```
# 1. Import Libraries
import pandas as pd
import numpy as np


# 2. Load and Explore Dataset
df = pd.read_csv("/content/product_usage.csv")

print("📌 First 5 rows of the dataset:")
print(df.head())

print("\n📌 Dataset Info:")
print(df.info())

print("\n📌 Summary Statistics:")
print(df.describe(include='all'))
```

```
📌 First 5 rows of the dataset:
   uid                ts       feat
0  101  2025-06-01 10:00  Dashboard
1  102  2025-06-01 10:05   Calendar
2  103  2025-06-01 10:10      Notes
3  101  2025-06-01 10:15  Dashboard
4  104  2025-06-01 10:20   Calendar

📌 Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   uid     10 non-null     int64
 1   ts      10 non-null     object
 2   feat    10 non-null     object
dtypes: int64(1), object(2)
memory usage: 372.0+ bytes
None

📌 Summary Statistics:
              uid                ts       feat
count    10.00000                10         10
unique        NaN                10          3
top           NaN  2025-06-01 10:00  Dashboard
freq          NaN                 1          4
mean    103.10000               NaN        NaN
std       1.66333               NaN        NaN
min     101.00000               NaN        NaN
25%     102.00000               NaN        NaN
50%     103.00000               NaN        NaN
75%     104.00000               NaN        NaN
max     106.00000               NaN        NaN
```

```
# 3. Clean and Prepare the Data
df.rename(columns={
    'uid': 'user_id',
    'ts': 'timestamp',
    'feat': 'feature_used'
}, inplace=True)

print("\n📌 Missing Values Before Cleaning:")
print(df.isnull().sum())

df.dropna(inplace=True)

df['timestamp'] = pd.to_datetime(df['timestamp'])

print("\n📌 Updated Data Types:")
print(df.dtypes)
```

```
📌 Missing Values Before Cleaning:
user_id         0
timestamp       0
feature_used    0
dtype: int64

📌 Updated Data Types:
user_id                  int64
timestamp       datetime64[ns]
feature_used            object
dtype: object
```

```
# 4. Analyze Feature Usage
feature_counts = df['feature_used'].value_counts().reset_index()
```

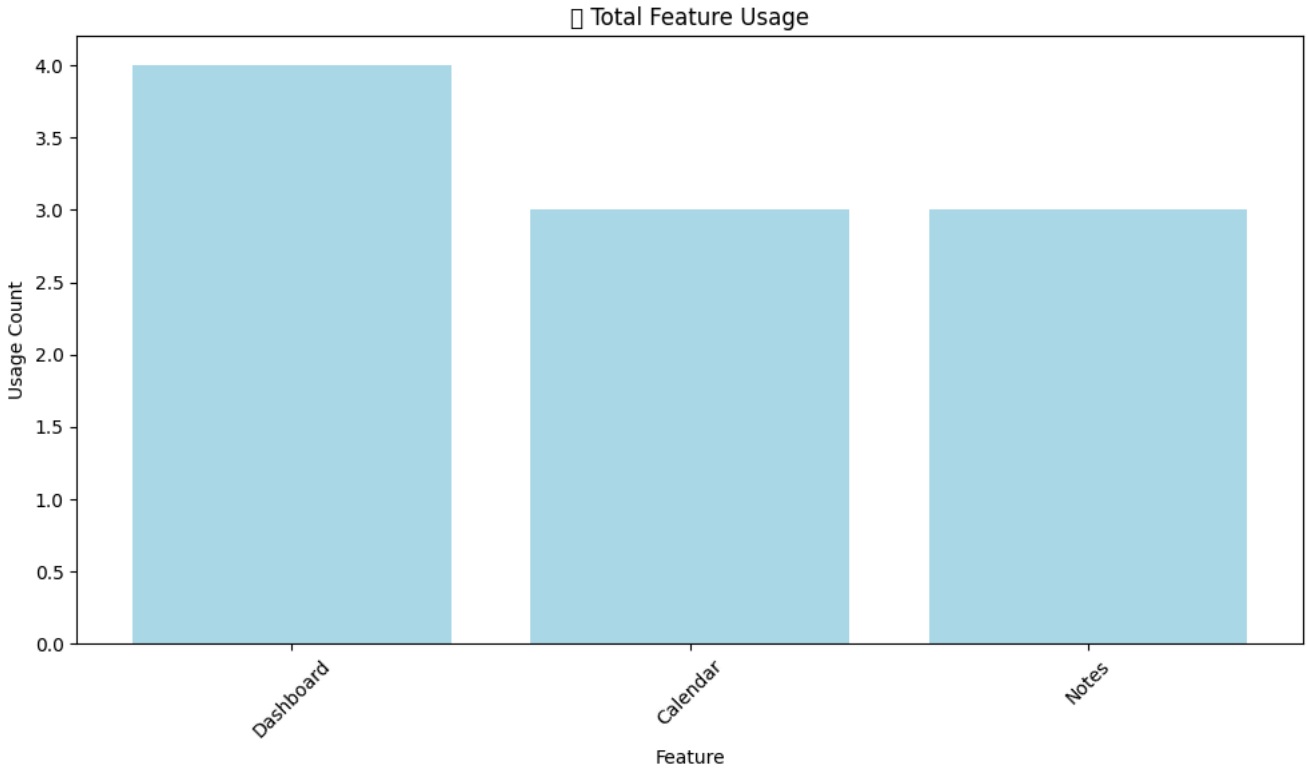
```
feature_counts.columns = ['feature', 'total_usage']
avg_usage = df.groupby(['user_id', 'feature_used']).size().groupby('feature_used').mean().reset_index()
avg_usage.columns = ['feature', 'avg_usage_per_user']
summary = pd.merge(feature_counts, avg_usage, on='feature')
print("\n📌 Feature Usage Summary Table:")
print(summary)
```

```
📌 Feature Usage Summary Table:
     feature  total_usage  avg_usage_per_user
0  Dashboard            4            1.333333
1   Calendar            3            1.000000
2      Notes            3            1.000000
```

```
# 5. Visualization (Optional)
import matplotlib.pyplot as plt

plt.figure(figsize=(10,6))
plt.bar(summary['feature'], summary['total_usage'], color='lightblue')
plt.title("📊 Total Feature Usage")
plt.xlabel("Feature")
plt.ylabel("Usage Count")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
<ipython-input-9-2651240757>:10: UserWarning: Glyph 128202 (\N{BAR CHART}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 128202 (\N{BAR CHART}) missing from font
  fig.canvas.print_figure(bytes_io, **kw)
```




```
# 6. Markdown Summary (Written observations)
from IPython.display import display, Markdown

display(Markdown("## 💡 Summary Insights"))
display(Markdown(f"- Total unique features: **{df['feature_used'].nunique()}**"))
display(Markdown(f"- Total unique users: **{df['user_id'].nunique()}**"))
top_feature = summary.loc[summary['total_usage'].idxmax()]
display(Markdown(f"- **Most Used Feature**: `{top_feature['feature']}` with `{top_feature['total_usage']}` uses"))
```

💡 Summary Insights

- Total unique features: **3**
- Total unique users: **6**
- **Most Used Feature**: `Dashboard` with `4` uses