Patients Data | Key Trends | Treatement & Cost

Medical Conditio...
All

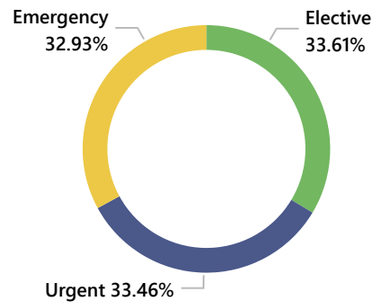
Year
All

Month
All

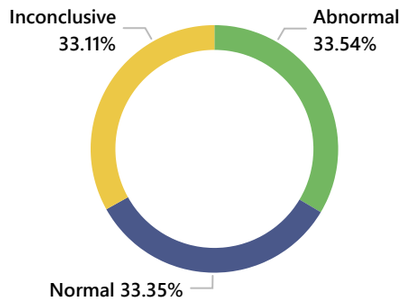
Highlights

Avg Billing Amount	Admitted Patients	Doctors	Hospitals visited	Avg Age	Bedspaces/Rooms	Avg Admission Days
25.54K	54.97K	40.34K	39.88K	51.54	400	15.50

Admitted patient by Admission Type

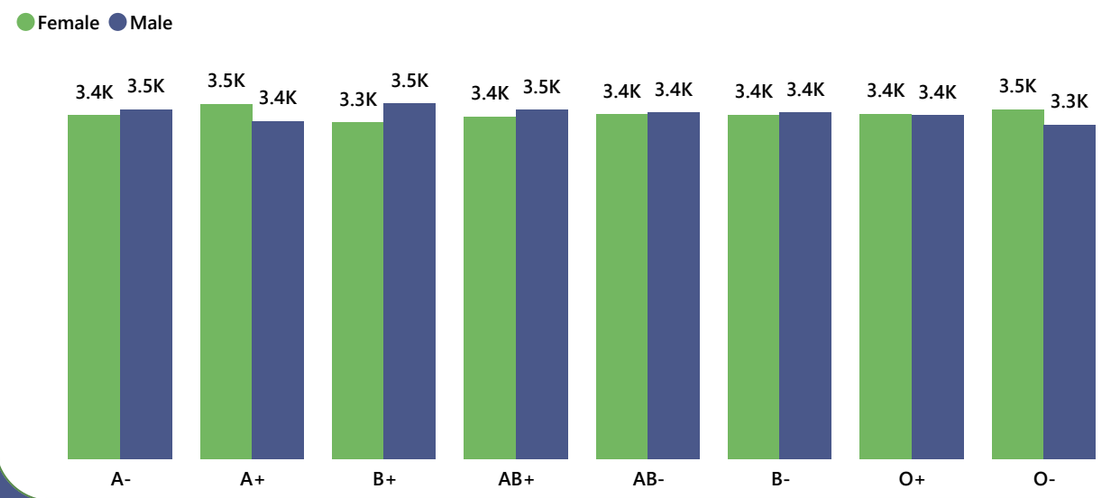


Admitted Patients by Test Results



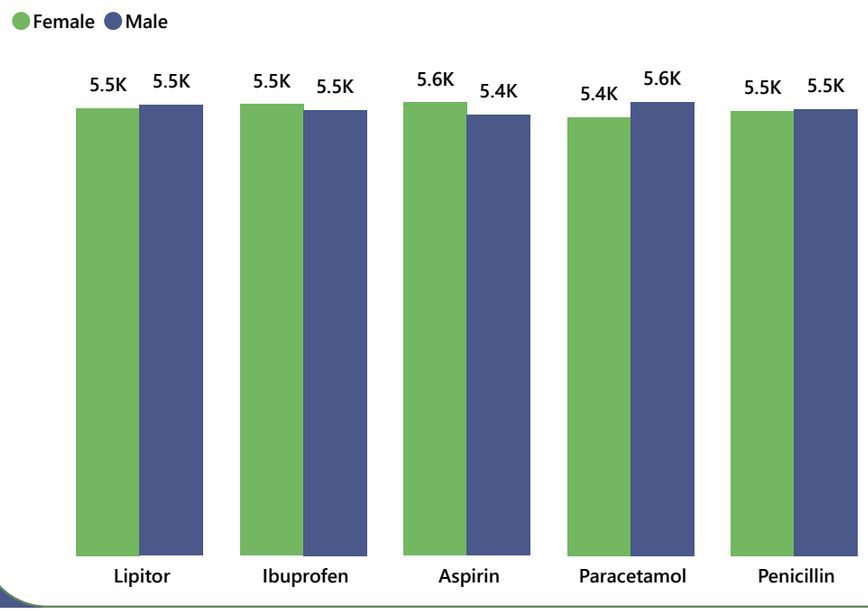
Blood Group | Medical Condition | Age Group

Admission By

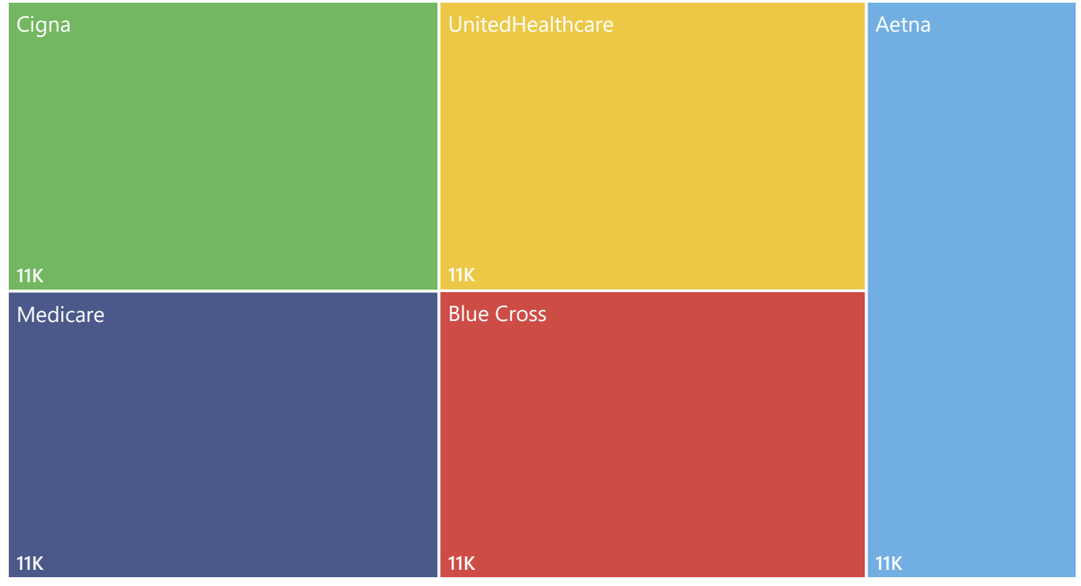


Admitted Patients

Medications | Stay(Days)



Admitted Patients By Insurance Providers



Patients Data | Key Trends | Treatement & Cost

Medical Conditio...
All

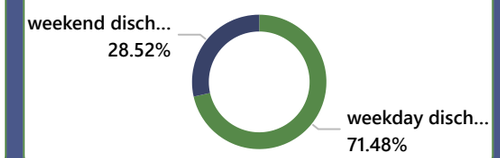
Year
All

Month
All

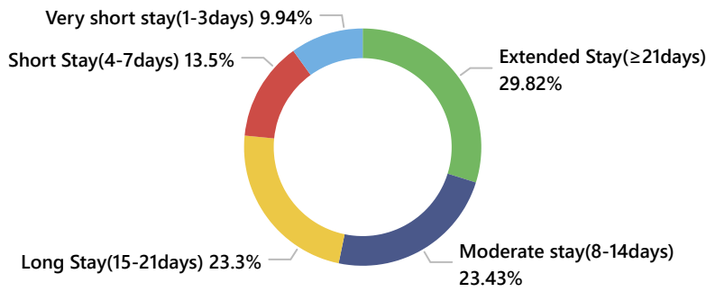
Avg Billing Amount	Admitted Patients	Doctors	Hospitals visited	Avg Age	Bedspaces/Rooms	Avg Admission Days
25.54K	54.97K	40.34K	39.88K	51.54	400	15.50

Admission - Weekend&Weekday

Discharge- Weekday&Weekend

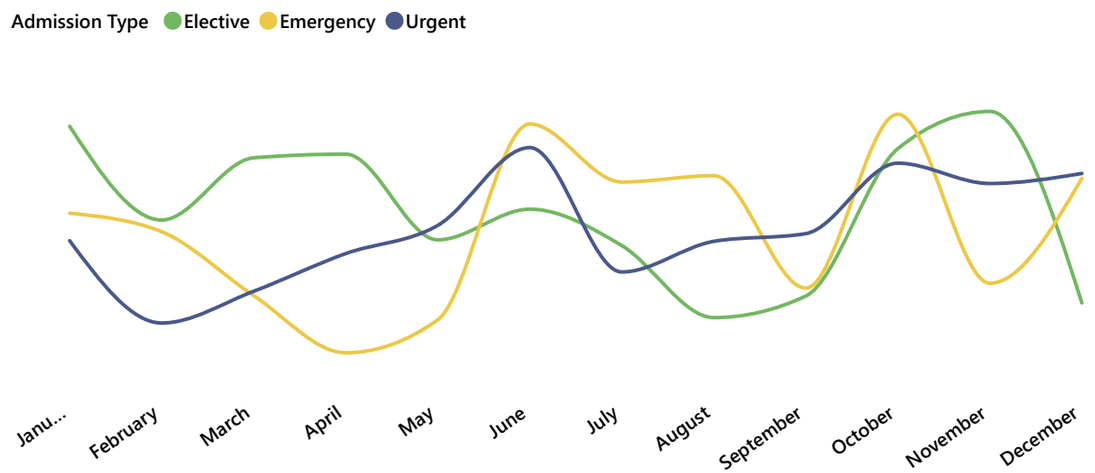


Admitted Patients by Days Admitted

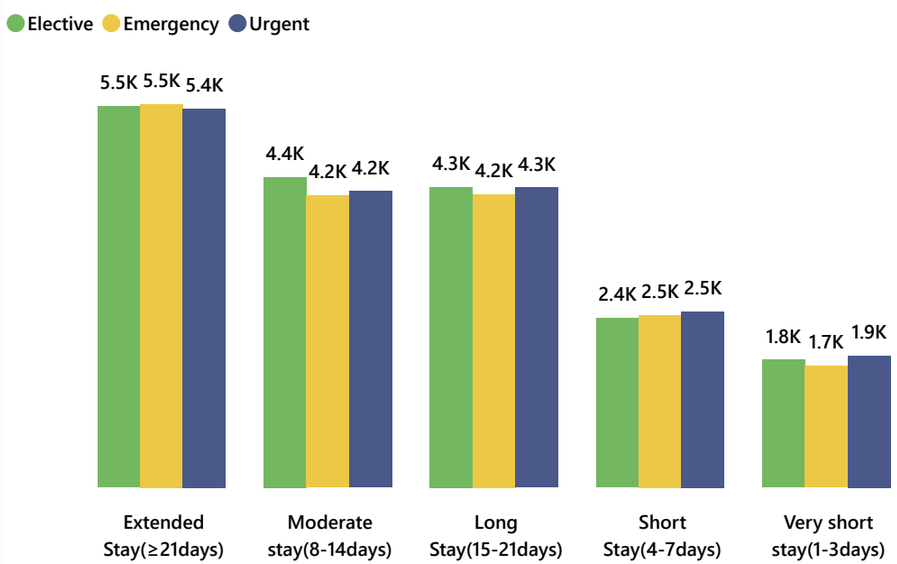


Avg Adm period | Avg Billing | Patients Admitted

Months & Admission Types By

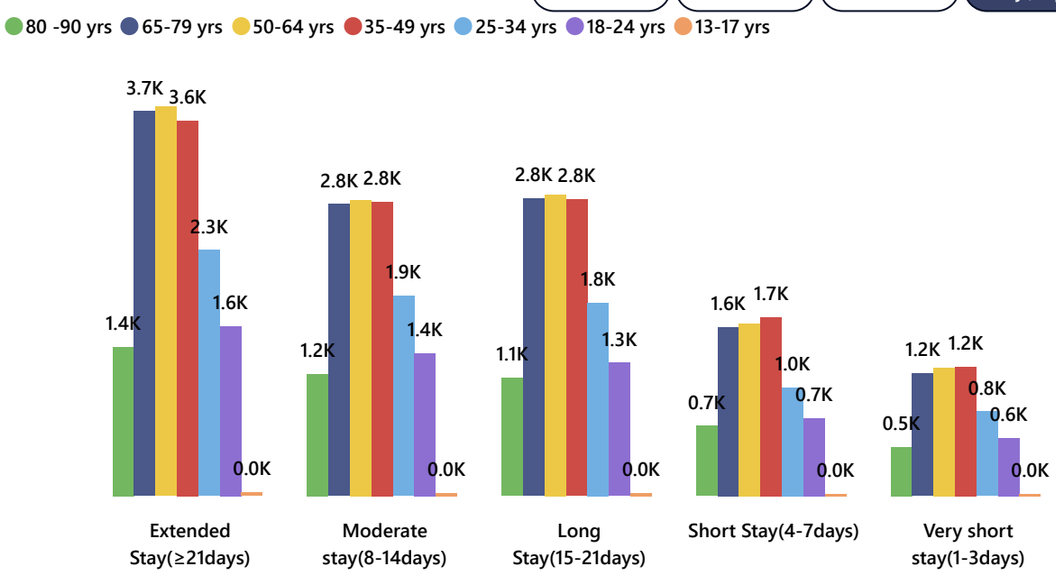


Admitted Patients by Days Admitted and Admission Type



Age Groups By

Conditions | Medications | Insurance | Stay(Days)



Patients Data | Key Trends | Treatement & Cost

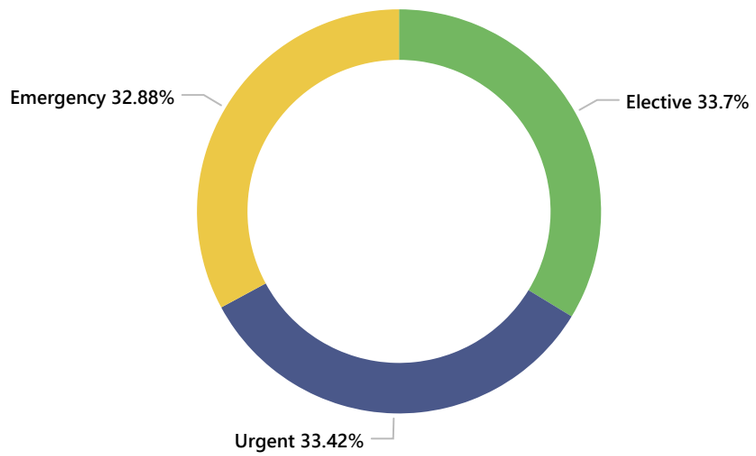
Medical Conditio...
All

Year
All

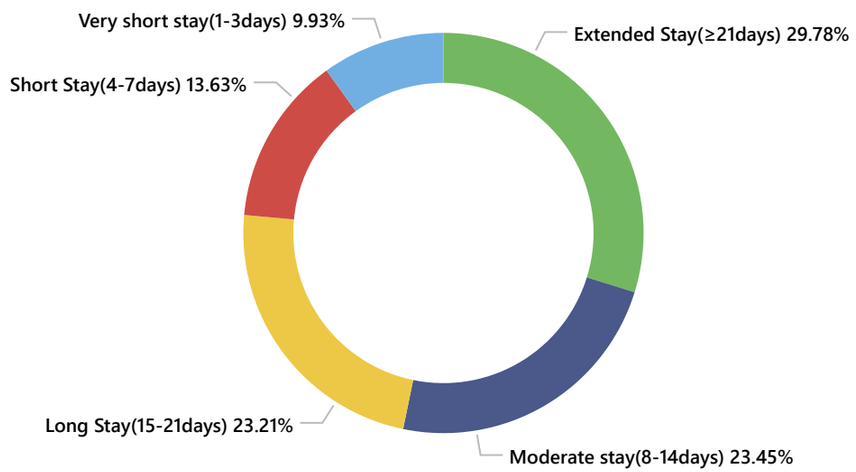
Month
All

Avg Billing Amount	Admitted Patients	Doctors	Hospitals visited	Avg Age	Bedspaces/Rooms	Avg Admission Days
25.54K	54.97K	40.34K	39.88K	51.54	400	15.50

Total Billing by Admission Type

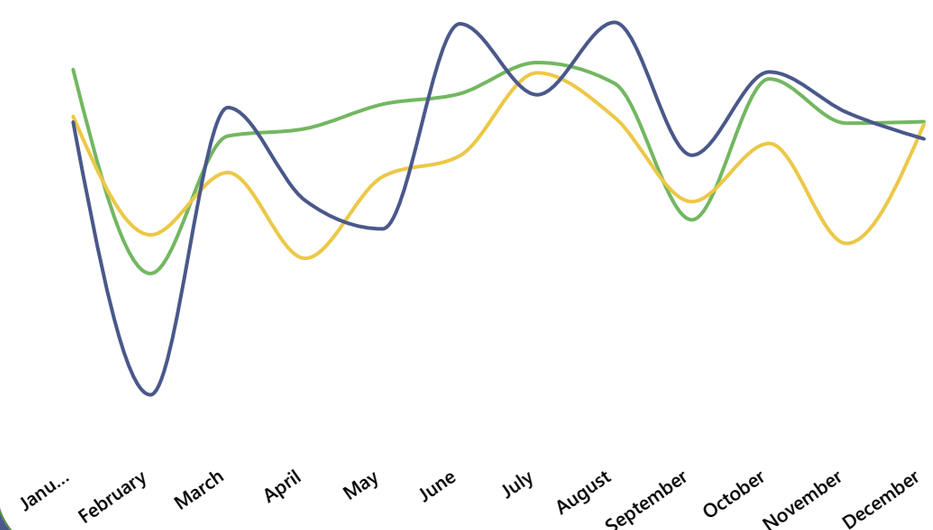


Total Billing By Days Admitted



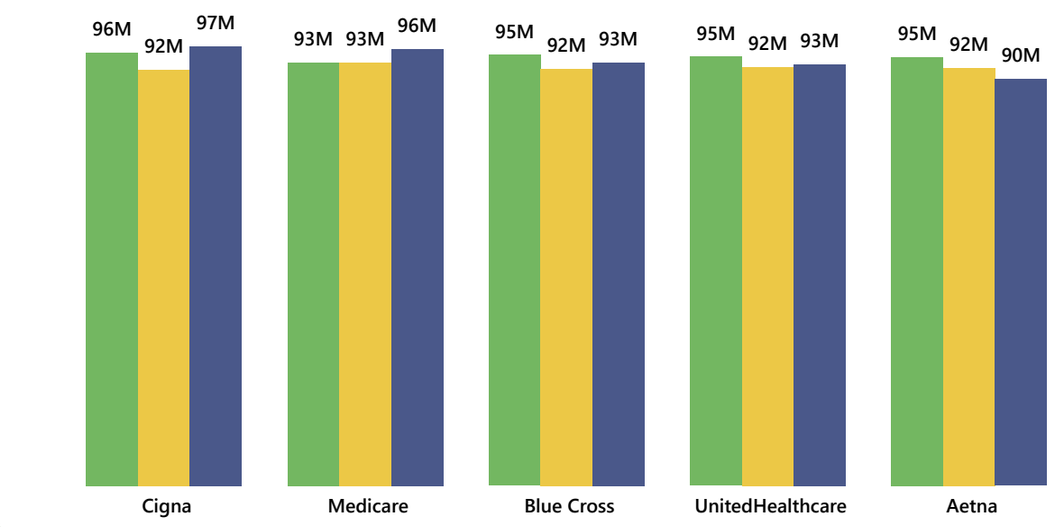
Total Billing By Quartely | Monthly

Elective Emergency Urgent



Total Billing By Condition | Insurance

Elective Emergency Urgent



Patient Dataset Requirements

KPIs

1.Sum of Billing_amount

```
[46] ✓ 0s
total_billing = df['Billing Amount'].sum()
display(total_billing)

np.float64(1404068339.2274523)
```

2.Count of patients

```
patient_count = df['Name']
display(patient_count)
```

	Name
0	Bobby Jackson
1	Leslie Terry
2	Danny Smith
3	Andrew Watts
4	Adrienne Bell
...	...
54274	Elizabeth Jackson
54275	Kyle Perez
54276	Heather Wang
54277	Jennifer Jones
54278	James Garcia

54279 rows × 1 columns

dtype: object

3.count of doctors

```
doctor_count = df['Doctor'].nunique()
display(doctor_count)

40341
```

4.average age

```
average_age = df['Age'].mean()
display(average_age)

np.float64(51.53518538733035)
```

5.Count of hospital

```
hospital_count = df['Hospital'].nunique()
display(hospital_count)

39876
```

6.count of insurance_provider

```
insurance_count = df['Insurance Provider'].nunique()
display(insurance_count)

5
```

7. total number of male and female

```
male_count = df[df['Gender'].str.lower() == 'male'].shape[0]
display(male_count)
```

```
27774
```

```
female_count = df[df['Gender'].str.lower() == 'female'].shape[0]
display(female_count)
```

```
27726
```

9. count of medication

```
medication_count = df['Medication'].nunique()
display(medication_count)
```

```
5
```

Metrics

1. Count of Patients by Insurance_ provider

```
#Patients by Insurance Provider
insurance_patient_count = df.groupby('Insurance Provider')['Name'].count().sort_values(ascending=False)
display(insurance_patient_count)
```

Insurance Provider	Name
Cigna	11139
Medicare	11039
UnitedHealthcare	11014
Blue Cross	10952
Aetna	10822

dtype: int64

2. Top 20 Count of patients by doctors

```
#Patients by Doctor
hospital_patient_count = df.groupby('Doctor')['Name'].count().sort_values(ascending=False).head(20)
display(hospital_patient_count)
```

Doctor	Name
Michael Smith	27
John Smith	22
Robert Smith	21
Michael Johnson	20
James Smith	20
David Smith	19
Robert Johnson	19
Michael Williams	18
Christopher Smith	17
Matthew Smith	17
John Johnson	17
Michael Brown	16
Christopher Brown	15
David Johnson	15
Jennifer Johnson	15
Daniel Smith	15
Elizabeth Smith	15
William Johnson	15
Lisa Smith	14
Anthony Smith	14

dtype: int64

3. Count of patients by hospital

```
#Top 20 Hospitals by Patient Count
top_hospitals = df.groupby('Hospital')['Name'].count().sort_values(ascending=False).head(20)
display(top_hospitals)
```

Hospital	Name
LLC Smith	44
Ltd Smith	39
Smith Ltd	37
Johnson PLC	37
Smith PLC	36
Smith Group	36
Johnson Inc	34
Smith Inc	33
Group Smith	32
Smith LLC	32
LLC Johnson	30
PLC Williams	30
PLC Smith	29
Brown Inc	28
Inc Brown	27
Inc Smith	27
Johnson Group	27
Group Johnson	26
Inc Johnson	26
Inc Jones	25

dtype: int64

4. Patients by test results

```
#Patients by Test Results
test_result_patient_count = df.groupby('Test Results')['Name'].count().sort_values(ascending=False)
display(test_result_patient_count)
```

Test Results	Name
Abnormal	18437
Normal	18331
Inconclusive	18198

dtype: int64

5. Patients by admission type

```
#Patients by Admission Type
admission_type_patient_count = df.groupby('Admission Type')['Name'].count().sort_values(ascending=False)
display(admission_type_patient_count)
```

Admission Type	Name
Elective	18473
Urgent	18391
Emergency	18102

dtype: int64

6. Patients by medications

```
#Patients by Medications
medication_patient_count = df.groupby('Medication')['Name'].count().sort_values(ascending=False)
display(medication_patient_count)
```

Medication	Name
Lipitor	11038
Ibuprofen	11023
Aspirin	10984
Paracetamol	10985
Penicillin	10956

dtype: int64

7. Patients by medical conditions

```
#Patients by medical conditions
medical_condition_patient_count = df.groupby('Medical Condition')['Name'].count().sort_values(ascending=False)
display(medical_condition_patient_count)
```

Medical Condition	Name
Arthritis	9218
Diabetes	9216
Hypertension	9151
Obesity	9146
Cancer	9140
Asthma	9095

dtype: int64

```
#Age Distribution by Condition
age_condition_distribution = df.groupby(['Medical Condition', 'Age'])['Name'].count()
display(age_condition_distribution)
```

Medical Condition	Age	Name
Arthritis	13	2
	14	4
	15	2
	16	3
	17	10
...	...	...
Obesity	85	111
	86	7
	87	6
	88	2
	89	1

461 rows × 1 columns

dtype: int64

```python
#Gender vs Condition
gender_condition_distribution = df.groupby(['Gender', 'Medical Condition'])['Name'].count()
display(gender_condition_distribution)
```

Gender	Medical Condition	Name
Female	Arthritis	4642
	Asthma	4511
	Cancer	4566
	Diabetes	4609
	Hypertension	4569
	Obesity	4573
Male	Arthritis	4576
	Asthma	4584
	Cancer	4574
	Diabetes	4607
	Hypertension	4582
	Obesity	4573

dtype: int64

```python
#average admission period
df['Date of Admission'] = pd.to_datetime(df['Date of Admission'])
df['Discharge Date'] = pd.to_datetime(df['Discharge Date'])
df['Admission Period'] = (df['Discharge Date'] - df['Date of Admission']).dt.days
average_admission_period = df['Admission Period'].mean()
display(average_admission_period)
```

```
np.float64(15.499290470472657)
```