Exploratory Data Analysis

Student ID – 009412995
Student Name – Rishika Gottepally

Performance Assessment

A1. Which patients are at risk of Re-admission.Is "Services" variable dependent on the target variable Re-admission?

A2. From the analysis,Stakeholders will benefit by knowing the dependency of variable with re-admission and how effective mesaures can be taken towards the primary service being provided by hospital on initial admission of the patient.

A3. Most relevant variable to the analysis is 'ReAdmis' which is a categorical variable with values -('Yes,No) and the predictor variable "Services" is also a categorical variable with values -['Blood Work', 'Intravenous', 'CT Scan', 'MRI'] and other variables like - 'Marital','Gender','Employment','Complication_risk' that will add value to our findings.The clean data contanis-Data columns (total 50 columns),dtypes: float64(7), int64(16), object(27) and 10000 entries.

In [64]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [46]:
```
# Load Dataset
df = pd.read_csv('medical_cleaned.csv')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 50 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   CaseOrder              10000 non-null  int64
 1   Customer_id            10000 non-null  object
 2   Interaction            10000 non-null  object
 3   UID                    10000 non-null  object
 4   City                   10000 non-null  object
 5   State                  10000 non-null  object
 6   County                 10000 non-null  object
 7   Zip                    10000 non-null  int64
 8   Lat                    10000 non-null  float64
 9   Lng                    10000 non-null  float64
 10  Population             10000 non-null  int64
 11  Area                   10000 non-null  object
 12  TimeZone               10000 non-null  object
 13  Job                    10000 non-null  object
 14  Children               10000 non-null  int64
 15  Age                    10000 non-null  int64
 16  Income                 10000 non-null  float64
 17  Marital                10000 non-null  object
 18  Gender                 10000 non-null  object
 19  ReAdmis                10000 non-null  object
 20  VitD_levels            10000 non-null  float64
 21  Doc_visits             10000 non-null  int64
 22  Full_meals_eaten       10000 non-null  int64
 23  vitD_supp              10000 non-null  int64
 24  Soft_drink             10000 non-null  object
 25  Initial_admin          10000 non-null  object
 26  HighBlood              10000 non-null  object
 27  Stroke                 10000 non-null  object
 28  Complication_risk      10000 non-null  object
 29  Overweight             10000 non-null  object
 30  Arthritis              10000 non-null  object
 31  Diabetes               10000 non-null  object
 32  HyperlipidemIa         10000 non-null  object
 33  BackPain               10000 non-null  object
 34  Anxiety                10000 non-null  object
 35  Allergic_rhinitis      10000 non-null  object
 36  Reflux_esophagitis     10000 non-null  object
 37  Asthma                 10000 non-null  object
 38  Services               10000 non-null  object
 39  Initial_days           10000 non-null  float64
 40  TotalCharge            10000 non-null  float64
 41  Additional_charges     10000 non-null  float64
 42  Item1                  10000 non-null  int64
 43  Item2                  10000 non-null  int64
 44  Item3                  10000 non-null  int64
 45  Item4                  10000 non-null  int64
 46  Item5                  10000 non-null  int64
 47  Item6                  10000 non-null  int64
 48  Item7                  10000 non-null  int64
 49  Item8                  10000 non-null  int64
dtypes: float64(7), int64(16), object(27)
memory usage: 3.8+ MB
```

B1. Below is the code for performing chi-square test analysis on 'ReAdmis' and 'Services'

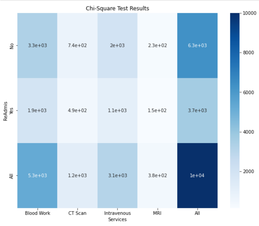
In [55]:
```
#creating a contingency table for the two selected variables
chisqt = pd.crosstab(df.ReAdmis, df.Services, margins=True)
print(chisqt)
```

```
Services  Blood Work  CT Scan  Intravenous  MRI   All
ReAdmis
No              3335      737         2027  232  6331
Yes             1930      488         1103  148  3669
All             5265     1225         3130  380  10000
```

In [56]:
```
#applying chi2_contingency() function on the table to get the statistics, p-value and degree of freedom value.
from scipy.stats import chi2_contingency
chisqt = pd.crosstab(df.ReAdmis, df.Services, margins=True)
value = np.array([chisqt.iloc[0][0:5].values,
                  chisqt.iloc[1][0:5].values])
print(chi2_contingency(value)[0:3])
```

```
(8.892645054628433, 0.06383957953923903, 4)
```

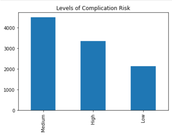
In [57]:
```
# Plotting a heatmap
fig = plt.figure(figsize=(10,8))
sns.heatmap(chisqt, annot=True, cmap='Blues')
plt.title('Chi-Square Test Results')
plt.show()
```

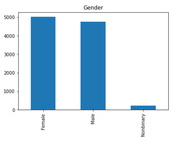


B2. From the above output, 0.06383 is the p-value, 8.892 is the statistical value and 4 is the degree of freedom.

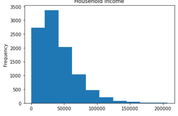
B3. Chi-Square test is performed because it measures how well the observed distribution of data fits with the distribution that is expected and if the variables are independent.This is the exact analysis required to identify if "Service" variable is dependent on 'ReAdmis' and there is a chi2_contingency() function which can help us perform the test directly.

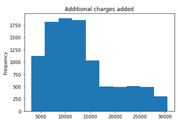
In [59]:
```
#Two catgeorical variables, considering ordinal and nominal
#ordinal - Complication_risk
#nominal - Marital

df['Complication_risk'].value_counts().plot(kind='bar')
plt.title('Levels of Complication Risk')
plt.show()
df['Gender'].value_counts().plot(kind='bar')
plt.title('Gender')
plt.show()
```





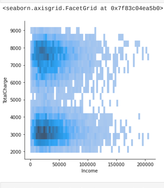
In [51]:
```
#Univariate distribution for two numerical variables
#Income and Additional Charges
df['Income'].plot(kind='hist')
plt.title('Household Income')
plt.show()
df['Additional_charges'].plot(kind='hist')
plt.title('Additional charges added')
plt.show()
```





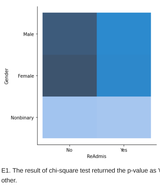
In [52]:
```
#Selecting two continuous variables for bi-variate analysis - 'Income' 'TotalCharge'
sns.displot(df,x='Income',y='TotalCharge')
```

Out[52]:
```
<seaborn.axisgrid.FacetGrid at 0x7f83c04eab60>
```



In [53]:
```
#bi-variate analysis for two categorical variables -
sns.displot(df,x='ReAdmis',y='Gender')
```

Out[53]:
```
<seaborn.axisgrid.FacetGrid at 0x7f83e230deb0>
```



E1. The result of chi-square test returned the p-value as '0.06383' which is greater than 0.05, we accept the Null hypothesis and assume that the variables 'ReAdmis' and 'Services' are not dependent on each other.

E2. The limitation with the analysis performed is that we will not be able to confidently say that just working on one variable'Services' would improve the ReAdmission rate to a any extent as there are many other relevant variables which might provide more insights that will help the analysis.

E3. The test results do not explain the relationship,but only indicate possibility of relationship so further action is required in exploring this relationship and recommending hospitals to work towards providing different primary services to the patients.

F. Link to the video -https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=c40433a2-1f67-406a-9a21-ae3d0174e769

G&H- No external references or resources were used.

In [ ]: