```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
d = pd.read_csv("Diwali_sales.csv",encoding="unicode_escape")
sns.set(style = "whitegrid")
```

```
d
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Categ
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	A
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	A
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	A
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	A
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	A
...	...	...	...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	Chemical	O
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	Healthcare	Veteri
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	Textile	O
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	O
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	O

11239 rows × 14 columns

Next steps: Generate code with d | View recommended plots | New interactive sheet

```
d.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 15 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   User_ID           11251 non-null  int64
 1   Cust_name         11251 non-null  object
 2   Product_ID        11251 non-null  object
 3   Gender            11251 non-null  object
 4   Age Group         11251 non-null  object
 5   Age               11251 non-null  int64
 6   Marital_Status    11251 non-null  int64
 7   State             11251 non-null  object
 8   Zone              11251 non-null  object
 9   Occupation        11251 non-null  object
 10  Product_Category  11251 non-null  object
 11  Orders            11251 non-null  int64
 12  Amount            11239 non-null  float64
 13  Status            0 non-null      float64
 14  unnamed1          0 non-null      float64
dtypes: float64(3), int64(4), object(8)
memory usage: 1.3+ MB
```

```
print(d.isnull().sum())
```

```
User_ID             0
Cust_name           0
Product_ID          0
Gender              0
Age Group           0
```

Gem •••

disc
the
use
enc
in th
code
snip
you
prov

The
enc
argu
is
used
in th
pd.r
func
to
hand
char
enc
issu
that
migh
arise
whe
read
a CS
file.

Reas

```
Age                   0
Marital_Status        0
State                 0
Zone                  0
Occupation            0
Product_Category      0
Orders                0
Amount               12
Status            11251
unnamed1          11251
dtype: int64
```

```
print(d.drop(columns = "unnamed1: 0",errors = "ignore"))
print(d.columns)
```

```
       User_ID    Cust_name Product_ID Gender Age Group  Age  Marital_Status  \
0      1002903    Sanskriti  P00125942      F     26-35   28               0
1      1000732       Kartik  P00110942      F     26-35   35               1
2      1001990        Bindu  P00118542      F     26-35   35               1
3      1001425       Sudevi  P00237842      M      0-17   16               0
4      1000588         Joni  P00057942      M     26-35   28               1
...        ...          ...        ...    ...       ...  ...             ...
11246  1000695      Manning  P00296942      M     18-25   19               1
11247  1004089  Reichenbach  P00171342      M     26-35   33               0
11248  1001209        Oshin  P00201342      F     36-45   40               0
11249  1004023       Noonan  P00059442      M     36-45   37               0
11250  1002744      Brumley  P00281742      F     18-25   19               0

                State      Zone       Occupation Product_Category  Orders  \
0         Maharashtra   Western       Healthcare             Auto       1
1      Andhra Pradesh  Southern             Govt             Auto       3
2       Uttar Pradesh   Central       Automobile             Auto       3
3           Karnataka  Southern     Construction             Auto       2
4             Gujarat   Western  Food Processing             Auto       2
...               ...       ...              ...              ...     ...
11246     Maharashtra   Western         Chemical           Office       4
11247         Haryana  Northern       Healthcare       Veterinary       3
11248  Madhya Pradesh   Central          Textile           Office       4
11249       Karnataka  Southern      Agriculture           Office       3
11250     Maharashtra   Western       Healthcare           Office       3

        Amount  Status  unnamed1
0      23952.0     NaN       NaN
1      23934.0     NaN       NaN
2      23924.0     NaN       NaN
3      23912.0     NaN       NaN
4      23877.0     NaN       NaN
...        ...     ...       ...
11246    370.0     NaN       NaN
11247    367.0     NaN       NaN
11248    213.0     NaN       NaN
11249    206.0     NaN       NaN
11250    188.0     NaN       NaN

[11251 rows x 15 columns]
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount', 'Status', 'unnamed1'],
      dtype='object')
```

```
d.drop(columns = ["unnamed1"] , inplace = True)
```

Double-click (or enter) to edit

```
d["Amount"].head(10)
```

	Amount
0	23952.00
1	23934.00
2	23924.00
3	23912.00
4	23877.00
5	23877.00
6	23841.00
7	NaN
8	23809.00
9	23799.99

dtype: float64

```
print(d.columns)
```

```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount'],
      dtype='object')
```

```
d.dtypes
```

	0
User_ID	int64
Cust_name	object
Product_ID	object
Gender	object
Age Group	object
Age	int64
Marital_Status	int64
State	object
Zone	object
Occupation	object
Product_Category	object
Orders	int64
Amount	float64

dtype: object

```
d.dropna(inplace = True)
print(d["Amount"].head(10))
```

```
0     23952.00
1     23934.00
2     23924.00
3     23912.00
4     23877.00
5     23877.00
6     23841.00
8     23809.00
9     23799.99
10    23770.00
Name: Amount, dtype: float64
```

```
d
```

In th spec case

It's likely that the "Diw file cont som spec char or sym relat to th Diwa festi (whi ofte uses non- ASC char To ensu thes char are read corr by panc the `enco` argu is usec

I hop this clari the use `unic` in yc code If yo have

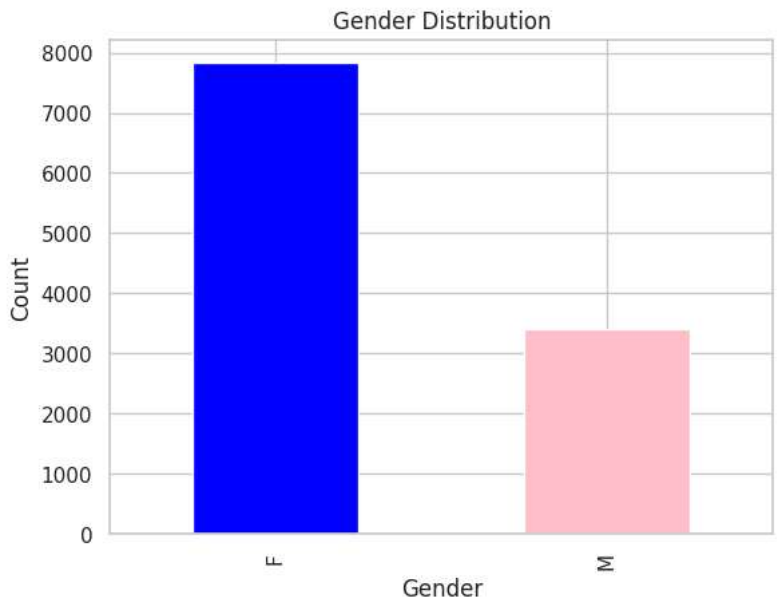
User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount

```
#exploratery data analysis
exp  = d["Product_Category"].value_counts().head(10)

print(exp)
```

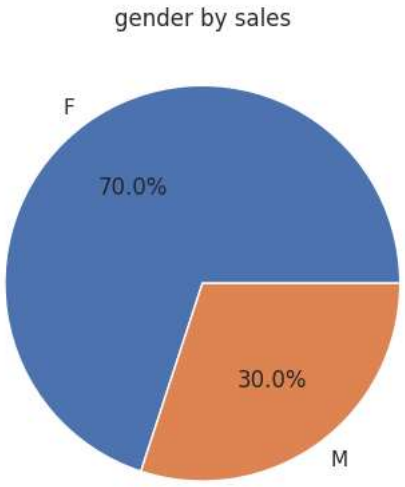
```
Product_Category
Clothing & Apparel       2655
Food                     2490
Electronics & Gadgets    2087
Footwear & Shoes         1059
Household items           520
Beauty                    422
Games & Toys              386
Sports Products           356
Furniture                 352
Pet Care                  212
Name: count, dtype: int64
```

```
d['Gender'].value_counts().plot(kind='bar', color=['blue', 'pink'])
plt.title('Gender Distribution')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.show()
```

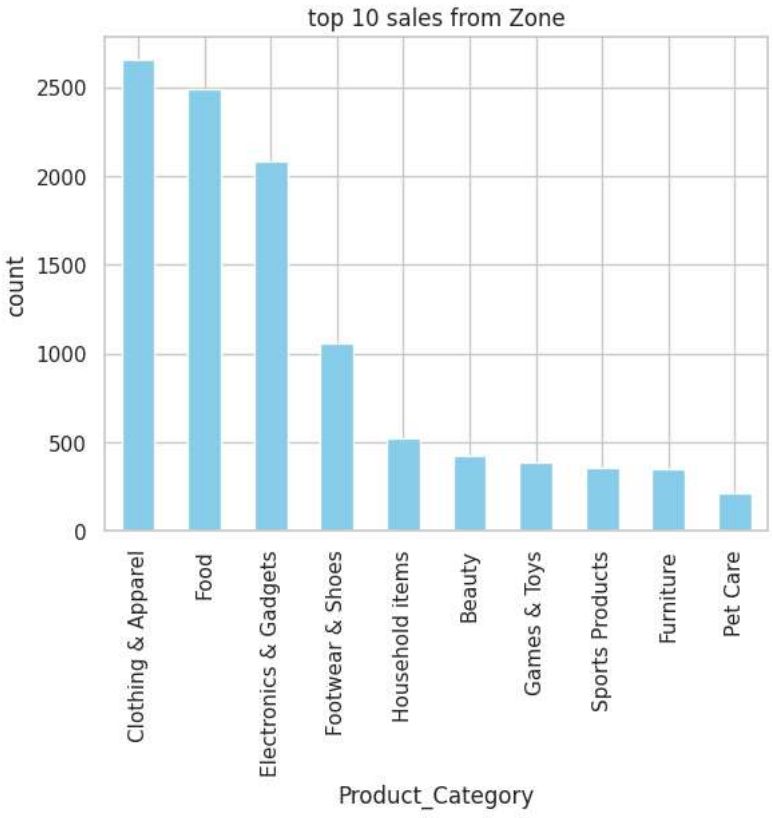



```
gender_by_group = d.groupby("Gender")["Amount"].sum()
print(gender_by_group)
gender_by_group.plot(kind = "pie" ,autopct = "%1.1f%%",title = "gender by sales")
plt.ylabel("")
plt.show()
```

```
Gender
F    74335856.43
M    31913276.00
Name: Amount, dtype: float64
```

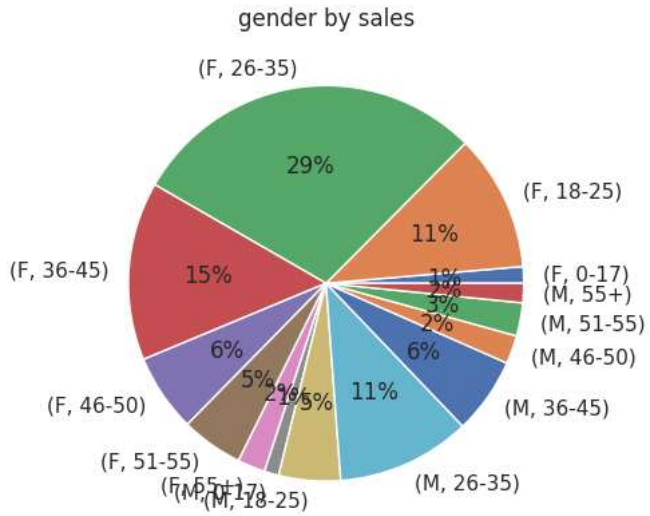



```
exp.plot(kind = "bar",title = "top 10 sales from Zone" , color = "skyblue")
plt.ylabel("count")
plt.show()
```

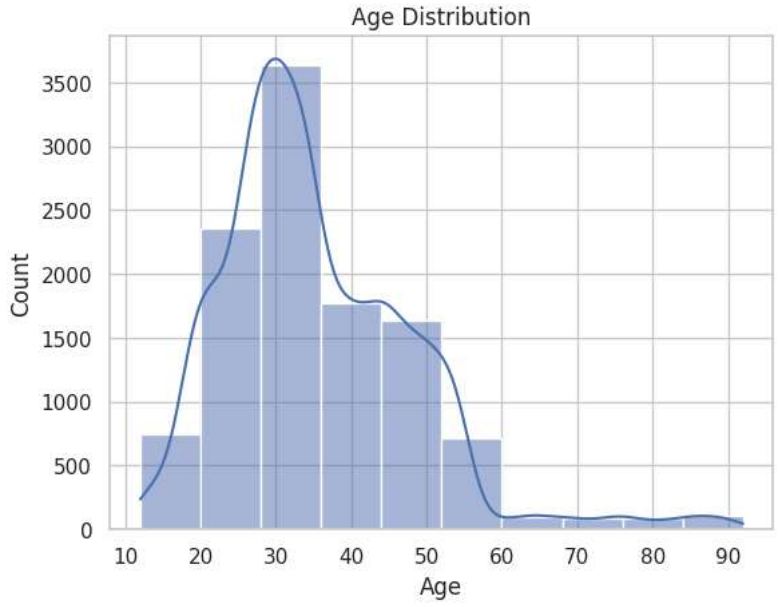



```
gender_by_group = d.groupby(["Gender","Age Group"])["Amount"].sum()
print(gender_by_group)
gender_by_group.plot(kind = "pie" ,autopct = "%1.f%%",title = "gender by sales")
plt.ylabel("")
plt.show()
```

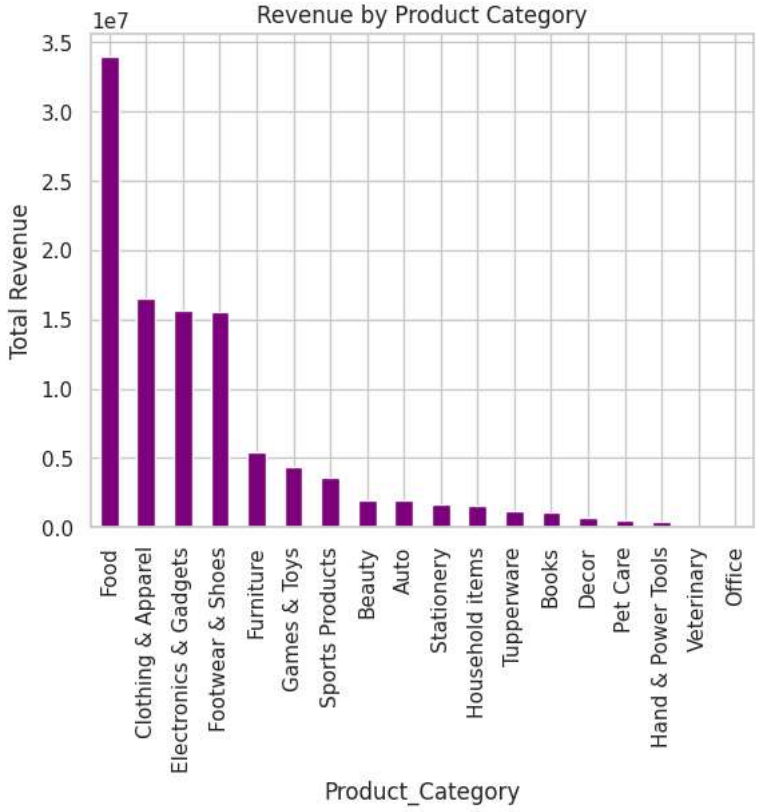
```
Gender  Age Group
F       0-17          1441409.00
        18-25        11887003.00
        26-35        30963954.94
        36-45        15509957.49
        46-50         6743393.00
        51-55         5385208.00
        55+           2404931.00
M       0-17          1258244.00
        18-25         5353729.00
        26-35        11649489.00
        36-45         6635038.00
        46-50         2464451.00
        51-55         2876269.00
        55+           1676056.00
Name: Amount, dtype: float64
```




```
sns.histplot(d['Age'], bins=10, kde=True)
plt.title('Age Distribution')
plt.show()
```




```
category_revenue = d.groupby('Product_Category')['Amount'].sum().sort_values(ascending=False)
category_revenue.plot(kind='bar', color='purple')
plt.title('Revenue by Product Category')
plt.ylabel('Total Revenue')
plt.show()
```




```
d['date'] = pd.date_range(start='1995-10-01', periods=len(d))
```

```
d
```

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Categ
0	1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	A
1	1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	A
2	1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	A
3	1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	A
4	1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	A
...	...	...	...	...	...	...	...	...	...	...	
11246	1000695	Manning	P00296942	M	18-25	19	1	Maharashtra	Western	Chemical	O
11247	1004089	Reichenbach	P00171342	M	26-35	33	0	Haryana	Northern	Healthcare	Veteri
11248	1001209	Oshin	P00201342	F	36-45	40	0	Madhya Pradesh	Central	Textile	O
11249	1004023	Noonan	P00059442	M	36-45	37	0	Karnataka	Southern	Agriculture	O
11250	1002744	Brumley	P00281742	F	18-25	19	0	Maharashtra	Western	Healthcare	O

11239 rows × 14 columns

Next steps: Generate code with d | View recommended plots | New interactive sheet

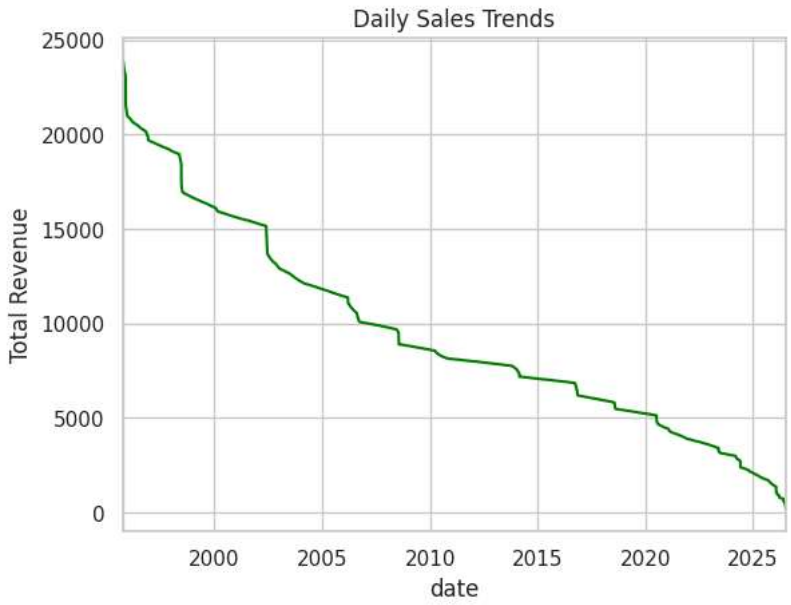
```
d['date'] = pd.to_datetime(d['date'])  # Ensure date format
daily_sales = d.groupby('date')['Amount'].sum()
```

char
in th
worl
Each
char
in th
dicti
has
uniq
code
poin
UTF-
is lik
a
trans
that
conv
thes
code
poin
into
form
that
com
can
unde
(a
sequ
of
byte
It's a
flexi
and
effic
way
repre
text
from
any
lang

Exar

The
lette
"A" h
the
Unic
code
poin
U+0
In
UTF-

```
daily_sales.plot(kind='line', color='green')
plt.title('Daily Sales Trends')
plt.ylabel('Total Revenue')
plt.show()
```



Start coding or generate with AI.

poin
U+2(
In
UTF-
it's
encc
as
thre
byte
0xE2
0x82
0xA(

I hop
this
expl
help
Let r
knov
if yo
have
any
othe
ques
Rate
this
answ

YAM
BHA
KUM

plea
clea
abou
data
featu

Gem

Okay
let's
clari
the
data
featu
in yc
Diwa
sale
data

To
unde
the
data
featu