Double-click (or enter) to edit

```
import pandas  as pd
df = pd.read_csv('/content/Sales_April_2019.csv')
print("FIrst 5 rows:")
print(df.head(5))
```

```
FIrst 5 rows:
  Order ID                    Product Quantity Ordered Price Each  \
0   176558        USB-C Charging Cable                2      11.95
1      NaN                         NaN              NaN        NaN
2   176559  Bose SoundSport Headphones                1      99.99
3   176560                Google Phone                1        600
4   176560            Wired Headphones                1      11.99

       Order Date                      Purchase Address
0  04/19/19 08:46          917 1st St, Dallas, TX 75001
1             NaN                                   NaN
2  04/07/19 22:30     682 Chestnut St, Boston, MA 02215
3  04/12/19 14:38  669 Spruce St, Los Angeles, CA 90001
4  04/12/19 14:38  669 Spruce St, Los Angeles, CA 90001
```

```
print("Shape of dataset:", df.shape)
print("Column names:", df.columns)
print("Info:")
print(df.info())
```

```
Shape of dataset: (18383, 6)
Column names: Index(['Order ID', 'Product', 'Quantity Ordered', 'Price Each', 'Order Date',
       'Purchase Address'],
      dtype='object')
Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18383 entries, 0 to 18382
Data columns (total 6 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Order ID          18324 non-null  object
 1   Product           18324 non-null  object
 2   Quantity Ordered  18324 non-null  object
 3   Price Each        18324 non-null  object
 4   Order Date        18324 non-null  object
 5   Purchase Address  18324 non-null  object
dtypes: object(6)
memory usage: 861.8+ KB
None
```

```
print("Missing values in each column:", df.isnull().sum())
```

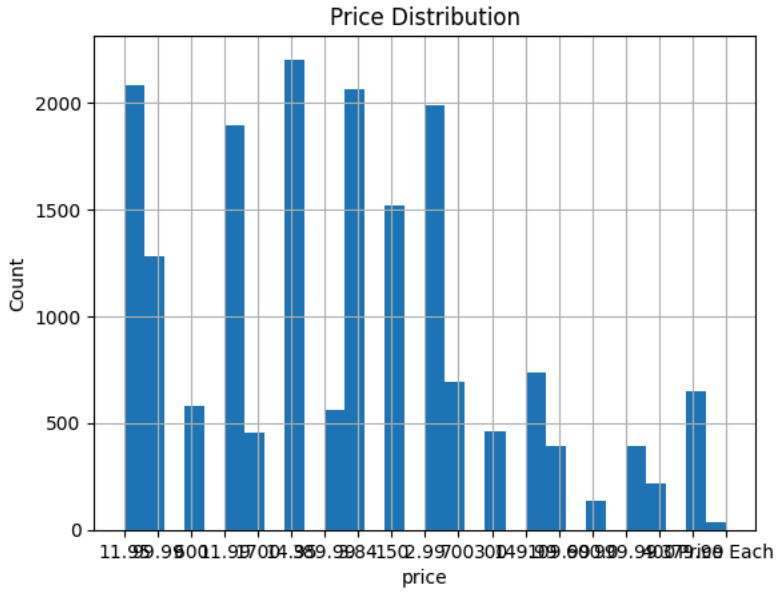
```
Missing values in each column: Order ID           59
Product             59
Quantity Ordered    59
Price Each          59
Order Date          59
Purchase Address    59
dtype: int64
```

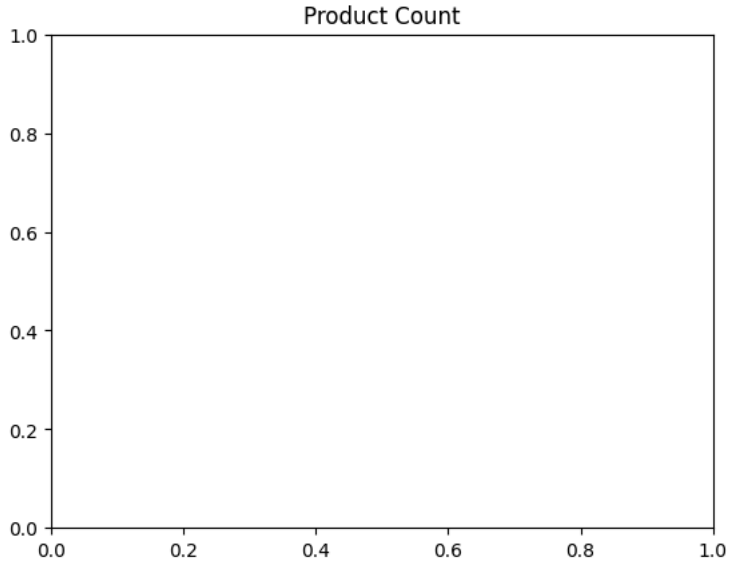
```
print("Number of duplicate rows:", df.duplicated().sum())
```

```
Number of duplicate rows: 114
```

```
import matplotlib.pyplot as plt
import seaborn as sns
df['Price Each'].hist(bins=30)
plt.title('Price Distribution')
plt.xlabel('price')
plt.ylabel('Count')
plt.show()

sns.countplot(Y='Product', data=df)
plt.title('Product Count')
plt.show()
```





Start coding or generate with AI.

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipython-input-32-957416276.py in <cell line: 0>()
----> 1 sns.countplot(y='product', data=df)
      2 plt.title('Product Count')
      3 plt.show()

                                    5 frames
/usr/local/lib/python3.11/dist-packages/seaborn/_core/data.py in _assign_variables(self, data, variables)
    230                 else:
    231                     err += "An entry with this name does not appear in `data`."
--> 232                 raise ValueError(err)
    233
    234             else:

ValueError: Could not interpret value `product` for `y`. An entry with this name does not appear in `data`.
```

```
# Basic Banking System

# Predefined login details
username = "kartik"
password = "1234"

# Login
print("🔐 Welcome to Python Bank 🔐")
input_user = input("Enter username: ")
input_pass = input("Enter password: ")
```

```
# Check login
if input_user == username and input_pass == password:
    print("\n✅ Login Successful!")
    balance = 1000  # Initial balance

    while True:
        print("\n📋 --- Banking Menu ---")
        print("1. Check Balance")
        print("2. Credit (Deposit)")
        print("3. Debit (Withdraw)")
        print("4. Exit")

        choice = input("Enter your choice (1-4): ")

        if choice == '1':
            print(f"💰 Current Balance: ₹{balance}")

        elif choice == '2':
            amount = float(input("Enter amount to credit: ₹"))
            if amount > 0:
                balance += amount
                print(f"✅ Credited ₹{amount}. New Balance: ₹{balance}")
            else:
                print("❌ Invalid amount!")

        elif choice == '3':
            amount = float(input("Enter amount to debit: ₹"))
            if 0 < amount <= balance:
                balance -= amount
                print(f"✅ Debited ₹{amount}. New Balance: ₹{balance}")
            else:
                print("❌ Insufficient balance or invalid amount!")

        elif choice == '4':
            print("👋 Thank you for using Python Bank!")
            break

        else:
            print("❌ Invalid choice. Please select 1 to 4.")

else:
    print("❌ Login failed. Invalid username or password.")
```

```
🔐 Welcome to Python Bank 🔐
Enter username: kartik
Enter password: 1234

✅ Login Successful!

📋 --- Banking Menu ---
1. Check Balance
2. Credit (Deposit)
3. Debit (Withdraw)
4. Exit
Enter your choice (1-4): 1
💰 Current Balance: ₹1000

📋 --- Banking Menu ---
1. Check Balance
2. Credit (Deposit)
3. Debit (Withdraw)
4. Exit
Enter your choice (1-4): 2
Enter amount to credit: ₹5000
✅ Credited ₹5000.0. New Balance: ₹6000.0

📋 --- Banking Menu ---
1. Check Balance
2. Credit (Deposit)
3. Debit (Withdraw)
4. Exit
Enter your choice (1-4): 3
Enter amount to debit: ₹1500
✅ Debited ₹1500.0. New Balance: ₹4500.0

📋 --- Banking Menu ---
1. Check Balance
2. Credit (Deposit)
3. Debit (Withdraw)
4. Exit
Enter your choice (1-4): 4
👋 Thank you for using Python Bank!
```