```python
import pandas as pd

# Step 1: Create Customers Dataset

customers = pd.DataFrame({
 'customer_id': [101, 102, 103, 104, 105],
 'name': ['Ram', 'Jai', 'charan', 'Dev', 'kam'],
 'city': ['Mumbai', 'Delhi', 'Mumbai', 'Chennai', 'Delhi']
 })

# Step 2: Create Transactions Dataset

transactions = pd.DataFrame({
 'transaction_id': [1, 2, 3, 4, 5, 6, 7],
 'customer_id': [101, 102, 103, 101, 104, 102, 105],
 'amount': [200, 150, 300, 450, 500, 100, 250]
 })

# Step 3: Merge on customer_id

merged_df = pd.merge(transactions, customers, on='customer_id')
print("□ Merged Dataset:")
print(merged_df)

# Step 4: Total Spend per Customer

total_spend = merged_df.groupby(['customer_id', 'name'])
['amount'].sum().reset_index()
print("\n□ Total Spend per Customer:")
print(total_spend)

# Step 5: City-wise Spending

city_spend = pd.merge(total_spend, customers[['customer_id', 'city']],
on='customer_id')
city_summary = city_spend.groupby('city')
['amount'].sum().reset_index()
print("\n□ City-wise Spending:")
print(city_summary)

# Step 6: Top Spender

top_spender = total_spend.sort_values(by='amount',
ascending=False).head(1)
print("\n□ Top Spender:")
print(top_spender)
```

```
 Merged Dataset:
   transaction_id  customer_id  amount    name     city
0               1          101     200     Ram   Mumbai
1               2          102     150     Jai    Delhi
2               3          103     300  charan   Mumbai
3               4          101     450     Ram   Mumbai
4               5          104     500     Dev  Chennai
5               6          102     100     Jai    Delhi
6               7          105     250     kam    Delhi

 Total Spend per Customer:
   customer_id    name  amount
0          101     Ram     650
1          102     Jai     250
2          103  charan     300
3          104     Dev     500
4          105     kam     250

 City-wise Spending:
      city  amount
0  Chennai     500
1    Delhi     500
2   Mumbai     950

 Top Spender:
   customer_id name  amount
0          101  Ram     650
```

```python
import pandas as pd

# Step 1: Create Patients Dataset

patients = pd.DataFrame({
 'patient_id': [201, 202, 203, 204, 205],
 'name': ['Arjun', 'Bhavya', 'Chirag', 'Deepa', 'Eshan'],
 'age': [32, 45, 28, 36, 52],
 'gender': ['M', 'F', 'M', 'F', 'M']
 })

# Step 2: Create Appointments Dataset

appointments = pd.DataFrame({
 'appointment_id': [1, 2, 3, 4, 5, 6, 7, 8],
 'patient_id': [201, 202, 201, 204, 203, 205, 202, 202],
 'doctor': ['Dr. Rao', 'Dr. Singh', 'Dr. Rao', 'Dr. Mehta', 'Dr.
Singh', 'Dr. Mehta', 'Dr. Singh', 'Dr. Singh'],
 'date': ['2025-09-01', '2025-09-01', '2025-09-05', '2025-09-02',
'2025-09-03', '2025-09-06', '2025-09-08', '2025-09-10']
 })
```

```python
# Step 3: Merge Datasets

merged_df = pd.merge(appointments, patients, on='patient_id')
print("🔗 Merged Dataset:")
print(merged_df)

# Step 4: Count Appointments per Patient

appointments_per_patient = merged_df.groupby(['patient_id', 'name'])
['appointment_id'].count().reset_index(name='visit_count')
print("\n📊 Appointments per Patient:")
print(appointments_per_patient)

# Step 5: Group by Doctor (Workload)

doctor_workload = merged_df.groupby('doctor')
['appointment_id'].count().reset_index(name='appointment_count')
print("\n👨‍⚕️ Doctor Workload:")
print(doctor_workload)

# Step 6: Find Patient with Max Visits

top_patient = appointments_per_patient.sort_values(by='visit_count',
ascending=False).head(1)
print("\n🏆 Most Frequent Patient:")
print(top_patient)
```

```
🔗 Merged Dataset:
   appointment_id  patient_id     doctor        date    name  age
gender
0               1         201    Dr. Rao  2025-09-01   Arjun   32
M
1               2         202  Dr. Singh  2025-09-01  Bhavya   45
F
2               3         201    Dr. Rao  2025-09-05   Arjun   32
M
3               4         204  Dr. Mehta  2025-09-02   Deepa   36
F
4               5         203  Dr. Singh  2025-09-03  Chirag   28
M
5               6         205  Dr. Mehta  2025-09-06   Eshan   52
M
6               7         202  Dr. Singh  2025-09-08  Bhavya   45
F
7               8         202  Dr. Singh  2025-09-10  Bhavya   45
F

📊 Appointments per Patient:
   patient_id    name  visit_count
0         201   Arjun            2
```

```
1          202  Bhavya            3
2          203  Chirag            1
3          204   Deepa            1
4          205   Eshan            1

 Doctor Workload:
       doctor  appointment_count
0  Dr. Mehta                  2
1    Dr. Rao                  2
2  Dr. Singh                  4

 Most Frequent Patient:
   patient_id    name  visit_count
1          202  Bhavya            3
```