Healthcare Dataset Analysis

Data Source Link - [Healthcare Dataset Kaggle Link (https://www.kaggle.com/datasets/prasad22/healthcare-dataset)](https://www.kaggle.com/datasets/prasad22/healthcare-dataset)

Project started on 8th May, 2024.

In [1]:
```
# Importing Library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('darkgrid')

# Showing max columns and rows
pd.set_option("display.max_columns",None)
pd.set_option('display.max_rows',None)
pd.option_context('mode.use_inf_as_na', True)

# Ignore warnings
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
# Importing Dataset
healthcare = pd.read_csv('E:\Project for Resume\Python Healthcare Analysis Project\He
healthcare.head()
```

Out[2]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billi Amou
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.9833
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.0648
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.8969
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.3220
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.3441

In [ ]:
```
```

Checking for Null or blank values from the dataset.

In [3]:
```
# Checking for null values
healthcare.isnull().sum()
```

Out[3]:
```
Name                  0
Age                   0
Gender                0
Blood Type            0
Medical Condition     0
Date of Admission     0
Doctor                0
Hospital              0
Insurance Provider    0
Billing Amount        0
Room Number           0
Admission Type        0
Discharge Date        0
Medication            0
Test Results          0
dtype: int64
```

In [4]:
```
# checking for Blank Values
healthcare.isna().sum()
```

Out[4]:
```
Name                  0
Age                   0
Gender                0
Blood Type            0
Medical Condition     0
Date of Admission     0
Doctor                0
Hospital              0
Insurance Provider    0
Billing Amount        0
Room Number           0
Admission Type        0
Discharge Date        0
Medication            0
Test Results          0
dtype: int64
```

In [ ]:
```

```

Checking for Duplicate values

In [5]:
```
healthcare.duplicated().sum()
```

Out[5]:
```
0
```

In [ ]:
```

```

Transformation of Attributes

In [6]:
```
healthcare.head()
```

Out[6]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billi Amou
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.9833
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.0648
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.8969
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.3220
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.3441

In [7]:
```
healthcare.columns
```

Out[7]:
```
Index(['Name', 'Age', 'Gender', 'Blood Type', 'Medical Condition',
       'Date of Admission', 'Doctor', 'Hospital', 'Insurance Provider',
       'Billing Amount', 'Room Number', 'Admission Type', 'Discharge Date',
       'Medication', 'Test Results'],
      dtype='object')
```

In [8]:
```
healthcare.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   Name                10000 non-null  object 
 1   Age                 10000 non-null  int64  
 2   Gender              10000 non-null  object 
 3   Blood Type          10000 non-null  object 
 4   Medical Condition   10000 non-null  object 
 5   Date of Admission   10000 non-null  object 
 6   Doctor              10000 non-null  object 
 7   Hospital            10000 non-null  object 
 8   Insurance Provider  10000 non-null  object 
 9   Billing Amount      10000 non-null  float64
 10  Room Number         10000 non-null  int64  
 11  Admission Type      10000 non-null  object 
 12  Discharge Date      10000 non-null  object 
 13  Medication          10000 non-null  object 
```

In [9]:
```
healthcare['Date of Admission'] = pd.to_datetime(healthcare['Date of Admission'],form
healthcare.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   Name                10000 non-null  object        
 1   Age                 10000 non-null  int64         
 2   Gender              10000 non-null  object        
 3   Blood Type          10000 non-null  object        
 4   Medical Condition   10000 non-null  object        
 5   Date of Admission   10000 non-null  datetime64[ns]
 6   Doctor              10000 non-null  object        
 7   Hospital            10000 non-null  object        
 8   Insurance Provider  10000 non-null  object        
 9   Billing Amount      10000 non-null  float64       
 10  Room Number         10000 non-null  int64         
 11  Admission Type      10000 non-null  object        
 12  Discharge Date      10000 non-null  object        
 13  Medication          10000 non-null  object        
 14  Test Results        10000 non-null  object        
dtypes: datetime64[ns](1), float64(1), int64(2), object(11)
memory usage: 1.1+ MB
```

In [10]:
```
healthcare['Discharge Date'] = pd.to_datetime(healthcare['Discharge Date'])
healthcare['Discharge Date'].info()
```

```
<class 'pandas.core.series.Series'>
RangeIndex: 10000 entries, 0 to 9999
Series name: Discharge Date
Non-Null Count  Dtype         
--------------  -----         
10000 non-null  datetime64[ns]
dtypes: datetime64[ns](1)
memory usage: 78.3 KB
```

In [11]:
```
healthcare['Billing Amount']=round(healthcare[['Billing Amount']],2)
healthcare.head()
```

Out[11]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	R
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [ ]:

Descriptive Analysis

In [12]:
```
healthcare.describe()
```

Out[12]:

	Age	Date of Admission	Billing Amount	Room Number	Discharge Date
count	10000.000000	10000	10000.000000	10000.000000	10000
mean	51.452200	2021-05-01 21:53:25.439999744	25516.806801	300.082000	2021-05-17 11:22:24.960000
min	18.000000	2018-10-30 00:00:00	1000.180000	101.000000	2018-11-01 00:00:00
25%	35.000000	2020-02-10 00:00:00	13506.522500	199.000000	2020-02-23 18:00:00
50%	52.000000	2021-05-02 00:00:00	25258.115000	299.000000	2021-05-18 00:00:00
75%	68.000000	2022-07-23 06:00:00	37733.917500	400.000000	2022-08-07 00:00:00
max	85.000000	2023-10-30 00:00:00	49995.900000	500.000000	2023-11-27 00:00:00
std	19.588974	NaN	14067.292682	115.806027	NaN

In [ ]:

Solving Questions Generated by ChatGPT.

In [ ]:

Q1. What is the average age of patients in the dataset?

In [13]:
```
healthcare['Age'].mean()
print('The Average age of overall pateints is {}.'.format(round(healthcare['Age'].mea
```

```
The Average age of overall pateints is 51.45.
```

In [14]:
```
avg_age_by_gender = round(healthcare[['Gender','Age']].groupby('Gender')[['Age']].mea
avg_age_by_gender
```

Out[14]:

	Age
Gender	
Female	51.61
Male	51.29

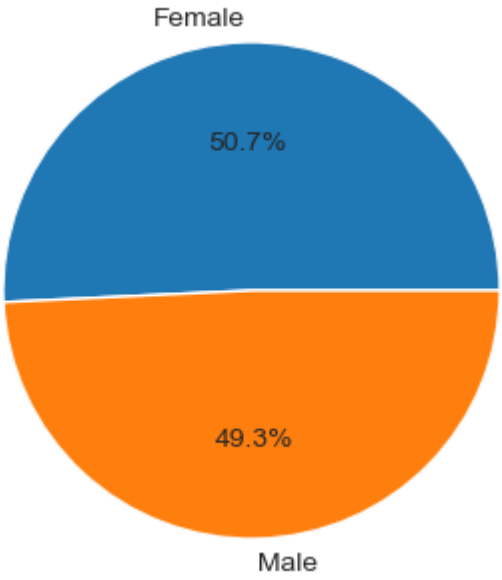
In [15]:
```
gender=pd.DataFrame(healthcare[['Gender']].value_counts())
gender.reset_index(inplace =True)
gender
```

Out[15]:

	Gender	count
0	Female	5075
1	Male	4925

In [ ]:

In [16]:
```
plt.figure(figsize = (4,5))
plt.pie(x=gender['count'],labels=gender['Gender'], autopct='%1.1f%%');
```



In [ ]:

Q2. How many male and female patients are there in the dataset?

In [17]:
```
healthcare.head()
```

Out[17]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [18]:
```
male_female_patients =  healthcare[['Gender']].value_counts()
male_female_patients
```

Out[18]:
```
Gender
Female    5075
Male      4925
Name: count, dtype: int64
```

In [ ]:

Q3. What are the unique blood types present in the dataset?

In [19]:
```
healthcare.head()
```

Out[19]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [20]:
```
healthcare['Blood Type'].unique()
print('Different Blood Types are {}.'.format(healthcare['Blood Type'].unique()))
```

```
Different Blood Types are ['O-' 'O+' 'B-' 'AB+' 'A+' 'AB-' 'A-' 'B+'].
```

In [ ]:

Q4. How many patients were admitted to each hospital?

In [21]:
```
healthcare[['Hospital']].nunique()
```

Out[21]:
```
Hospital    8639
dtype: int64
```

In [22]:
```
# there are 8639 Hospitals , but top 20 hospitals with count of patients are
healthcare[['Hospital','Name']].groupby('Hospital')[['Name']].count().rename(
    columns = {'Name':'Patient Count'}).sort_values('Patient Count',ascending = False
```

Out[22]:

	Hospital	Patient Count
0	Smith PLC	19
1	Smith and Sons	17
2	Smith Ltd	14
3	Smith Inc	14
4	Johnson PLC	13
5	Smith Group	12
6	Williams Inc	12
7	Williams LLC	12
8	Thomas Group	11
9	Johnson Ltd	11
10	Johnson Group	11

In [ ]:
```

```

Q5. What is the most common medical condition among patients?

In [23]:
```
healthcare.head()
```

Out[23]:

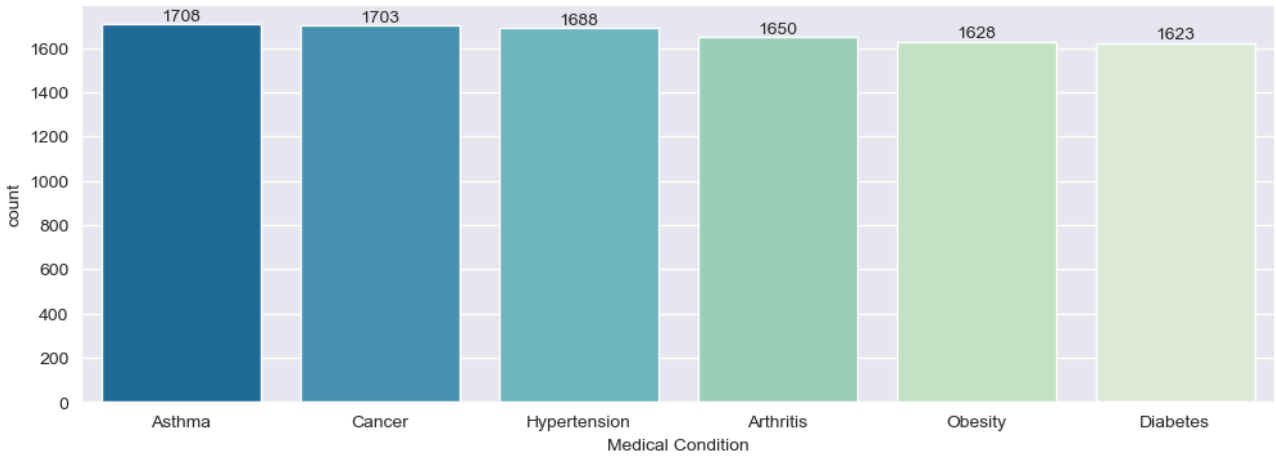
	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [24]:
```
common_medical_patients = healthcare.groupby('Medical Condition')[['Medical Condition
common_medical_patients
```

Out[24]:

Medical Condition	Medical Condition
Arthritis	1650
Asthma	1708
Cancer	1703
Diabetes	1623
Hypertension	1688
Obesity	1628

In [25]:
```
plt.figure(figsize =(12,4))
ax= sns.countplot(data= healthcare,
                  x = 'Medical Condition',
                  palette = 'GnBu_r',
                  order = healthcare['Medical Condition'].value_counts().index)
for bars in ax.containers:
    ax.bar_label(ax.containers[0])
```



In [ ]:
```

```

Q6. How many patients have insurance and how many do not?

In [26]:
```
healthcare.head()
```

Out[26]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34

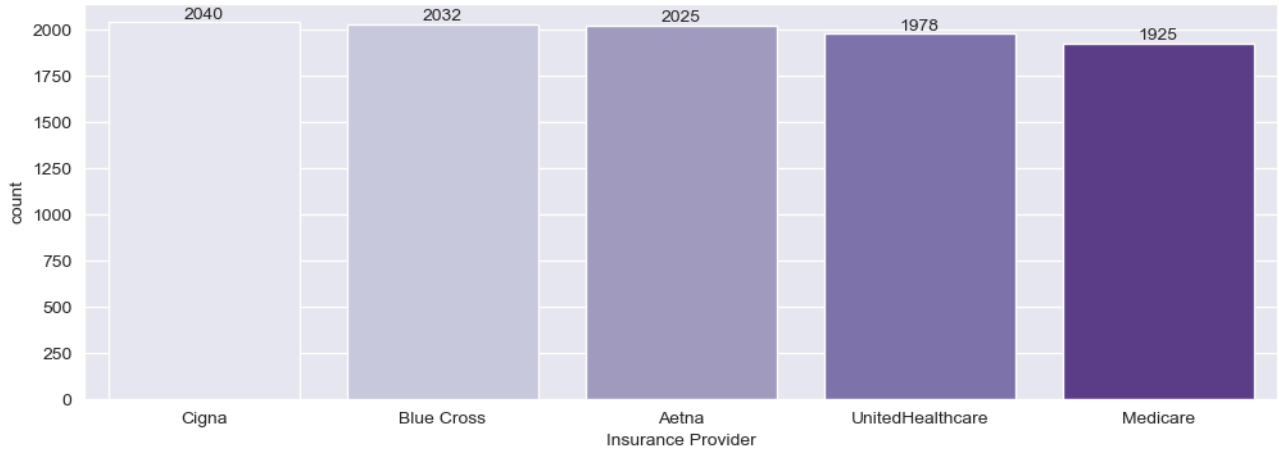
In [27]:
```
healthcare['Insurance Provider'].unique()
```

Out[27]:
```
array(['Medicare', 'UnitedHealthcare', 'Aetna', 'Cigna', 'Blue Cross'],
      dtype=object)
```

In [28]:
```
healthcare['Insurance Provider'].value_counts()
```

Out[28]:
```
Insurance Provider
Cigna               2040
Blue Cross          2032
Aetna               2025
UnitedHealthcare    1978
Medicare            1925
Name: count, dtype: int64
```

In [29]:
```
plt.figure(figsize =(12,4))
ax = sns.countplot(data = healthcare,
                   x='Insurance Provider',
                   palette = 'Purples',
                  order = healthcare['Insurance Provider'].value_counts().index)
for bars in ax.containers:
    ax.bar_label(ax.containers[0]);
```



In [ ]:

Q7. What is the average billing amount for each admission type?

In [30]:
```
healthcare.head()
```

Out[30]:

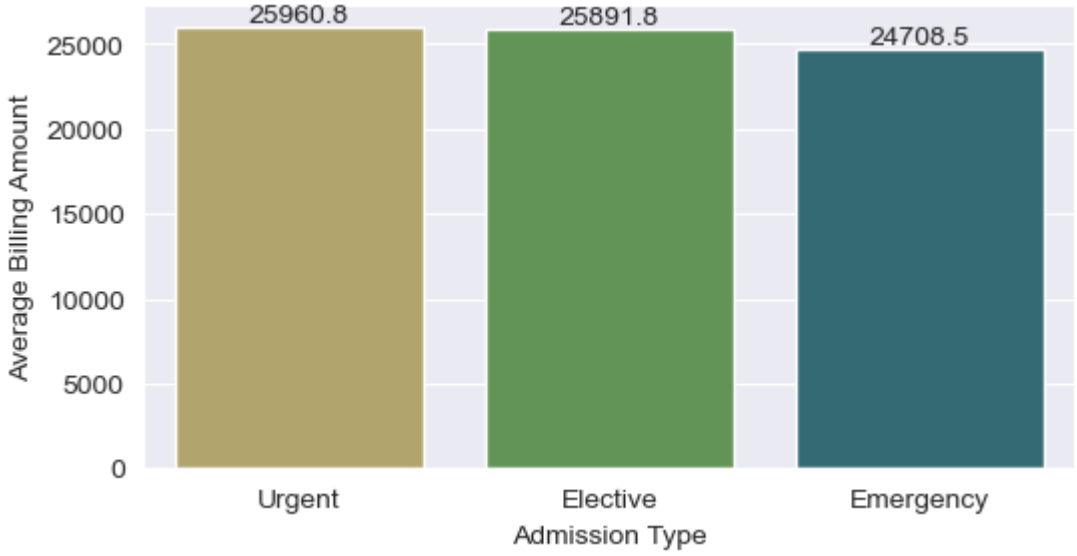
	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [31]:
```
avg_billing_amt = round(healthcare[['Admission Type','Billing Amount']].groupby('Admi
avg_billing_amt.rename(columns={'Billing Amount':'Average Billing Amount'},inplace =
avg_billing_amt = avg_billing_amt.sort_values('Average Billing Amount',ascending = Fa
avg_billing_amt
```

Out[31]:

Admission Type	Average Billing Amount
Urgent	25960.83
Elective	25891.83
Emergency	24708.51

In [32]:
```
plt.figure(figsize =(6,3))

ax = sns.barplot(data = avg_billing_amt ,
                 x = avg_billing_amt.index,
                 y = 'Average Billing Amount',
                 palette = 'gist_earth_r' )
for bars in ax.containers:
    ax.bar_label(bars)
```



In [ ]:

Q8. What is the distribution of room numbers among patients?

In [33]:
```
healthcare.head()
```

Out[33]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [34]:
```
distribution = pd.DataFrame(healthcare[['Name','Room Number']].value_counts() )
distribution
```

Out[34]:

		count
Name	**Room Number**	
Michael Miller	**271**	2
Jeremy Johnson	**404**	2
Meghan Burns	**430**	1
Meghan Horne DVM	**377**	1
Meghan Jordan	**479**	1
Meghan Lee	**265**	1
Meghan Robinson	**126**	1
Melanie Alvarado	**465**	1
Melanie Berger	**403**	1
Melanie Best	**327**	1

In [35]:
```
distribution[distribution['count']>=2]
```

Out[35]:

		count
Name	**Room Number**	
Michael Miller	**271**	2
Jeremy Johnson	**404**	2

In [ ]:
```

```

Q9.Which doctor treated the most number of patients?

In [36]:
```
healthcare.head()
```

Out[36]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	R
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [37]:
```
doctor = pd.DataFrame(healthcare[['Doctor','Name']].groupby('Doctor')['Name'].count()
doctor.rename(columns={'Name':'Counts'},inplace= True)
doctor = doctor[doctor['Counts']>=2].sort_values('Counts',ascending = False)
doctor
```

Out[37]:

Doctor	Counts
Michael Johnson	7
Jennifer Smith	5
Michelle Anderson	5
Michael Smith	5
Robert Brown	5
James Perez	5
James Williams	5
Matthew Smith	5
David Johnson	4
Robert Miller	4

In [38]:
```
doctor[(doctor['Counts']>=2) & (doctor['Counts']<=7)]
```

Out[38]:

Doctor	Counts
Michael Johnson	7
Jennifer Smith	5
Michelle Anderson	5
Michael Smith	5
Robert Brown	5
James Perez	5
James Williams	5
Matthew Smith	5
David Johnson	4
Robert Miller	4

In [ ]:
```
```

Q10. How many patients were admitted each month?

In [39]:
```
healthcare.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   Name                10000 non-null  object        
 1   Age                 10000 non-null  int64         
 2   Gender              10000 non-null  object        
 3   Blood Type          10000 non-null  object        
 4   Medical Condition   10000 non-null  object        
 5   Date of Admission   10000 non-null  datetime64[ns]
 6   Doctor              10000 non-null  object        
 7   Hospital            10000 non-null  object        
 8   Insurance Provider  10000 non-null  object        
 9   Billing Amount      10000 non-null  float64       
 10  Room Number         10000 non-null  int64         
 11  Admission Type      10000 non-null  object        
 12  Discharge Date      10000 non-null  datetime64[ns]
 13  Medication          10000 non-null  object        
 14  Test Results        10000 non-null  object        
dtypes: datetime64[ns](2), float64(1), int64(2), object(10)
memory usage: 1.1+ MB
```

In [40]:
```
healthcare['doa_month'] = healthcare['Date of Admission'].dt.month
healthcare['doa_year'] = healthcare['Date of Admission'].dt.year
healthcare.info()
```

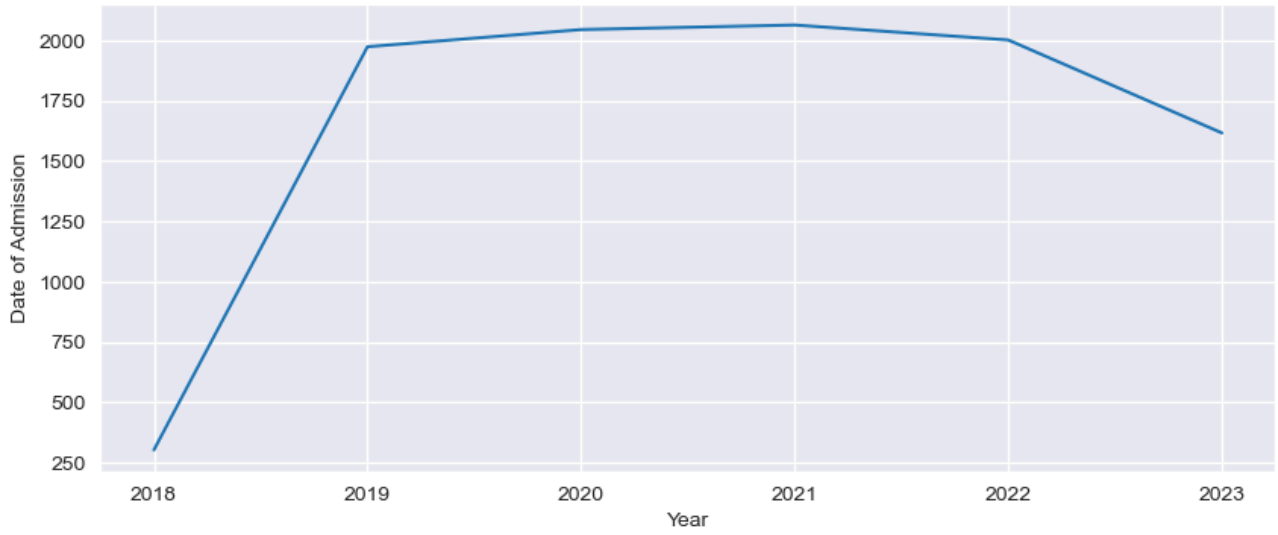
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 17 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   Name                10000 non-null  object        
 1   Age                 10000 non-null  int64         
 2   Gender              10000 non-null  object        
 3   Blood Type          10000 non-null  object        
 4   Medical Condition   10000 non-null  object        
 5   Date of Admission   10000 non-null  datetime64[ns]
 6   Doctor              10000 non-null  object        
 7   Hospital            10000 non-null  object        
 8   Insurance Provider  10000 non-null  object        
 9   Billing Amount      10000 non-null  float64       
 10  Room Number         10000 non-null  int64         
 11  Admission Type      10000 non-null  object        
 12  Discharge Date      10000 non-null  datetime64[ns]
 13  Medication          10000 non-null  object        
 14  Test Results        10000 non-null  object        
 15  doa_month           10000 non-null  int32         
 16  doa_year            10000 non-null  int32         
dtypes: datetime64[ns](2), float64(1), int32(2), int64(2), object(10)
memory usage: 1.2+ MB
```

In [41]:
```
admission_by_year = healthcare[['Date of Admission','doa_year']].groupby(['doa_year']
admission_by_year
```

Out[41]:

doa_year	Date of Admission
2018	303
2019	1973
2020	2044
2021	2063
2022	2001
2023	1616

In [42]:
```
plt.figure(figsize=(10,4))
sns.lineplot(data = admission_by_year,y='Date of Admission',x='doa_year')
plt.xlabel('Year');
```



In [43]:
```
year = healthcare[['Date of Admission','doa_year','doa_month']].groupby(['doa_year','
year.reset_index(inplace = True)
year.rename(columns={'Date of Admission': 'Counts'},inplace = True)
year
```

Out[43]:

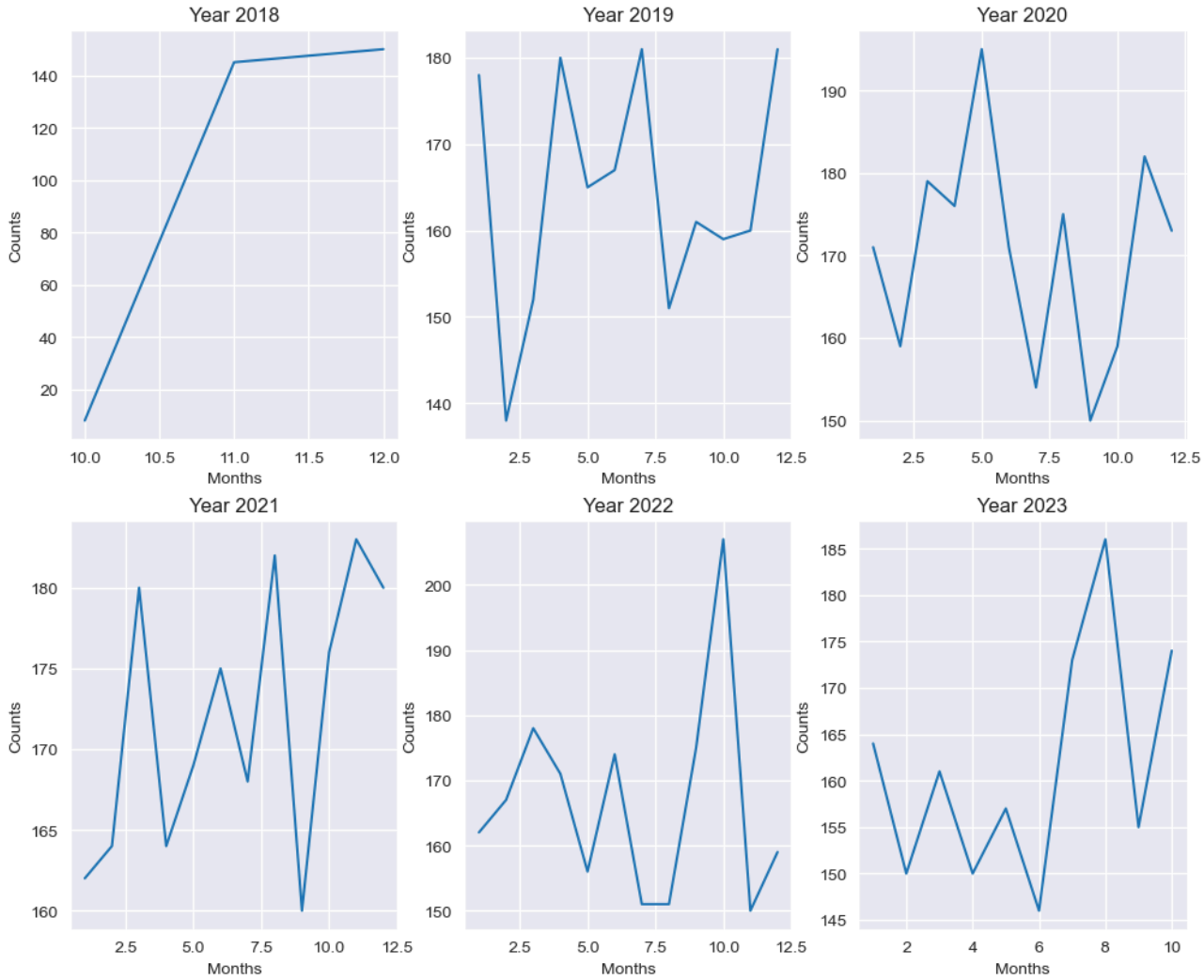
	doa_year	doa_month	Counts
0	2018	10	8
1	2018	11	145
2	2018	12	150
3	2019	1	178
4	2019	2	138
5	2019	3	152
6	2019	4	180
7	2019	5	165
8	2019	6	167
9	2019	7	181
10	2019	8	151
11	2019	9	161
12	2019	10	159
13	2019	11	160
14	2019	12	181
15	2020	1	171
16	2020	2	159
17	2020	3	179
18	2020	4	176
19	2020	5	195
20	2020	6	171
21	2020	7	154
22	2020	8	175
23	2020	9	150
24	2020	10	159
25	2020	11	182
26	2020	12	173
27	2021	1	162
28	2021	2	164
29	2021	3	180
30	2021	4	164
31	2021	5	169
32	2021	6	175
33	2021	7	168
34	2021	8	182
35	2021	9	160
36	2021	10	176
37	2021	11	183
38	2021	12	180
39	2022	1	162
40	2022	2	167

	doa_year	doa_month	Counts
41	2022	3	178
42	2022	4	171
43	2022	5	156
44	2022	6	174
45	2022	7	151
46	2022	8	151
47	2022	9	175
48	2022	10	207
49	2022	11	150
50	2022	12	159
51	2023	1	164
52	2023	2	150
53	2023	3	161
54	2023	4	150
55	2023	5	157
56	2023	6	146
57	2023	7	173
58	2023	8	186
59	2023	9	155
60	2023	10	174

In [44]:
```
year_2018 = year[year['doa_year'] == 2018]

year_2019 = year[year['doa_year'] == 2019]

year_2020 = year[year['doa_year'] == 2020]

year_2021 = year[year['doa_year'] == 2021]

year_2022 = year[year['doa_year'] == 2022]

year_2023 = year[year['doa_year'] == 2023]
```

In [ ]:
```

```

In [45]:
```
# creating subplots
plt.figure(figsize=(12,15))

plt.subplot(3,3,1)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2018')
sns.lineplot(data = year_2018,x='doa_month', y='Counts');

plt.subplot(3,3,2)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2019')
sns.lineplot(data = year_2019,x='doa_month', y='Counts');

plt.subplot(3,3,3)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2020')
sns.lineplot(data = year_2020,x='doa_month', y='Counts');

plt.subplot(3,3,4)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2021')
sns.lineplot(data = year_2021,x='doa_month', y='Counts');

plt.subplot(3,3,5)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2022')
sns.lineplot(data = year_2022,x='doa_month', y='Counts');

plt.subplot(3,3,6)
plt.xlabel('Months')
plt.ylabel('Counts')
plt.title('Year 2023')
sns.lineplot(data = year_2023,x='doa_month', y='Counts');
```



In [ ]:

Q11. Is there any correlation between age and billing amount?

In [46]:
```
healthcare.head()
```

Out[46]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [47]:
```
healthcare.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 17 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   Name                10000 non-null  object        
 1   Age                 10000 non-null  int64         
 2   Gender              10000 non-null  object        
 3   Blood Type          10000 non-null  object        
 4   Medical Condition   10000 non-null  object        
 5   Date of Admission   10000 non-null  datetime64[ns]
 6   Doctor              10000 non-null  object        
 7   Hospital            10000 non-null  object        
 8   Insurance Provider  10000 non-null  object        
 9   Billing Amount      10000 non-null  float64       
 10  Room Number         10000 non-null  int64         
 11  Admission Type      10000 non-null  object        
 12  Discharge Date      10000 non-null  datetime64[ns]
 13  Medication          10000 non-null  object        
 14  Test Results        10000 non-null  object        
 15  doa_month           10000 non-null  int32         
 16  doa_year            10000 non-null  int32         
dtypes: datetime64[ns](2), float64(1), int32(2), int64(2), object(10)
memory usage: 1.2+ MB
```

In [48]:
```
Age_Billing_Amount_corr = healthcare['Age'].corr(healthcare['Billing Amount'])
print('Patients Age And Billing Amount has correlation of {}'.format(Age_Billing_Amou
```

```
Patients Age And Billing Amount has correlation of -0.009483331387154738
```

In [ ]:
```

```

Q12. What is the average length of stay in the hospital for patients?

In [49]:
```
healthcare.head()
```

Out[49]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [50]:
```
average_stay_hospital = (healthcare['Discharge Date'] - healthcare['Date of Admission
print('Average Stay of patients in Hospital is {}.'.format(round(average_stay_hospita
```

```
Average Stay of patients in Hospital is 16.
```

In [ ]:
```

```

13. Which gender tends to have a higher billing amount on average?

In [51]:
```
healthcare.head()
```

Out[51]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [52]:
```
avg_amt_paid_by_gender = round(healthcare[['Gender','Billing Amount']].groupby('Gende
avg_amt_paid_by_gender.rename(columns={'Billing Amount':'Average Billing amount'},inp
avg_amt_paid_by_gender
```

Out[52]:

Gender	Average Billing amount
Female	25484.39
Male	25550.22

Male Patients have the Highest average billing Amount that is 25,550.

In [ ]:
```

```

Q14. Are there any trends in admission types over the years?

In [53]:
```
healthcare.head()
```

Out[53]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [54]:
```
admission_type_trend = healthcare[['Admission Type','doa_year']].groupby(['Admission
admission_type_trend.rename(columns={'Admission Type':'Counts'},inplace = True)
admission_type_trend.reset_index(inplace = True)
admission_type_trend
```

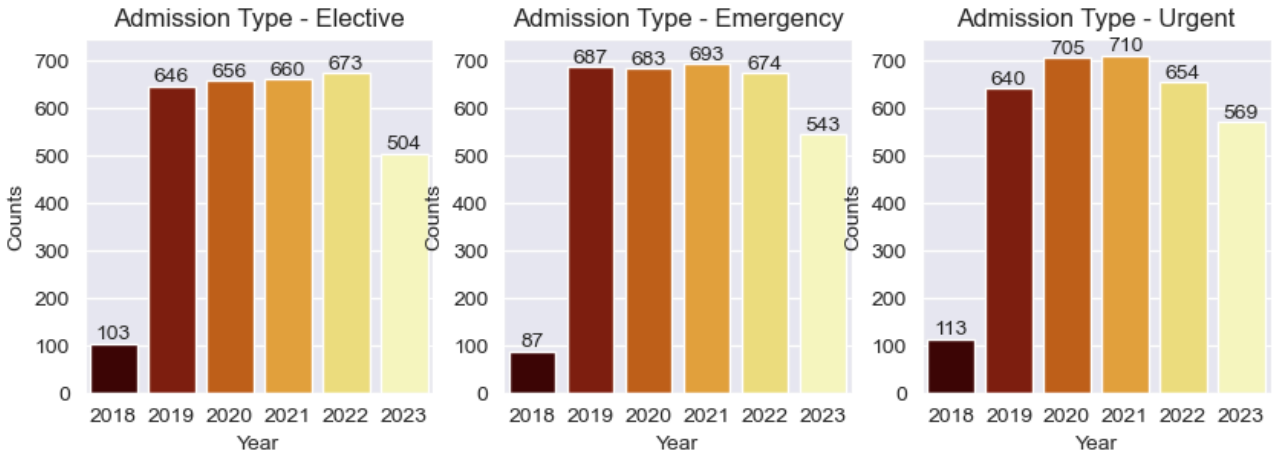
Out[54]:

	Admission Type	doa_year	Counts
0	Elective	2018	103
1	Elective	2019	646
2	Elective	2020	656
3	Elective	2021	660
4	Elective	2022	673
5	Elective	2023	504
6	Emergency	2018	87
7	Emergency	2019	687
8	Emergency	2020	683
9	Emergency	2021	693
10	Emergency	2022	674
11	Emergency	2023	543
12	Urgent	2018	113
13	Urgent	2019	640
14	Urgent	2020	705
15	Urgent	2021	710
16	Urgent	2022	654
17	Urgent	2023	569

In [55]:
```
elective = admission_type_trend[admission_type_trend['Admission Type'] == 'Elective']

emergency = admission_type_trend[admission_type_trend['Admission Type'] == 'Emergency

urgent = admission_type_trend[admission_type_trend['Admission Type'] == 'Urgent']
```

In [ ]:
```
```

In [56]:
```
plt.figure(figsize=(10,3))
plt.tight_layout()

ax1 = plt.subplot(1,3,1)
ax = sns.barplot(data = elective,x = 'doa_year', y = 'Counts', palette = 'afmhot')
for bars in ax.containers:
    ax.bar_label(ax.containers[0])
plt.xlabel('Year')
plt.ylabel('Counts')
plt.title('Admission Type - Elective');

ax2 = plt.subplot(1,3,2,sharey=ax1)
ax = sns.barplot(data = emergency,x = 'doa_year', y = 'Counts', palette = 'afmhot')
for bars in ax.containers:
    ax.bar_label(ax.containers[0])
plt.xlabel('Year')
plt.ylabel('Counts')
plt.title('Admission Type - Emergency');

ax3 = plt.subplot(1,3,3,sharey =ax1)
ax = sns.barplot(data = urgent,x = 'doa_year', y = 'Counts', palette = 'afmhot')
for bars in ax.containers:
    ax.bar_label(ax.containers[0])
plt.xlabel('Year')
plt.ylabel('Counts')
plt.title('Admission Type - Urgent');
```



In [ ]:
```
```

Q15. How does the billing amount vary across different medical conditions?

In [57]:
```
healthcare.head()
```

Out[57]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [58]:
```
healthcare.groupby('Medical Condition')['Billing Amount'].describe()
```

Out[58]:

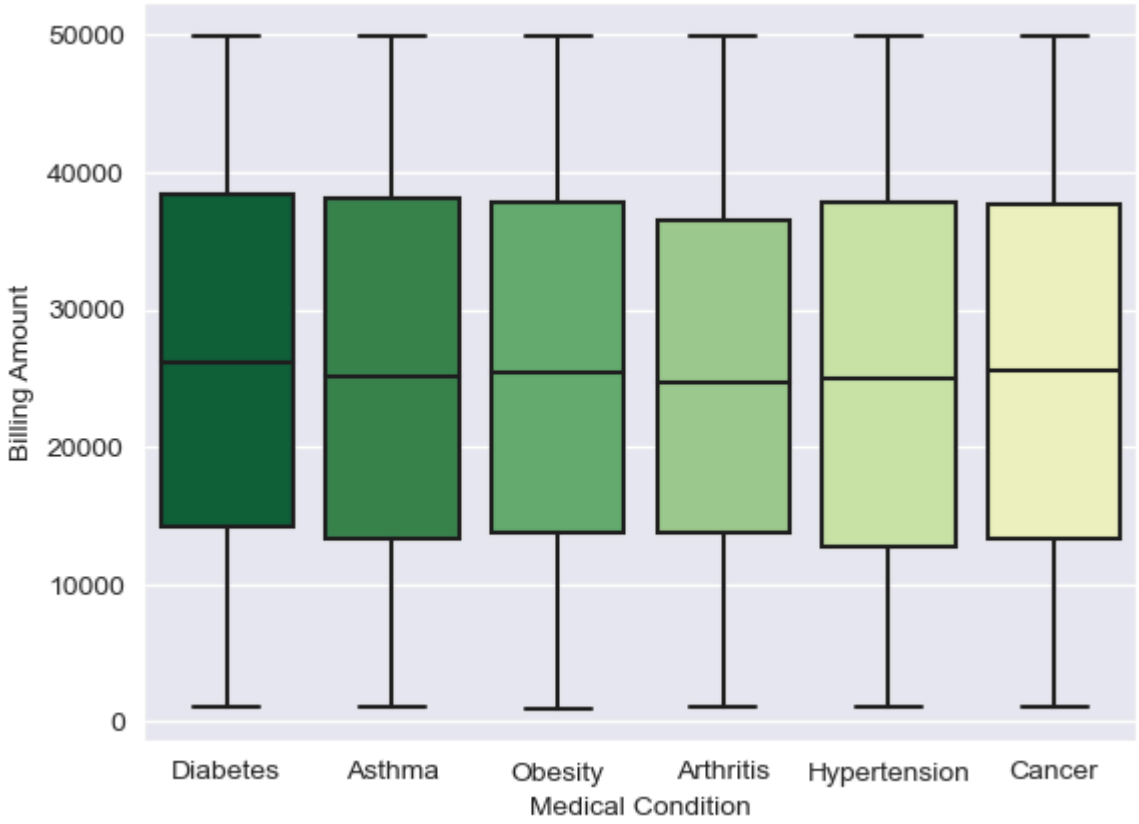
Medical Condition	count	mean	std	min	25%	50%	75%	max
Arthritis	1650.0	25187.631230	13765.174836	1009.42	13754.4250	24739.165	36579.1225	49985.97
Asthma	1708.0	25416.869813	14346.783976	1032.26	13252.5075	25073.450	38069.7575	49974.30
Cancer	1703.0	25539.096142	14081.988383	1020.34	13345.0900	25610.640	37712.8800	49994.98
Diabetes	1623.0	26060.116106	14013.920462	1071.46	14261.6300	26162.200	38420.8600	49954.97
Hypertension	1688.0	25198.034070	14137.135459	1084.42	12710.1650	24920.455	37773.1350	49995.90
Obesity	1628.0	25720.842850	14040.788115	1000.18	13784.8500	25365.025	37751.7150	49974.16

In [59]:
```
avg_billing_amt = round(healthcare.groupby('Medical Condition')[['Billing Amount']].m
    columns={'Billing Amount': 'Average Billing Amount'}).sort_values('Average Billin
avg_billing_amt
```

Out[59]:

	Medical Condition	Average Billing Amount
0	Diabetes	26060.12
1	Obesity	25720.84
2	Cancer	25539.10
3	Asthma	25416.87
4	Hypertension	25198.03
5	Arthritis	25187.63

In [60]:
```
ax = sns.boxplot(data = healthcare, x = 'Medical Condition', y = 'Billing Amount',pal
```



In [ ]:
```

```

Q16. Are there any patterns in medication prescriptions based on medical conditions?

In [61]:
```
healthcare.head()
```

Out[61]:

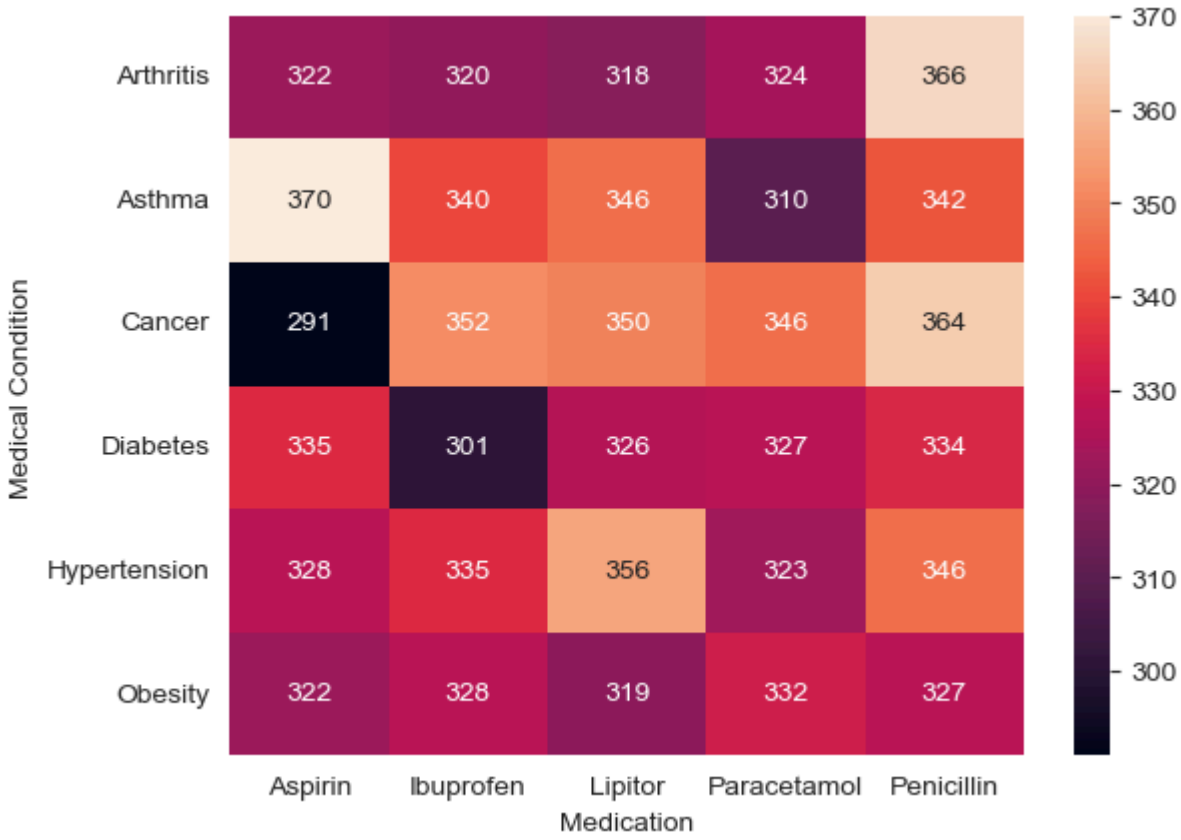
	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [62]:
```
medication_pattern = healthcare.pivot_table(index ='Medical Condition',columns = 'Med
medication_pattern
```

Out[62]:

Medication	Aspirin	Ibuprofen	Lipitor	Paracetamol	Penicillin
Medical Condition					
Arthritis	322	320	318	324	366
Asthma	370	340	346	310	342
Cancer	291	352	350	346	364
Diabetes	335	301	326	327	334
Hypertension	328	335	356	323	346
Obesity	322	328	319	332	327

In [63]:
```
sns.heatmap(medication_pattern, annot = True,fmt ='g' );
```



In [ ]:
```
```

Q17. What is the distribution of test results among patients?

In [64]:
```
healthcare.head()
```

Out[64]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [65]:
```
pd.DataFrame(healthcare['Test Results'].value_counts()).rename(columns = {'count':'Co
```

Out[65]:

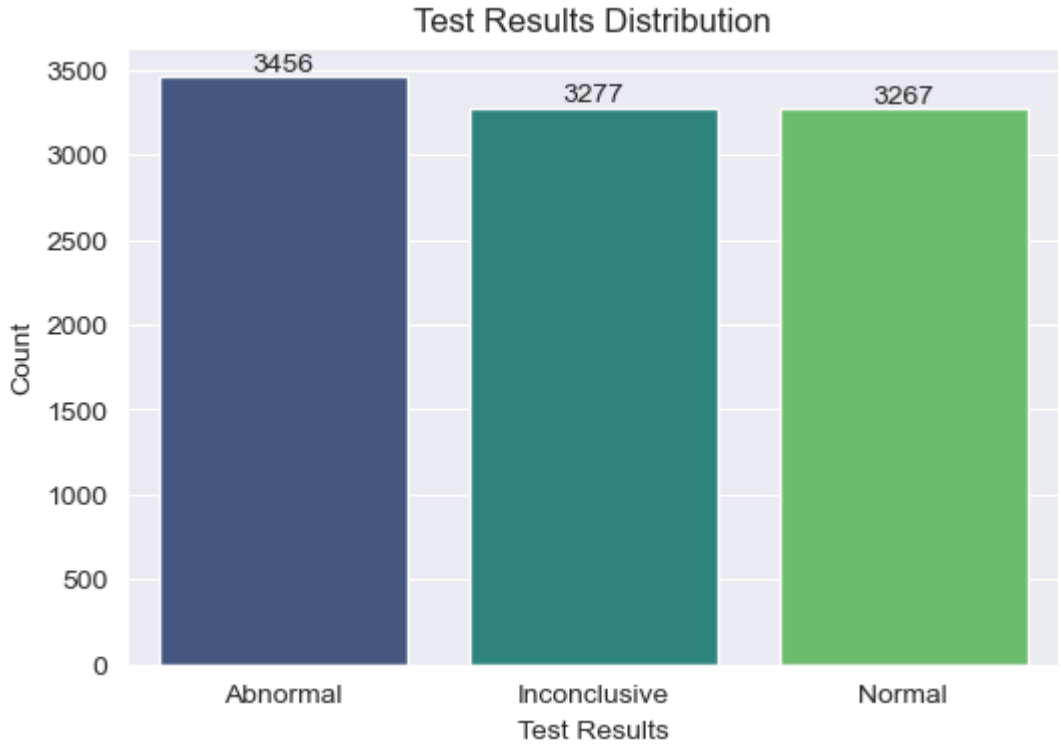
Test Results	Counts
Abnormal	3456
Inconclusive	3277
Normal	3267

In [66]:
```
test_results_distribution = healthcare[['Test Results','Name']].groupby('Test Results
test_results_distribution
```

Out[66]:

Test Results	Count
Abnormal	3456
Inconclusive	3277
Normal	3267

In [67]:
```
plt.figure(figsize = (6,4))
plt.title('Test Results Distribution')
plt.xlabel('Test Results')
ax = sns.barplot(data = test_results_distribution, x = test_results_distribution.inde
for bars in ax.containers:
    ax.bar_label(bars)
```



In [ ]:
```
```

Q18. Is there a relationship between the length of stay and medication prescribed?

In [68]:
```
healthcare.head()
```

Out[68]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [69]:
```
healthcare['Days in Hospital'] = (healthcare['Discharge Date'] - healthcare['Date of
```

In [70]:
```
healthcare[['Days in Hospital','Medication']].groupby('Medication')[['Days in Hospita
```

Out[70]:

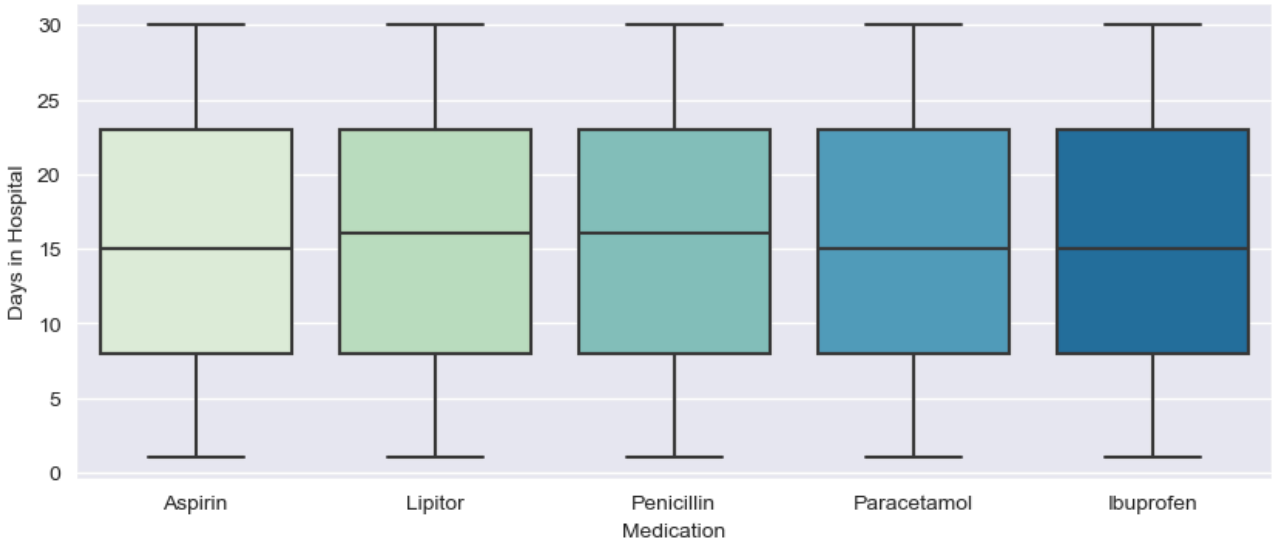
Medication	Days in Hospital
Aspirin	15.447663
Ibuprofen	15.626012
Lipitor	15.422829
Paracetamol	15.549439
Penicillin	15.755171

In [71]:
```
healthcare[['Days in Hospital','Medication']].groupby('Medication')[['Days in Hospita
```

Out[71]:

	Days in Hospital							
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
Medication								
Aspirin	1968.0	15.447663	8.627836	1.0	8.0	15.0	23.0	30.0
Ibuprofen	1976.0	15.626012	8.652698	1.0	8.0	15.0	23.0	30.0
Lipitor	2015.0	15.422829	8.633200	1.0	8.0	16.0	23.0	30.0
Paracetamol	1962.0	15.549439	8.517364	1.0	8.0	15.0	23.0	30.0
Penicillin	2079.0	15.755171	8.630708	1.0	8.0	16.0	23.0	30.0

In [72]:
```
plt.figure(figsize=(10,4))
sns.boxplot(data = healthcare, x = 'Medication', y='Days in Hospital', palette = 'GnB
```



In [ ]:
```

```

Q19. How does the billing amount vary between different insurance providers?

In [73]:
```
healthcare.head()
```

Out[73]:

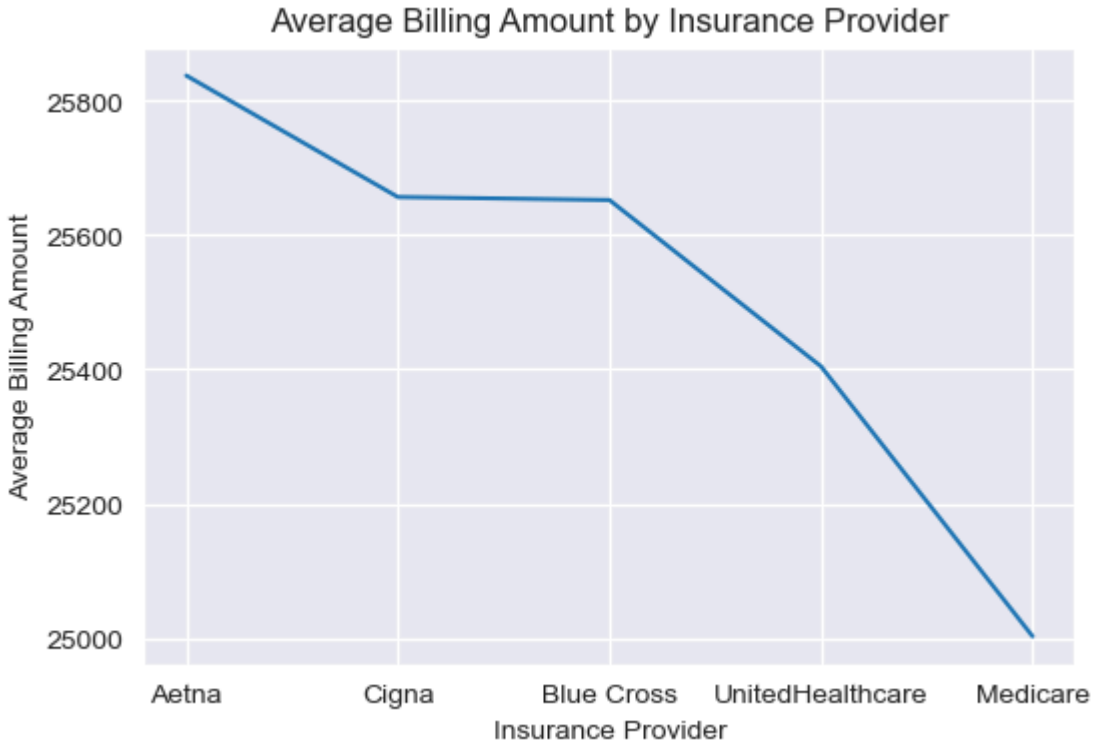
	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [74]:
```
Insurance_Provider_avg_Billing_amt = round(healthcare[['Insurance Provider','Billing
Insurance_Provider_avg_Billing_amt
```

Out[74]:

Insurance Provider	Average Billing Amount
Aetna	25837.92
Cigna	25656.95
Blue Cross	25652.49
UnitedHealthcare	25404.69
Medicare	25002.48

In [75]:
```
plt.figure(figsize=(6,4))
plt.title('Average Billing Amount by Insurance Provider')
sns.lineplot(data= Insurance_Provider_avg_Billing_amt,
             x =Insurance_Provider_avg_Billing_amt.index,
             y = 'Average Billing Amount');
```



In [ ]:
```
```

Q20. Are there any seasonal trends in hospital admissions?

In [76]:
```
healthcare.head()
```

Out[76]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	I
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [77]:
```
seasonal_trend_2018 = healthcare[healthcare['doa_year']==2018]
seasonal_trend_2018 = seasonal_trend_2018[['doa_month','Name']].groupby('doa_month')[
seasonal_trend_2018
```

Out[77]:

	doa_month	Patient_count
0	10	8
1	11	145
2	12	150

In [78]:
```
seasonal_trend_2019 = healthcare[healthcare['doa_year']==2019]
seasonal_trend_2019 = seasonal_trend_2019[['doa_month','Name']].groupby('doa_month')[
seasonal_trend_2019
```

Out[78]:

	doa_month	Patient_count
0	1	178
1	2	138
2	3	152
3	4	180
4	5	165
5	6	167
6	7	181
7	8	151
8	9	161
9	10	159
10	11	160
11	12	181

In [79]:
```
seasonal_trend_2020 = healthcare[healthcare['doa_year']==2020]
seasonal_trend_2020 = seasonal_trend_2020[['doa_month','Name']].groupby('doa_month')[
seasonal_trend_2020
```

Out[79]:

	doa_month	Patient_count
0	1	171
1	2	159
2	3	179
3	4	176
4	5	195
5	6	171
6	7	154
7	8	175
8	9	150
9	10	159
10	11	182
11	12	173

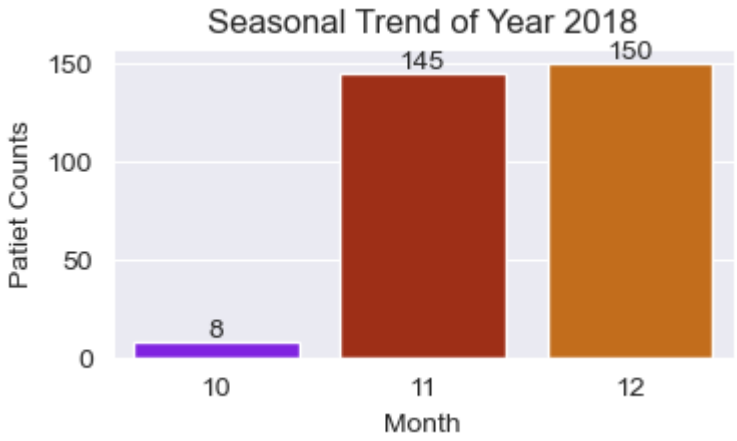
In [80]:
```
seasonal_trend_2021 = healthcare[healthcare['doa_year']==2021]
seasonal_trend_2021 = seasonal_trend_2021[['doa_month','Name']].groupby('doa_month')[
```

In [81]:
```
seasonal_trend_2022 = healthcare[healthcare['doa_year']==2022]
seasonal_trend_2022 = seasonal_trend_2022[['doa_month','Name']].groupby('doa_month')[
```

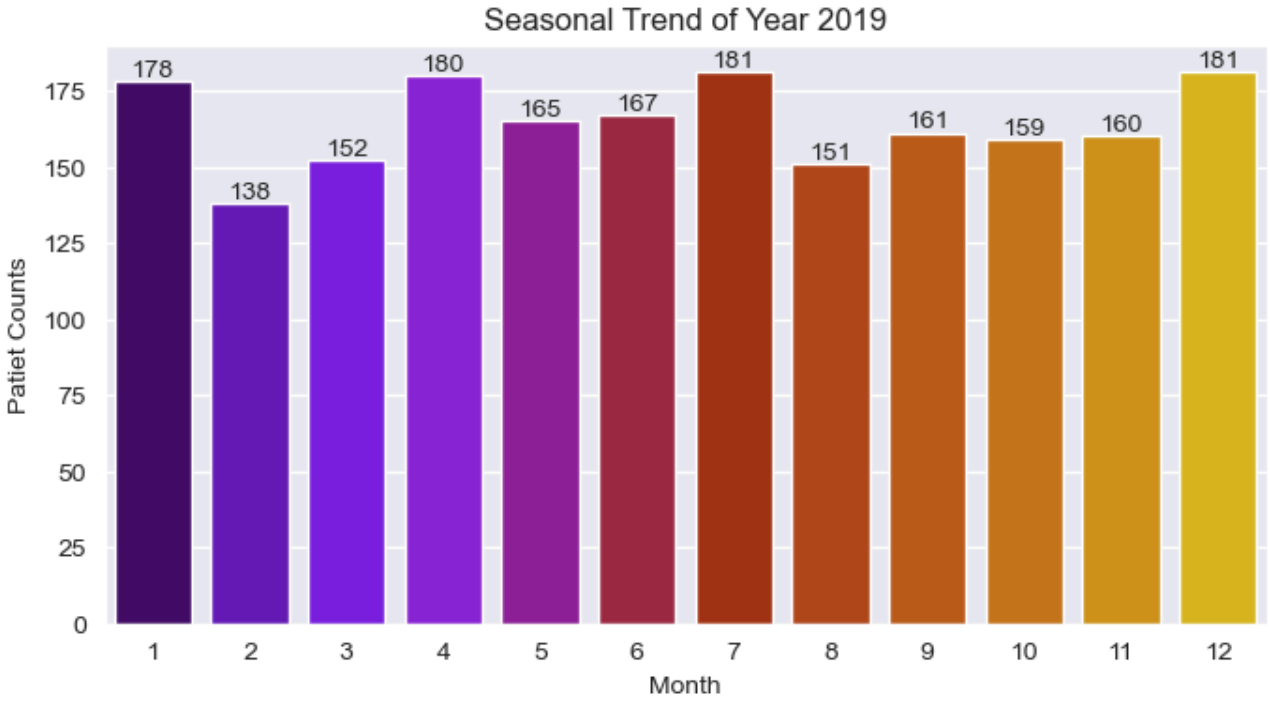
In [82]:
```
seasonal_trend_2023 = healthcare[healthcare['doa_year']==2023]
seasonal_trend_2023 = seasonal_trend_2023[['doa_month','Name']].groupby('doa_month')[
```

In [ ]:
```
```

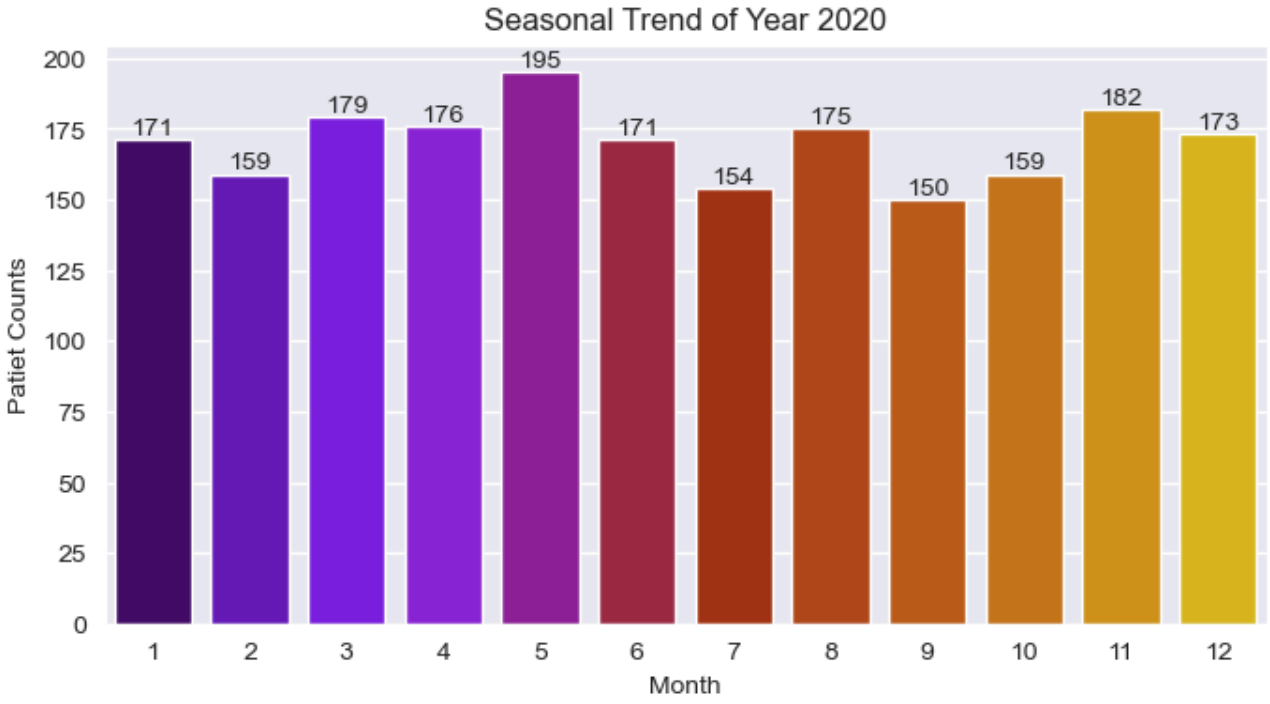
In [83]:
```
plt.figure(figsize=(4,2))
plt.title('Seasonal Trend of Year 2018')
ax = sns.barplot(data= seasonal_trend_2018, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



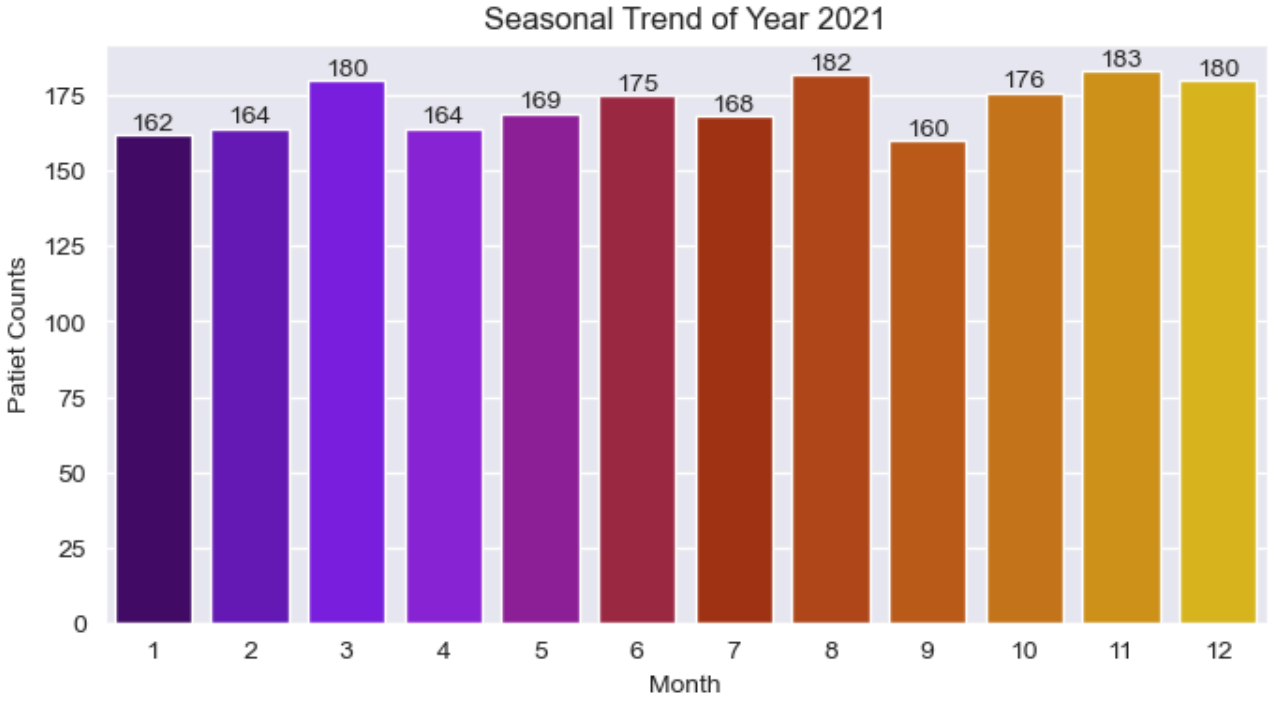
In [84]:
```
plt.figure(figsize=(8,4))
plt.title('Seasonal Trend of Year 2019')
ax = sns.barplot(data= seasonal_trend_2019, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



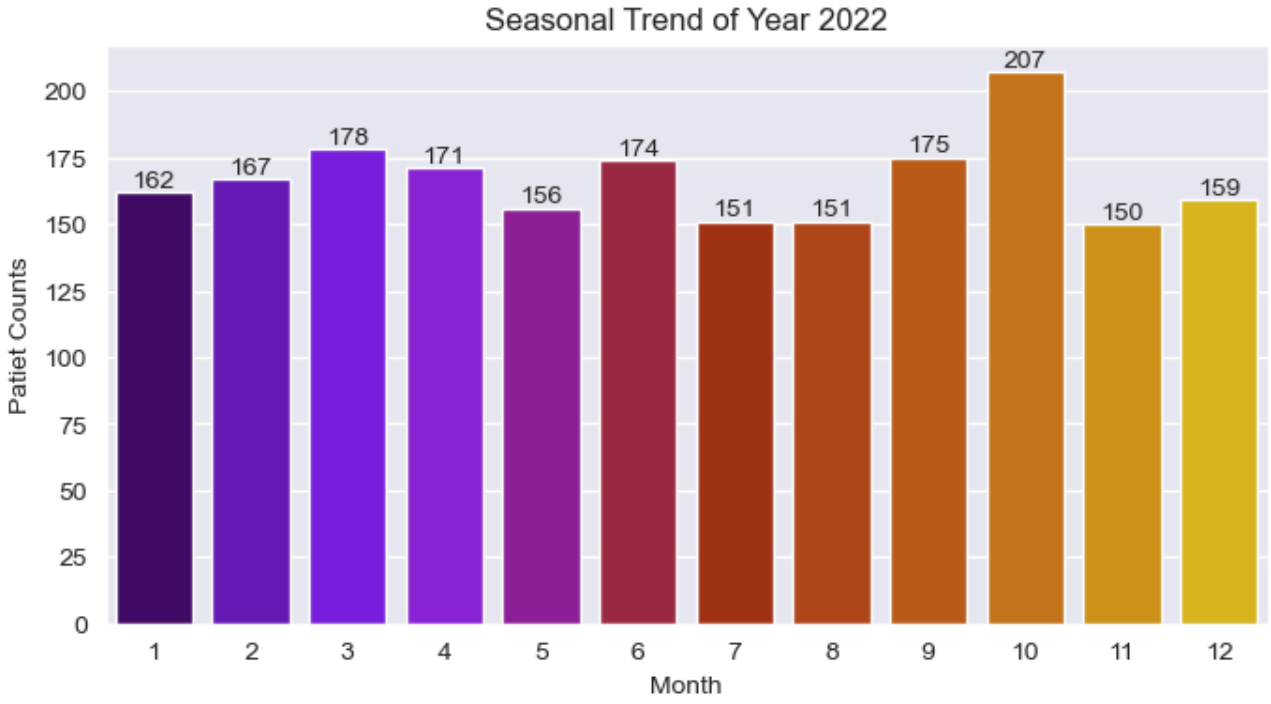
In [85]:
```
plt.figure(figsize=(8,4))
plt.title('Seasonal Trend of Year 2020')
ax = sns.barplot(data= seasonal_trend_2020, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



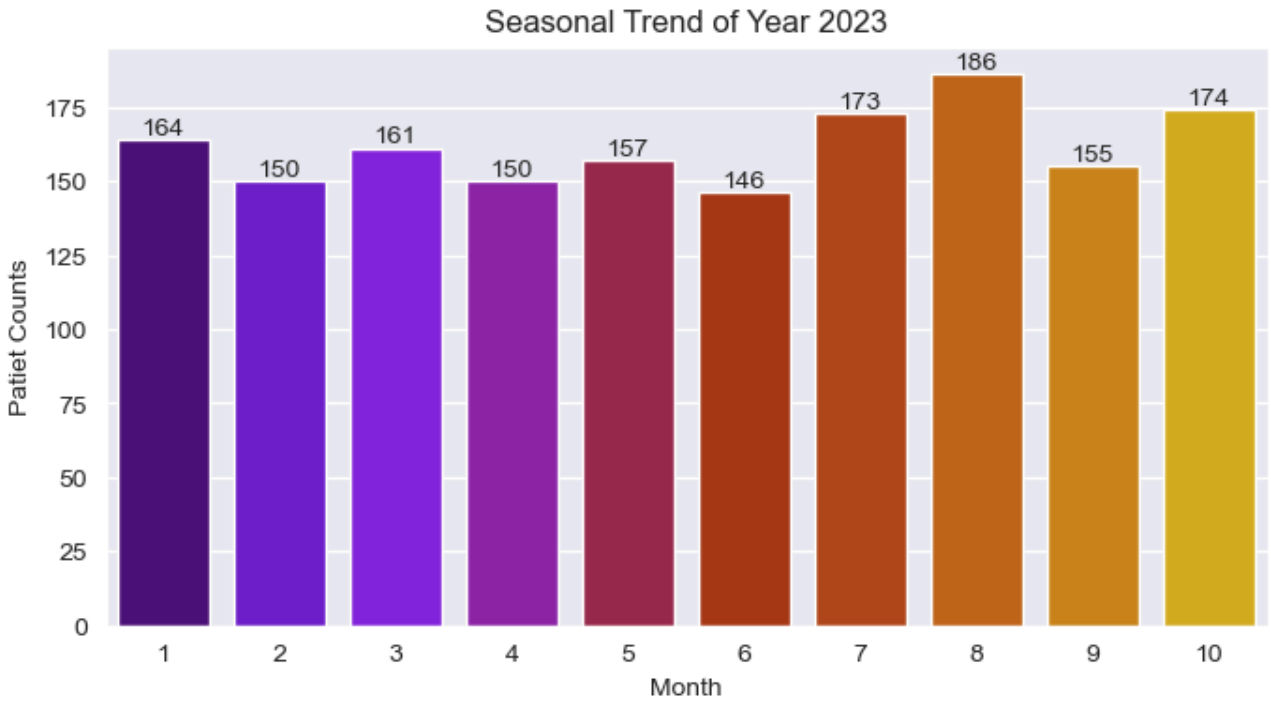
In [86]:
```
plt.figure(figsize=(8,4))
plt.title('Seasonal Trend of Year 2021')
ax = sns.barplot(data= seasonal_trend_2021, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



In [87]:
```
plt.figure(figsize=(8,4))
plt.title('Seasonal Trend of Year 2022')
ax = sns.barplot(data= seasonal_trend_2022, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```

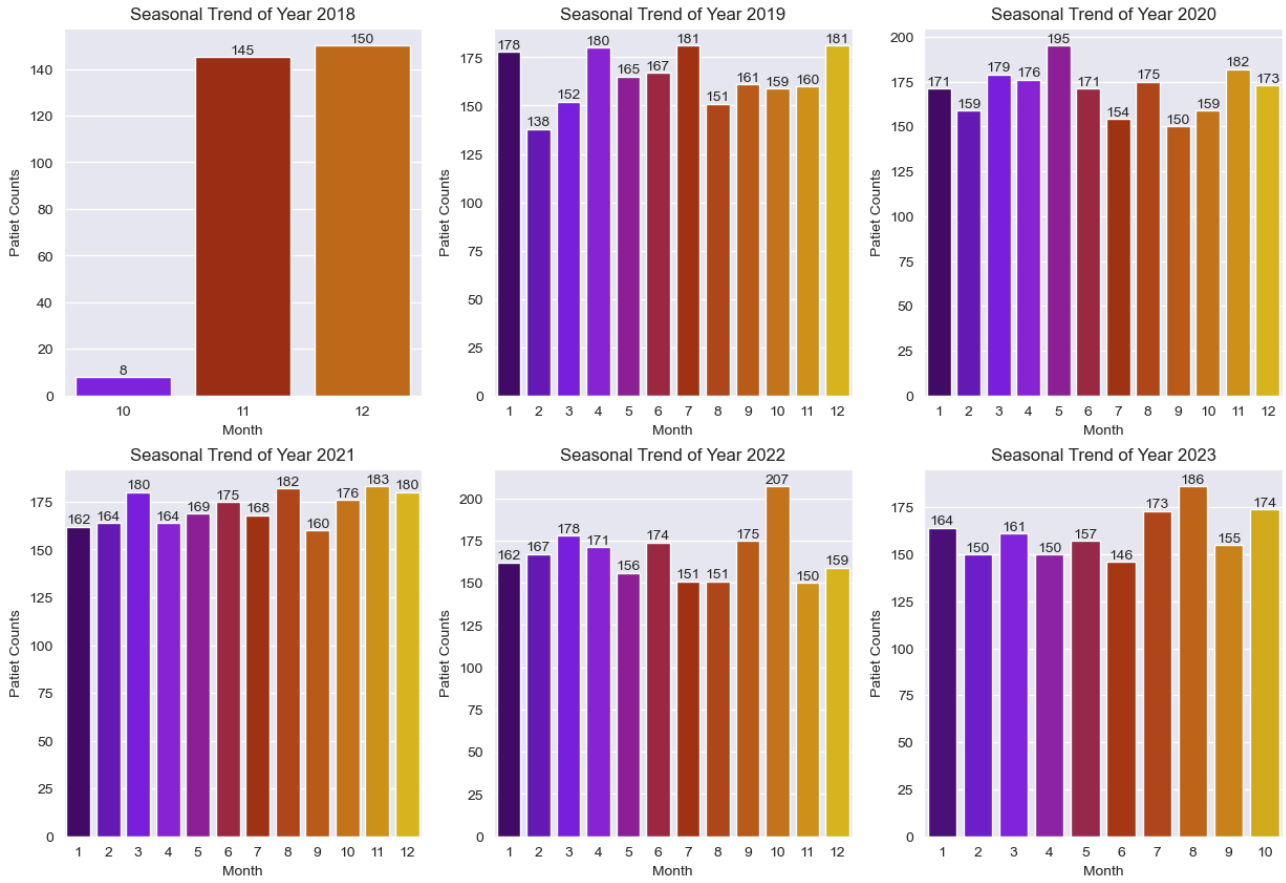


In [88]:
```
plt.figure(figsize=(8,4))
plt.title('Seasonal Trend of Year 2023')
ax = sns.barplot(data= seasonal_trend_2023, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



In [ ]:

In [89]:
```
plt.figure(figsize=(15,10))


plt.subplot(2,3,1)
plt.title('Seasonal Trend of Year 2018')
ax = sns.barplot(data= seasonal_trend_2018, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month')

plt.subplot(2,3,2)
plt.title('Seasonal Trend of Year 2019')
ax = sns.barplot(data= seasonal_trend_2019, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month')

plt.subplot(2,3,3)
plt.title('Seasonal Trend of Year 2020')
ax = sns.barplot(data= seasonal_trend_2020, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month')

plt.subplot(2,3,4)
plt.title('Seasonal Trend of Year 2021')
ax = sns.barplot(data= seasonal_trend_2021, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month')

plt.subplot(2,3,5)
plt.title('Seasonal Trend of Year 2022')
ax = sns.barplot(data= seasonal_trend_2022, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month')

plt.subplot(2,3,6)
plt.title('Seasonal Trend of Year 2023')
ax = sns.barplot(data= seasonal_trend_2023, x = 'doa_month',y = 'Patient_count',palet
for bars in ax.containers:
    ax.bar_label(bars)
plt.ylabel('Patiet Counts')
plt.xlabel('Month');
```



In [ ]:

Q21. Average Days Spend by Pateints for differnt Medical Condition?

In [90]:
```
healthcare.head()
```

Out[90]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34

In [91]:
```
avg_days_in_hospital = round(healthcare[['Medical Condition','Days in Hospital']].gro
columns = {'Days in Hospital':'Average Count of Days'})
avg_days_in_hospital.reset_index()
```

Out[91]:

	Medical Condition	Average Count of Days
0	Arthritis	16.0
1	Asthma	15.0
2	Cancer	15.0
3	Diabetes	16.0
4	Hypertension	15.0
5	Obesity	15.0

In [ ]:
```
```

Q22. Can we detect any anomalies or outliers in the billing amount that may indicate billing errors or fraud?

In [92]:
```
healthcare.head()
```

Out[92]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [93]:
```
(( healthcare['Billing Amount'] - healthcare['Billing Amount'].mean() )/healthcare['
healthcare['zscore_Billing_Amount'] = (( healthcare['Billing Amount'] - healthcare['B
healthcare.head()
```

Out[93]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	N
0	Tiffany Ramirez	81	Female	O-	Diabetes	2022-11-17	Patrick Parker	Wallace-Hamilton	Medicare	37490.98	
1	Ruben Burns	35	Male	O+	Asthma	2023-06-01	Diane Jackson	Burke, Griffin and Cooper	UnitedHealthcare	47304.06	
2	Chad Byrd	61	Male	B-	Obesity	2019-01-09	Paul Baker	Walton LLC	Medicare	36874.90	
3	Antonio Frederick	49	Male	B-	Asthma	2020-05-02	Brian Chandler	Garcia Ltd	Medicare	23303.32	
4	Mrs. Brandy Flowers	51	Male	O-	Arthritis	2021-07-09	Dustin Griffin	Jones, Brown and Murray	UnitedHealthcare	18086.34	

In [94]:
```
threshold_limit = 3
healthcare[healthcare['zscore_Billing_Amount'] > threshold_limit]
```

Out[94]:

	Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Billing Amount	Room Number	Adn

There are no outliers present in the dataset.

In [ ]:
```
```

In [95]:
```
%notebook "E:\Project for Resume\Python Healthcare Analysis Project\Healthcare Analys
```

Project Ended on 10th May 2024.