@example.org
58	lwilliamson@example.net
66	gillespiegeorge@example.com
...	...
8388	brittanycaldwell@example.org
8391	lisa70@example.com
8420	coxbrandon@example.org
8450	hhunt@example.com
8453	wanderson@example.com

544 rows × 1 columns

dtype: object

In [ ]:
```
df.loc[df['Customer Satisfaction Rating'] == '5', 'Customer Email'].shape[0]
```

Out[ ]: 544

In [ ]:
```
df.loc[df['Ticket Priority'] == 'High', ['Customer Name', 'Product Purchased',
```

Out[ ]:

	Customer Name	Product Purchased	Ticket Type	Ticket Priority
10	Joseph Moreno	Nintendo Switch	Cancellation request	High
11	Brandon Arnold	Microsoft Xbox Controller	Product inquiry	High
14	Amy Hill	Sony PlayStation	Billing inquiry	High
15	Elizabeth Foley	GoPro Action Camera	Billing inquiry	High
17	Joshua Castillo	Microsoft Xbox Controller	Product inquiry	High
...	...	...	...	...
8457	Amy Goodman	Fitbit Charge	Product inquiry	High
8459	Shelly Estrada	Canon DSLR Camera	Product inquiry	High
8460	Trevor Perry	Apple AirPods	Billing inquiry	High
8466	Michelle Kelley	GoPro Action Camera	Technical issue	High
8468	Steven Davis MD	Philips Hue Lights	Billing inquiry	High

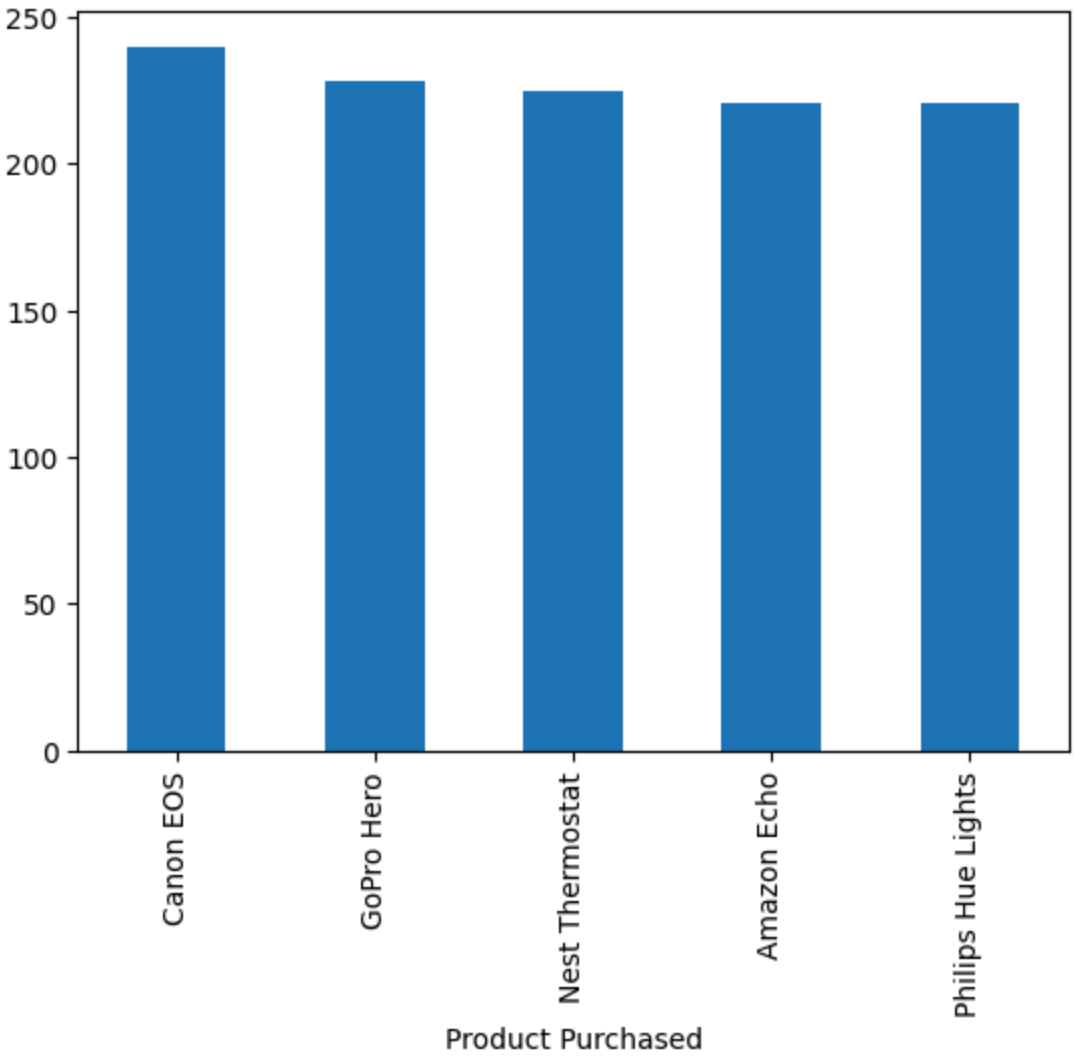
2085 rows × 4 columns

In [ ]:
```
df.loc[df['Ticket Priority'] == 'High', ['Customer Name', 'Product Purchased',
```

Out[ ]:
```
2085
```

In [ ]:
```
df['Product Purchased'].value_counts().head(5).plot(kind="bar")
```

Out[ ]:
```
<Axes: xlabel='Product Purchased'>
```



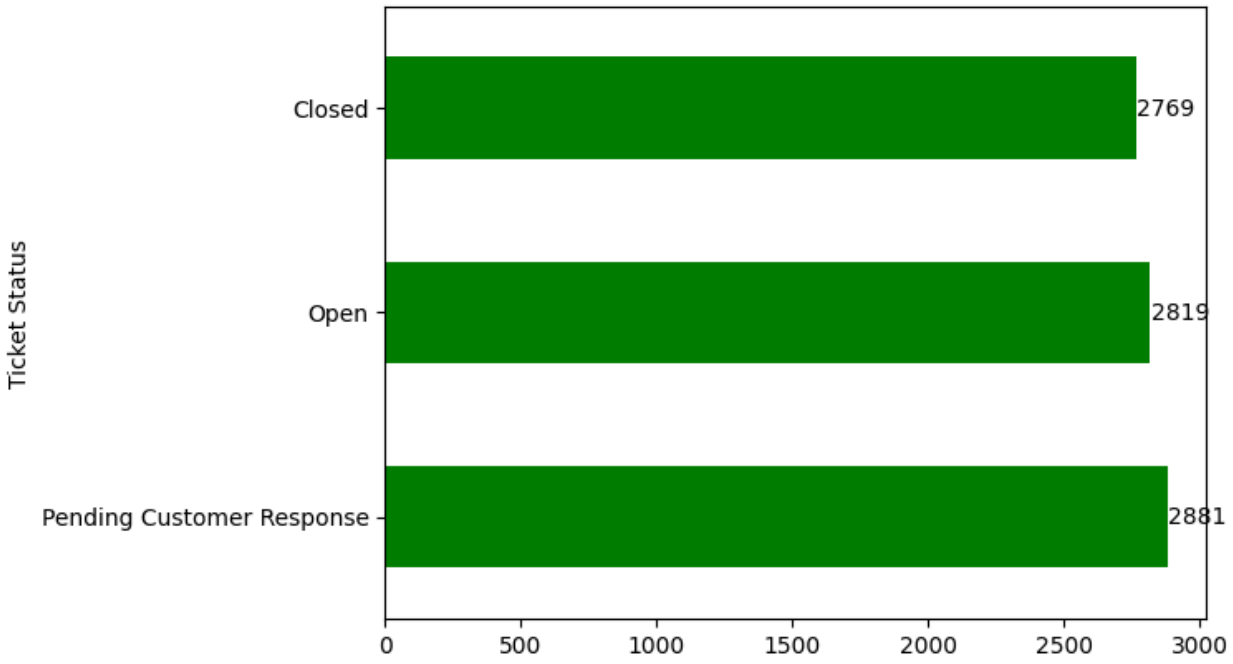
In [ ]:
```
df.groupby('Ticket Status')[ 'Ticket Type'].count()
```

Out[ ]:

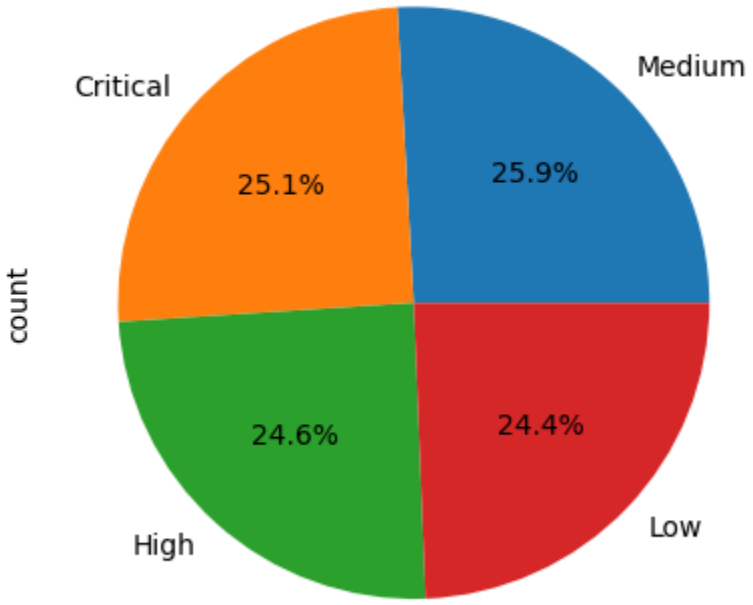
Ticket Status	Ticket Type
Closed	2769
Open	2819
Pending Customer Response	2881

dtype: int64

In [ ]:
```
Status_counts = df['Ticket Status'].value_counts()
ax = Status_counts.plot(kind='barh', color='green')
for i, v in enumerate(Status_counts):
    ax.text(v + 0.1, i, str(v), ha='left', va='center')
pyplot.show()
```



In [ ]:
```
df['Ticket Priority'].value_counts().plot(kind="pie", autopct='%1.1f%%')
pyplot.show()
```



In [ ]:
```
df.groupby('Product Purchased')['Ticket Type'].value_counts()
```

Out[ ]:

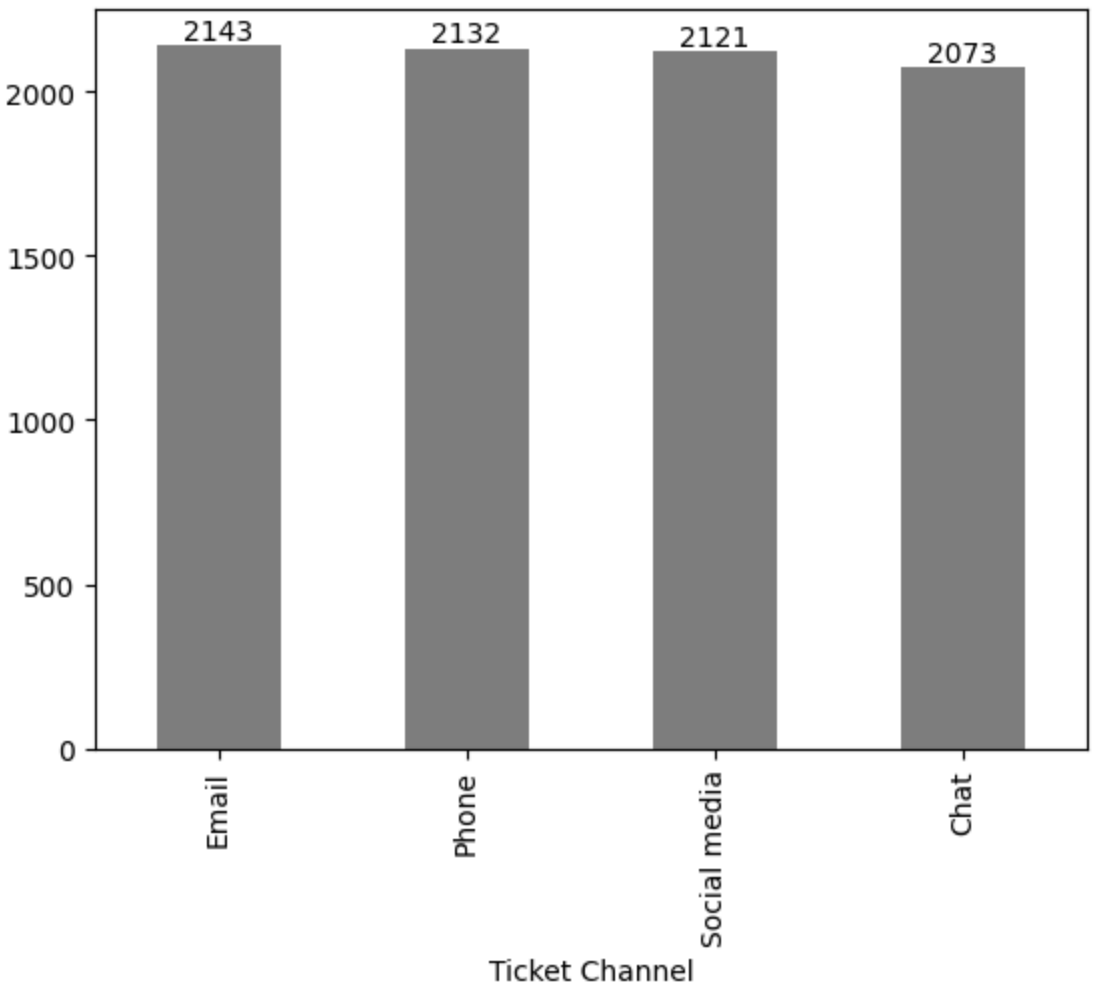
Product Purchased	Ticket Type	count
Adobe Photoshop	**Cancellation request**	39
	Billing inquiry	38
	Technical issue	36
	Refund request	35
	Product inquiry	33
...	**...**	...
iPhone	**Billing inquiry**	51
	Cancellation request	45
	Technical issue	43
	Refund request	42
	Product inquiry	31

211 rows × 1 columns

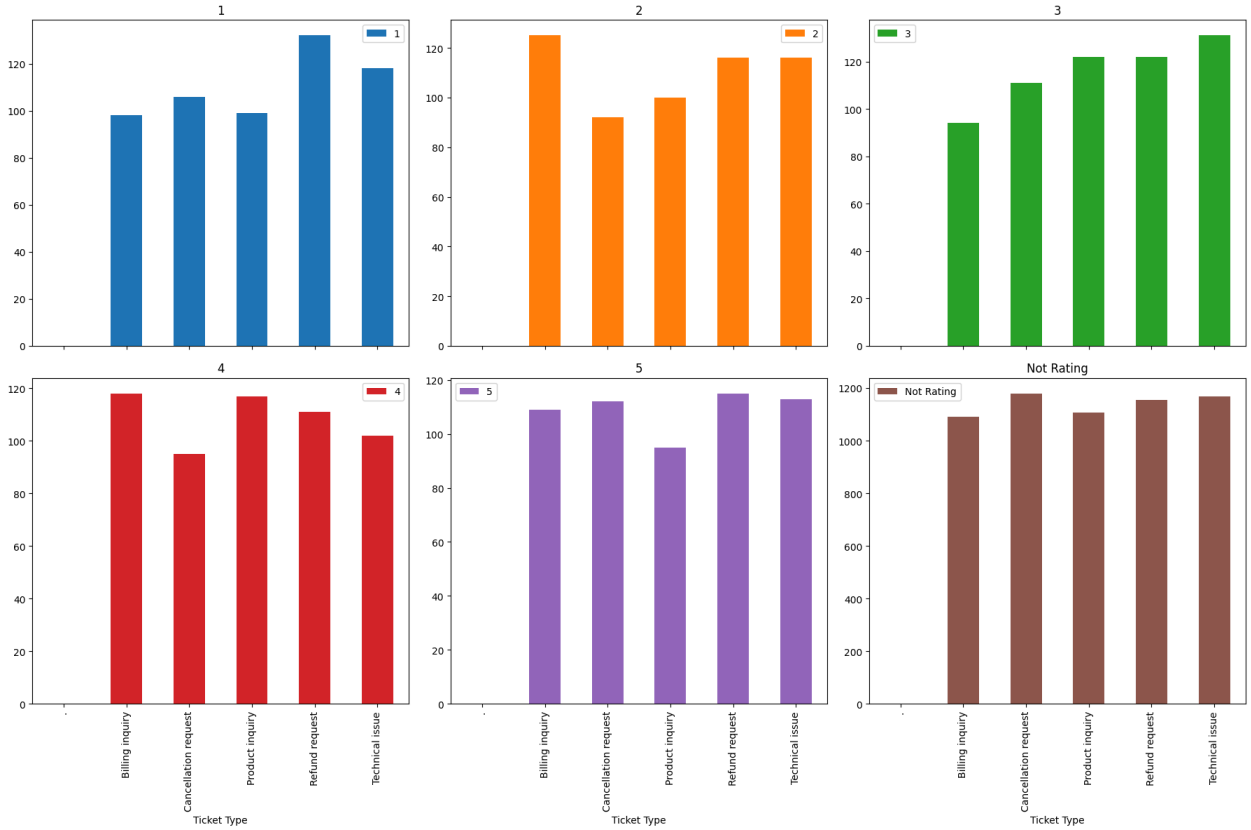
dtype: int64

In [ ]:

```
channel_counts = df['Ticket Channel'].value_counts()
ax = channel_counts.plot(kind="bar" , color = "grey")
for i, v in enumerate(channel_counts):
    ax.text(i, v + 0.1, str(v), ha='center', va='bottom')
pyplot.show()
```



In [12]:
```
# product wise rating
grouped_counts = df.groupby('Ticket Type')['Customer Satisfaction Rating'].val
pivot_table = grouped_counts.unstack(fill_value=0)
pivot_table.plot(kind='bar', subplots=True, figsize=(18, 12), layout=(2, 3), l
pyplot.tight_layout()
pyplot.show()
```



In [25]:
```
average_age = df['Customer Age'].mean().round(2)

# Count customers with age more than average
num_more_than_average = (df['Customer Age'] > average_age).sum()

# Count customers with age less than or equal to average
num_less_than_or_equal_to_average = (df['Customer Age'] <= average_age).sum()

print(f"Average Customer Age: {average_age}")
print(f"Number of customers with age more than average: {num_more_than_average
print(f"Number of customers with age less than or equal to average: {num_less_
```

```
Average Customer Age: 44.03
Number of customers with age more than average: 4188
Number of customers with age less than or equal to average: 4281
```

In [43]:
```
# Number of ticket per month
df['Date of Purchase'] = pd.to_datetime(df['Date of Purchase'])

# Create a list of month names in chronological order
month_order = ['January', 'February', 'March', 'April', 'May', 'June',
               'July', 'August', 'September', 'October', 'November', 'December

# Extract month names into a new 'Month' column
df['Month'] = df['Date of Purchase'].dt.month_name()

# Convert 'Month' column to categorical with the chronological order
df['Month'] = pd.Categorical(df['Month'], categories=month_order, ordered=True
```

```
# Group by 'Month' and count 'Ticket ID's
df.groupby('Month')['Ticket ID'].count()
```

```
/tmp/ipython-input-1457706915.py:15: FutureWarning: The default of observed=Fal
se is deprecated and will be changed to True in a future version of pandas. Pas
s observed=False to retain current behavior or observed=True to adopt the futur
e default and silence this warning.
  df.groupby('Month')['Ticket ID'].count()
```

Out[43]:

Month	Ticket ID
January	736
February	715
March	672
April	718
May	701
June	678
July	727
August	691
September	696
October	735
November	704
December	696

dtype: int64

In [ ]:

```
# prduct wise avg rating
print(df.groupby('Product Purchased')['Customer Satisfaction Rating'].value_cc
```

Out[ ]:

		count
Product Purchased	**Customer Satisfaction Rating**	
Adobe Photoshop	**Not Rating**	118
	1	16
	5	15
	3	14
	2	9
...	**...**	...
iPhone	**5**	20
	3	17
	1	15
	2	15
	4	15

252 rows × 1 columns

dtype: int64