

```
In [1]: import pandas as pd
```

```
In [2]: dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.head()
```

Out[2]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no

```
In [3]: dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5190 entries, 0 to 5189
Data columns (total 13 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Unnamed: 0  5190 non-null  int64
1   visits      5190 non-null  int64
2   gender      5190 non-null  object
3   age         5190 non-null  float64
4   income      5190 non-null  float64
5   illness     5190 non-null  int64
6   reduced     5190 non-null  int64
7   health      5190 non-null  int64
8   private     5190 non-null  object
9   freepoor    5190 non-null  object
10  freerepat    5190 non-null  object
11  nchronic     5190 non-null  object
12  lchronic     5190 non-null  object
dtypes: float64(2), int64(5), object(6)
memory usage: 527.2+ KB
```

```
In [4]: dv["age"]=dv["age"]*50
dv
```

Out[4]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	9.5	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	9.5	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	9.5	0.90	3	0	0	no	no	no	no	no
3	4	1	male	9.5	0.15	1	0	0	no	no	no	no	no
4	5	1	male	9.5	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	11.0	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	13.5	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	18.5	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	26.0	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	36.0	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

```
In [5]: dv["gender"].value_counts()
```

Out[5]:

```
female    2702
male      2488
Name: gender, dtype: int64
```

```
In [10]: dv
```

Out[10]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	9.5	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	9.5	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	9.5	0.90	3	0	0	no	no	no	no	no
3	4	1	male	9.5	0.15	1	0	0	no	no	no	no	no
4	5	1	male	9.5	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	11.0	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	13.5	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	18.5	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	26.0	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	36.0	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [11]:

dv["income"]=dv["income"]*20000
dv

Out[11]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	9.5	11000.0	1	4	1	yes	no	no	no	no
1	2	1	female	9.5	9000.0	1	2	1	yes	no	no	no	no
2	3	1	male	9.5	18000.0	3	0	0	no	no	no	no	no
3	4	1	male	9.5	3000.0	1	0	0	no	no	no	no	no
4	5	1	male	9.5	9000.0	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	11.0	11000.0	0	0	0	no	no	no	no	no
5186	5187	0	male	13.5	26000.0	0	0	1	no	no	no	no	no
5187	5188	0	female	18.5	5000.0	1	0	1	no	no	yes	no	no
5188	5189	0	female	26.0	13000.0	0	0	0	no	no	no	no	no
5189	5190	0	male	36.0	5000.0	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

Find the value count of different data types

In [16]:

dv["private"].value_counts()

Out[16]:

no 2892
yes 2298
Name: private, dtype: int64

In [13]:

dv["reduced"].value_counts()

Out[13]:

0 4454
1 188
2 177
3 108
4 74
5 45
6 40
7 38
8 17
8 17
10 12
9 7
12 6
13 5
11 2
Name: reduced, dtype: int64

In [14]:

dv["health"].value_counts()

```
Out[14]: 0      3026
1       823
2      446
3       273
4       187
5       132
6       104
7        61
8        42
9        32
11       24
10       21
12       19
Name: health, dtype: int64
```

```
In [15]: dv["age"].value_counts()
```

```
Out[15]: 11.0      1213
36.0       822
9.5        752
13.5       523
31.0       316
33.5       315
16.0       301
28.5       273
26.0       222
23.5       181
18.5       146
21.0       126
Name: age, dtype: int64
```

Describing the info of the datatypes

```
In [17]: dv.describe()
```

```
Out[17]:
```

	Unnamed: 0	visits	age	income	illness	reduced	health
count	5190.000000	5190.000000	5190.000000	5190.000000	5190.000000	5190.000000	5190.000000
mean	2595.500000	0.301734	20.319268	11663.198459	1.431985	0.861850	1.217534
std	1498.368279	0.798134	10.239091	7378.133967	1.384152	2.887628	2.124266
min	1.000000	0.000000	9.500000	0.000000	0.000000	0.000000	0.000000
25%	1298.250000	0.000000	11.000000	5000.000000	0.000000	0.000000	0.000000
50%	2595.500000	0.000000	16.000000	11000.000000	1.000000	0.000000	0.000000
75%	3892.750000	0.000000	31.000000	18000.000000	2.000000	0.000000	2.000000
max	5190.000000	9.000000	36.000000	30000.000000	5.000000	14.000000	12.000000

```
In [18]: dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.dropna(axis = 1)
```

```
Out[18]:
```

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	0.22	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	0.27	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	0.37	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	0.52	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	0.72	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

```
In [21]: dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.fillna("16")
```

Out[21]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	0.22	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	0.27	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	0.37	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	0.52	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	0.72	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [20]:

```
dv.ffill(axis = 1)
```

Out[20]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.9	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	0.22	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	0.27	1.3	0	0	1	no	no	no	no	no
5187	5188	0	female	0.37	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	0.52	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	0.72	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [22]:

```
dv.bfill(axis = 1)
```

Out[22]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.9	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	0.22	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	0.27	1.3	0	0	1	no	no	no	no	no
5187	5188	0	female	0.37	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	0.52	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	0.72	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [23]:

```
dv.drop_duplicates()
```

Out[23]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
...
5185	5186	0	female	0.22	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	0.27	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	0.37	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	0.52	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	0.72	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [24]:

dv.drop_duplicates(subset=['private'])

Out[24]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no

In [25]:

dv.drop_duplicates(subset=['freerepat','illness'])

Out[25]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no
5	6	1	female	0.19	0.35	5	1	9	no	no	no	yes	no
6	7	1	female	0.19	0.55	4	0	2	no	no	no	no	no
11	12	1	male	0.19	0.25	2	0	2	no	no	yes	no	no
82	83	1	female	0.19	0.25	1	0	9	no	no	yes	no	no
103	104	1	female	0.19	0.45	0	0	0	yes	no	no	no	no
152	153	2	female	0.22	0.55	5	2	3	no	no	yes	no	yes
303	304	1	male	0.27	0.25	3	0	3	no	no	yes	no	yes
505	506	1	male	0.52	0.25	4	2	7	no	no	yes	no	no
621	622	1	female	0.57	0.25	0	0	0	no	no	yes	no	no

In [26]:

dv.shape

Out[26]:

(5190, 13)

In [27]:

dv.columns

Out[27]:

Index(['Unnamed: 0', 'visits', 'gender', 'age', 'income', 'illness', 'reduced', 'health', 'private', 'freepoor', 'freerepat', 'nchronic', 'lchronic'], dtype='object')

In [28]:

dv.isna().sum()

Out[28]:

Unnamed: 0 0
visits 0
gender 0
age 0
income 0
illness 0
reduced 0
health 0
private 0
freepoor 0
freerepat 0
nchronic 0
lchronic 0
dtype: int64

Analyzing the variables

In [29]:

load libraries

```
import pandas as pd
dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.visits.unique()
```

```
Out[29]: array([1, 2, 3, 4, 8, 5, 7, 6, 9, 0], dtype=int64)
```

```
In [30]: dv.gender.unique()
```

```
Out[30]: array(['female', 'male'], dtype=object)
```

```
In [31]: dv.freerepat.unique()
```

```
Out[31]: array(['no', 'yes'], dtype=object)
```

```
In [32]: dv.private.unique()
```

```
Out[32]: array(['yes', 'no'], dtype=object)
```

```
In [33]: dv.nchronic.unique()
```

```
Out[33]: array(['no', 'yes'], dtype=object)
```

```
In [34]: dv.age.unique()
```

```
Out[34]: array([0.19, 0.22, 0.27, 0.32, 0.37, 0.42, 0.47, 0.52, 0.57, 0.62, 0.67,
               0.72])
```

```
In [35]: dv.income.unique()
```

```
Out[35]: array([0.55, 0.45, 0.9 , 0.15, 0.35, 0.65, 0.25, 0. , 0.06, 1.1 , 0.75,
               0.01, 1.3 , 1.5 ])
```

```
In [36]: dv.nunique()
```

```
Out[36]: Unnamed: 0    5190
visits          10
gender           2
age             12
income          14
illness          6
reduced         15
health          13
private          2
freepoor         2
freerepat        2
nchronic         2
lchronic         2
dtype: int64
```

Exploring and Plotting the data

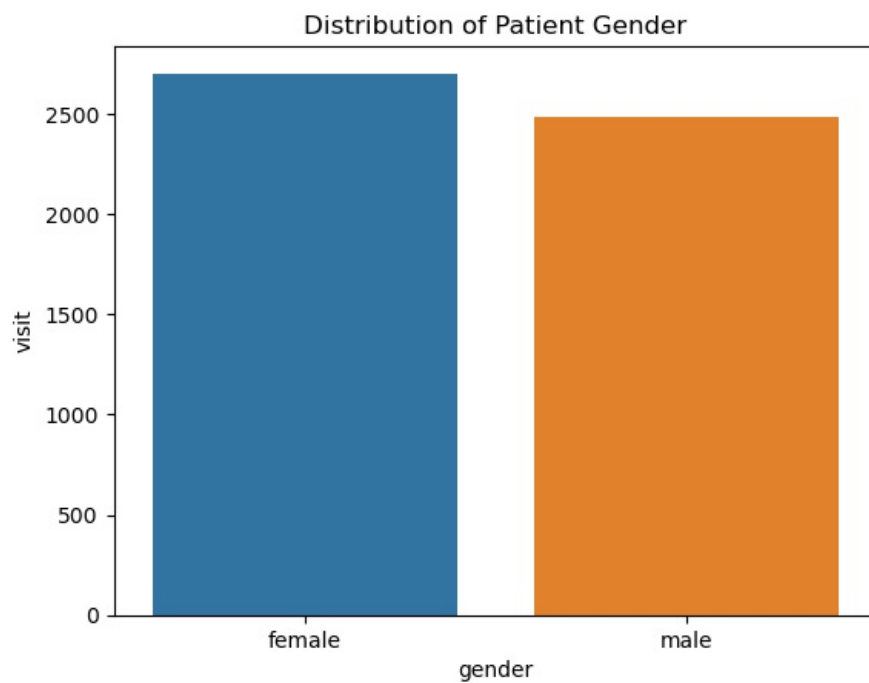
```
In [37]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [38]: dv=pd.read_excel("DoctorVisits (2).xlsx")
dv.head()
```

```
Out[38]:
```

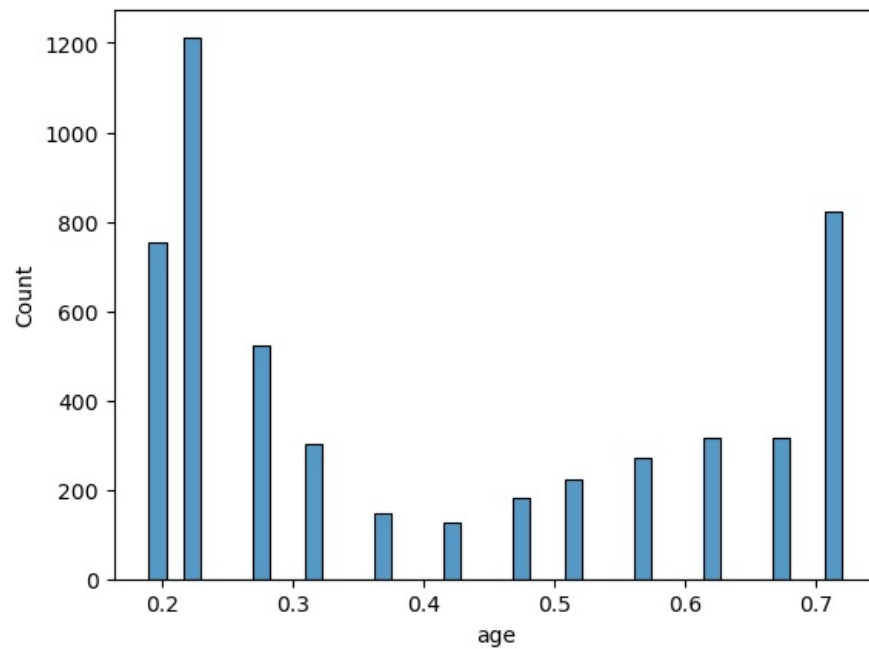
	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	0.19	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	0.19	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	0.19	0.90	3	0	0	no	no	no	no	no
3	4	1	male	0.19	0.15	1	0	0	no	no	no	no	no
4	5	1	male	0.19	0.45	2	5	1	no	no	no	yes	no

```
In [46]: gender_counts = dv['gender'].value_counts()
sns.barplot(x=gender_counts.index,y=gender_counts.values)
plt.xlabel('gender')
plt.ylabel('visit')
plt.title('Distribution of Patient Gender')
plt.show()
```



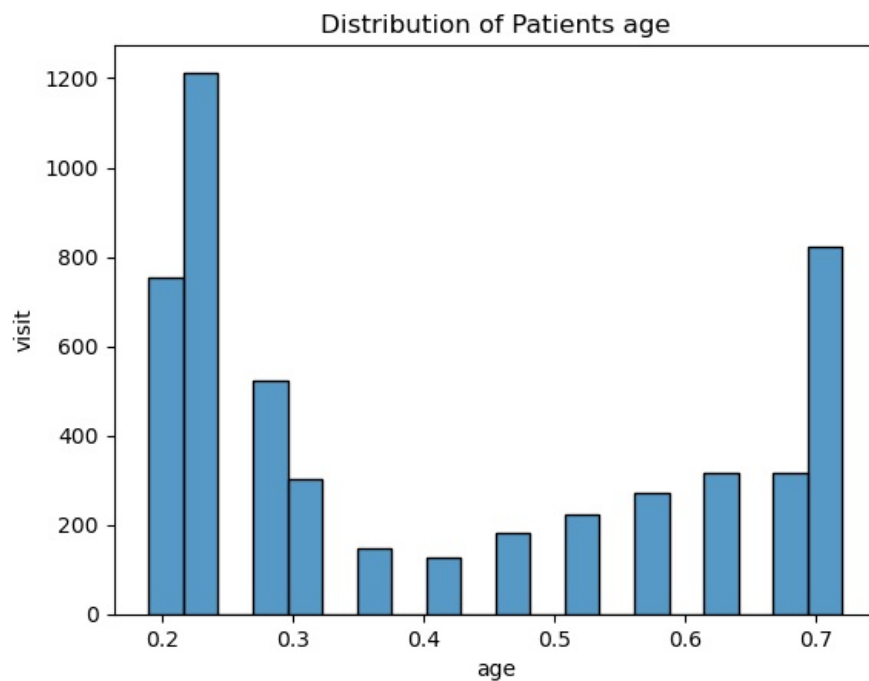
```
In [44]: sns.histplot(dv['age'], bins=40)
```

```
Out[44]: <AxesSubplot:xlabel='age', ylabel='Count'>
```



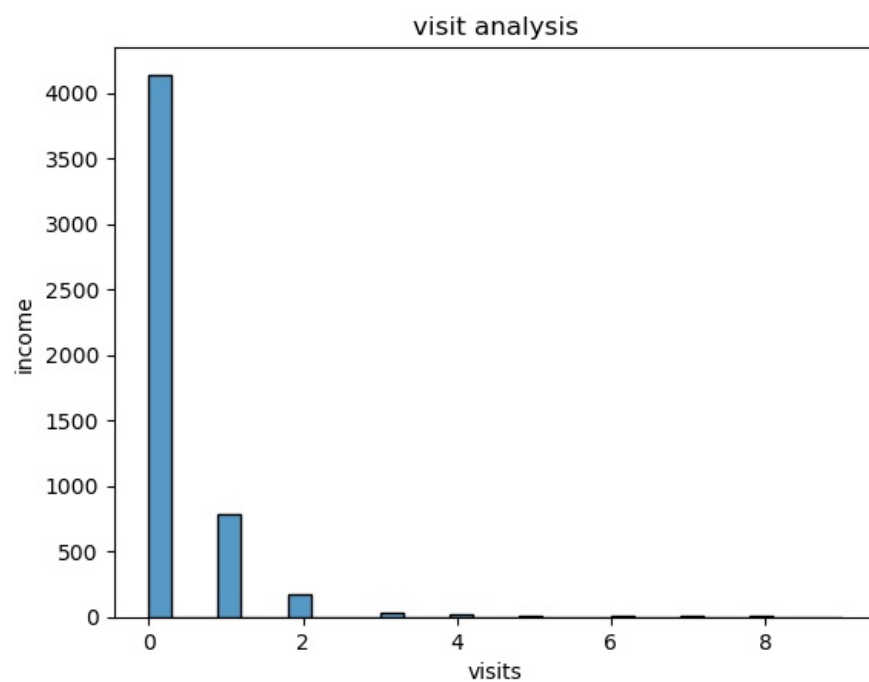
```
In [45]: sns.histplot(dv['age'], bins=20)
plt.xlabel('age')
plt.ylabel('visit')
plt.title('Distribution of Patients age')
plt.show
```

```
Out[45]: <function matplotlib.pyplot.show(close=None, block=None)>
```



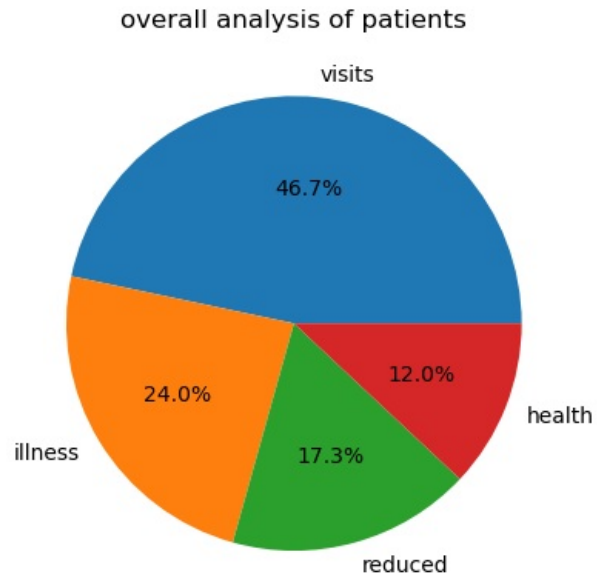
```
In [48]: sns.histplot(dv['visits'], bins=30)
plt.xlabel('visits')
plt.ylabel('income')
plt.title('visit analysis')
plt.show
```

```
Out[48]: <function matplotlib.pyplot.show(close=None, block=None)>
```

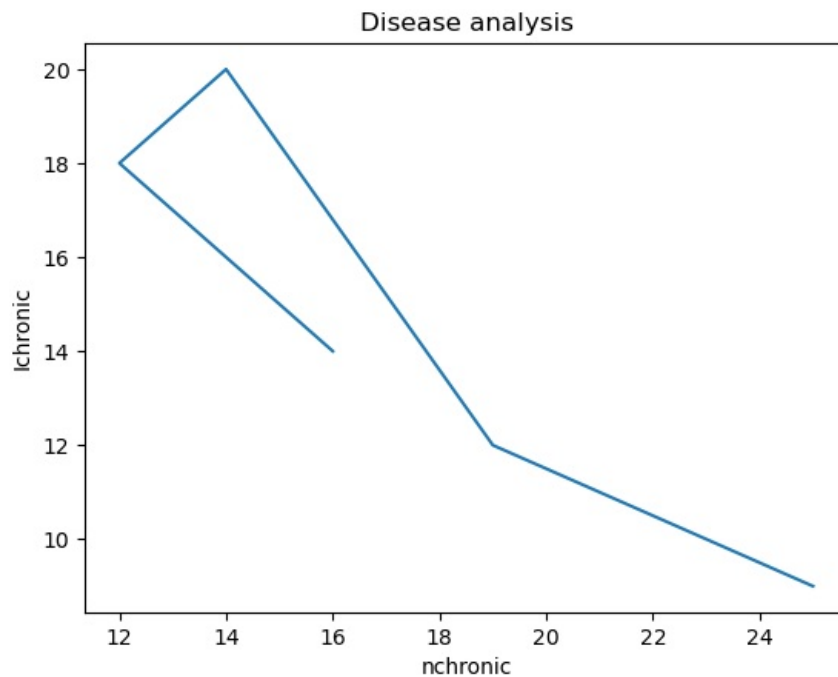


Observations


```
In [53]: labels=['visits','illness','reduced','health']
        sizes=[35,18,13,9]
        plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
        plt.title('overall analysis of patients')
        plt.show()
```

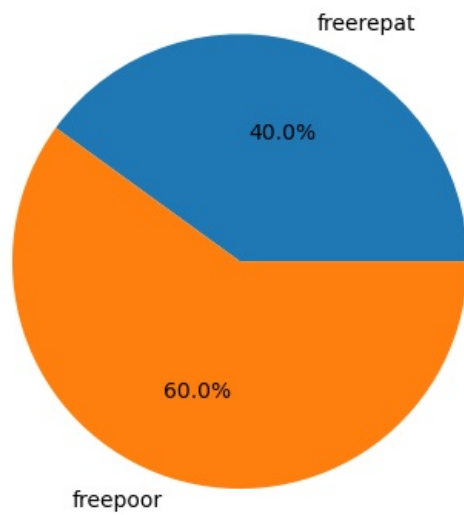


```
In [56]: x = [16,12,14,19,25]
        y = [14,18,20,12,9]
        plt.plot(x,y)
        plt.xlabel('nchronic')
        plt.ylabel('Ichronic')
        plt.title('Disease analysis')
        plt.show()
```



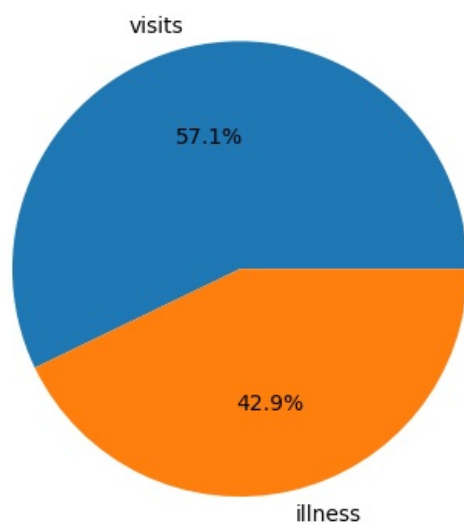
```
In [58]: labels=['freerepat','freepoor']
        sizes=[40,60]
        plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
        plt.title('patient health insurance analysis')
        plt.show()
```

patient health insurance analysis



```
In [60]: labels=['visits','illness']
        sizes=[60,45]
        plt.pie(sizes,labels=labels,autopct = '%1.1f%')
        plt.title('overall analysis of patients')
        plt.show()
```

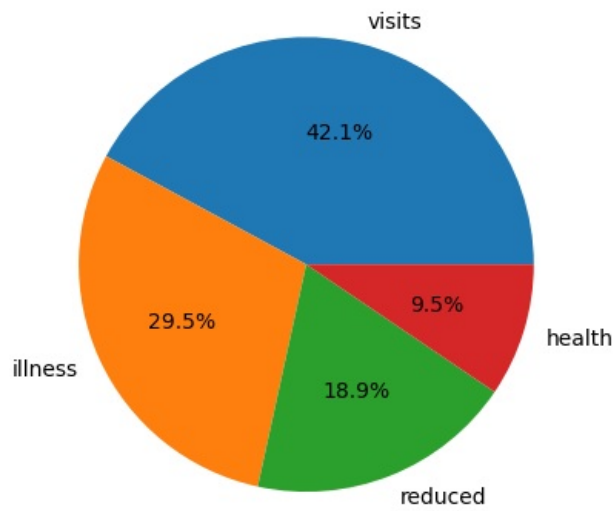
overall analysis of patients



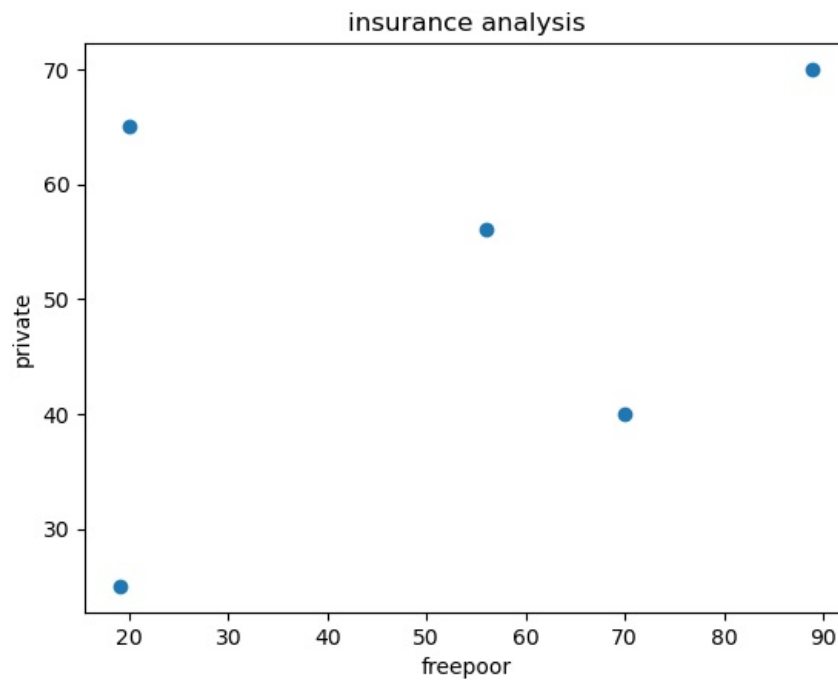
```
In [61]: import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```
In [63]: labels=['visits','illness','reduced','health']
        sizes=[40,28,18,9]
        plt.pie(sizes,labels=labels,autopct = '%1.1f%')
        plt.title('overall analysis of patients')
        plt.show()
```

overall analysis of patients



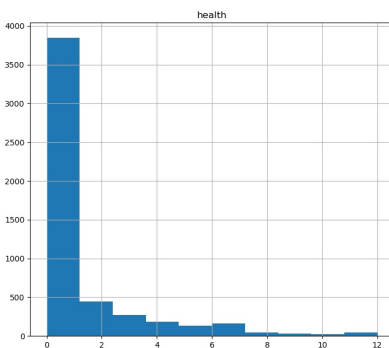
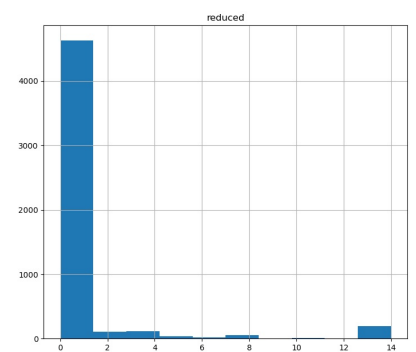
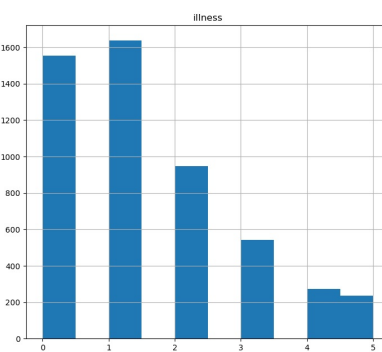
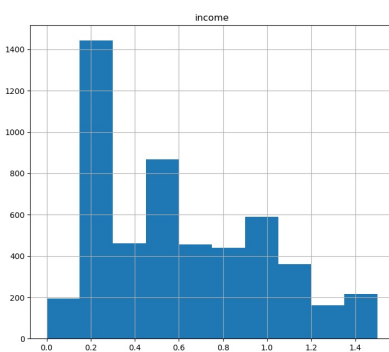
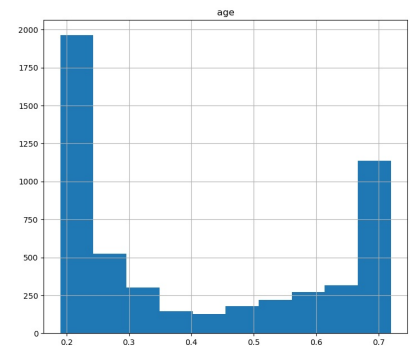
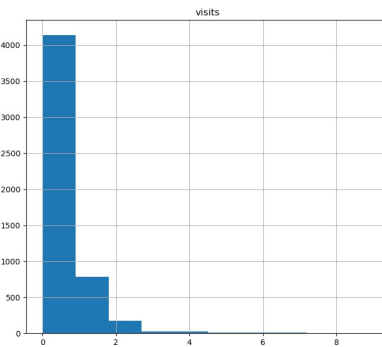
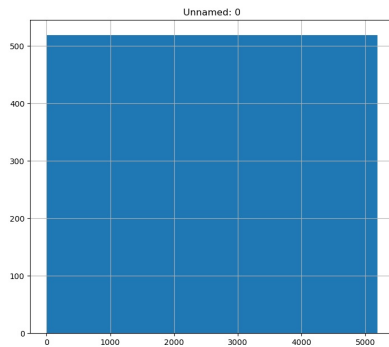
```
In [64]: x = [19,70,56,89,20]
y = [25,40,56,70,65]
plt.scatter(x,y)
plt.xlabel('freepoor')
plt.ylabel('private')
plt.title('insurance analysis')
plt.show()
```



```
In [65]: import pandas as pd
dv=pd.read_excel("DoctorVisits (2).xlsx")
```

```
In [67]: dv.hist(figsize=(30,26))
```

```
Out[67]: array([[<AxesSubplot:title={'center':'Unnamed: 0'}>,
<AxesSubplot:title={'center':'visits'}>,
<AxesSubplot:title={'center':'age'}>],
[<AxesSubplot:title={'center':'income'}>,
<AxesSubplot:title={'center':'illness'}>,
<AxesSubplot:title={'center':'reduced'}>],
[<AxesSubplot:title={'center':'health'}>, <AxesSubplot:>,
<AxesSubplot:>]], dtype=object)
```



```
In [68]: x= (dv[['health']]==1).sum()
y= (dv[['health']]==0).sum()
percent= ((x*y)/(x+y))*100
percent
```

```
Out[68]: health    64702.468174
dtype: float64
```

Conclusion

- We analyzed the dataset which is about the visiting of patients to doctor.
- We examined each variable in the dataset. In comparison to men, women are more numerous. Income has no significant impact on the dataset's consistency. Age and health conditions have a little greater impact on the analytics.
- Female gender is more in number comparable to male gender
- The dataset's private information isn't used very extensively.
- Coming to the factor of age condition and health condition those are some what creating some kind of difference in the analytics.
- Income doesn't create any kind of difference in the dataset it made it's consistency path asusally
- Many diseases are lchronic rather than nchronic.

