Import the libraries

In [3]:
```
# load libraries
import pandas as pd
```

Read the dataset

In [4]:
```
dv=pd.read_excel("DoctorVisits.xlsx")
dv.head()
```

Out[4]:

	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

Display all the columns of the dataset where datatypes,column name,count and overall memory

In [5]:
```
dv=pd.read_excel("DoctorVisits.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
```

Find the total no of people based on their count age,income,gender

In [6]:
```
dv["age"]=dv["age"]*70
```

In [7]:
```
dv
```

Out[7]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	13.3	0.55	1	4	1	yes	no	no	no	no
1	2	1	female	13.3	0.45	1	2	1	yes	no	no	no	no
2	3	1	male	13.3	0.90	3	0	0	no	no	no	no	no
3	4	1	male	13.3	0.15	1	0	0	no	no	no	no	no
4	5	1	male	13.3	0.45	2	5	1	no	no	no	yes	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5185	5186	0	female	15.4	0.55	0	0	0	no	no	no	no	no
5186	5187	0	male	18.9	1.30	0	0	1	no	no	no	no	no
5187	5188	0	female	25.9	0.25	1	0	1	no	no	yes	no	no
5188	5189	0	female	36.4	0.65	0	0	0	no	no	no	no	no
5189	5190	0	male	50.4	0.25	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [8]:
```
dv["income"]=dv["income"]*15000
dv
```

Out[8]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	13.3	8250.0	1	4	1	yes	no	no	no	no
1	2	1	female	13.3	6750.0	1	2	1	yes	no	no	no	no
2	3	1	male	13.3	13500.0	3	0	0	no	no	no	no	no
3	4	1	male	13.3	2250.0	1	0	0	no	no	no	no	no
4	5	1	male	13.3	6750.0	2	5	1	no	no	no	yes	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5185	5186	0	female	15.4	8250.0	0	0	0	no	no	no	no	no
5186	5187	0	male	18.9	19500.0	0	0	1	no	no	no	no	no
5187	5188	0	female	25.9	3750.0	1	0	1	no	no	yes	no	no
5188	5189	0	female	36.4	9750.0	0	0	0	no	no	no	no	no
5189	5190	0	male	50.4	3750.0	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

In [9]:
```
dv["gender"].value_counts()
```

Out[9]:
```
female    2702
male      2488
Name: gender, dtype: int64
```

In [11]:
```
dv
```

Out[11]:

	Unnamed: 0	visits	gender	age	income	illness	reduced	health	private	freepoor	freerepat	nchronic	lchronic
0	1	1	female	13.3	8250.0	1	4	1	yes	no	no	no	no
1	2	1	female	13.3	6750.0	1	2	1	yes	no	no	no	no
2	3	1	male	13.3	13500.0	3	0	0	no	no	no	no	no
3	4	1	male	13.3	2250.0	1	0	0	no	no	no	no	no
4	5	1	male	13.3	6750.0	2	5	1	no	no	no	yes	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5185	5186	0	female	15.4	8250.0	0	0	0	no	no	no	no	no
5186	5187	0	male	18.9	19500.0	0	0	1	no	no	no	no	no
5187	5188	0	female	25.9	3750.0	1	0	1	no	no	yes	no	no
5188	5189	0	female	36.4	9750.0	0	0	0	no	no	no	no	no
5189	5190	0	male	50.4	3750.0	0	0	0	no	no	yes	no	no

5190 rows × 13 columns

Find the value count of different data types

In [12]:
```
dv["visits"].value_counts()
```

Out[12]:
```
0    4141
1     782
2     174
3      30
4      24
7      12
6      12
5       9
8       5
9       1
Name: visits, dtype: int64
```

In [13]:
```
dv["age"].value_counts()
```

Out[13]:
```
15.4    1213
50.4     822
13.3     752
18.9     523
43.4     316
46.9     315
22.4     301
39.9     273
36.4     222
32.9     181
25.9     146
29.4     126
Name: age, 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["illness"].value_counts()
```

Out[15]:
```
1    1638
0    1554
2     946
3     542
4     274
5     236
Name: illness, dtype: int64
```

In [16]:
```
dv["reduced"].value_counts()
```

Out[16]:
```
0     4454
14     188
1      177
2      108
3       74
4       45
5       40
7       38
6       17
8       17
10      12
9        7
12       6
13       5
11       2
Name: reduced, dtype: int64
```

Describing the info of the datatypes

In [18]:
```
# load libraries
import pandas as pd
```

In [19]:
```
dv.describe()
```

Out[19]:

	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	28.446975	8747.398844	1.431985	0.861850	1.217534
std	1498.368279	0.798134	14.334727	5533.600476	1.384152	2.887628	2.124266
min	1.000000	0.000000	13.300000	0.000000	0.000000	0.000000	0.000000
25%	1298.250000	0.000000	15.400000	3750.000000	0.000000	0.000000	0.000000
50%	2595.500000	0.000000	22.400000	8250.000000	1.000000	0.000000	0.000000
75%	3892.750000	0.000000	43.400000	13500.000000	2.000000	0.000000	2.000000
max	5190.000000	9.000000	50.400000	22500.000000	5.000000	14.000000	12.000000

In [20]:
```
dv=pd.read_excel("DoctorVisits.xlsx")
dv.dropna(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	0.90	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 [22]:
```
dv=pd.read_excel("DoctorVisits.xlsx")
dv.fillna("14")
```

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	0.90	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 [23]:
```
dv.ffill(axis = 1)
```

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	0.9	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 [24]:
```
dv.bfill(axis = 1)
```

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
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 [25]:
```
dv.drop_duplicates()
```

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
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 [26]:
```
dv.drop_duplicates(subset=['private'])
```

Out[26]:

	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 [27]:
```
dv.drop_duplicates(subset=['freerepat','illness'])
```

Out[27]:

	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 [28]:
```
dv.shape
```

Out[28]:
```
(5190, 13)
```

In [29]:
```
dv.columns
```

Out[29]:
```
Index(['Unnamed: 0', 'visits', 'gender', 'age', 'income', 'illness', 'reduced',
       'health', 'private', 'freepoor', 'freerepat', 'nchronic', 'lchronic'],
      dtype='object')
```

In [30]:
```
dv.isna().sum()
```

Out[30]:
```
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 [31]:
```
# load libraries
import pandas as pd
dv=pd.read_excel("DoctorVisits.xlsx")
dv.visits.unique()
```

Out[31]:
```
array([1, 2, 3, 4, 8, 5, 7, 6, 9, 0], dtype=int64)
```

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

Out[32]:
```
array(['female', 'male'], dtype=object)
```

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

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

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

Out[34]:
```
array(['yes', 'no'], dtype=object)
```

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

Out[35]:
```
array(['no', 'yes'], dtype=object)
```

In [36]:
```
dv.age.unique()
```

Out[36]:
```
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 [37]:
```
dv.income.unique()
```

Out[37]:
```
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 [38]:
```
dv.nunique()
```

Out[38]:
```
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 [39]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

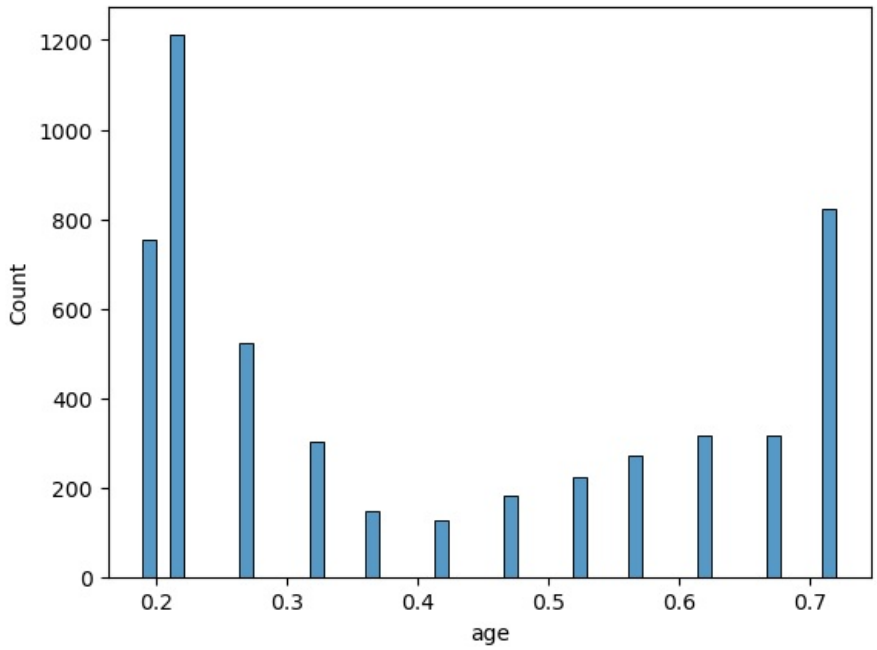
In [41]:
```
dv=pd.read_excel("DoctorVisits.xlsx")
dv.head()
```

Out[41]:

	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

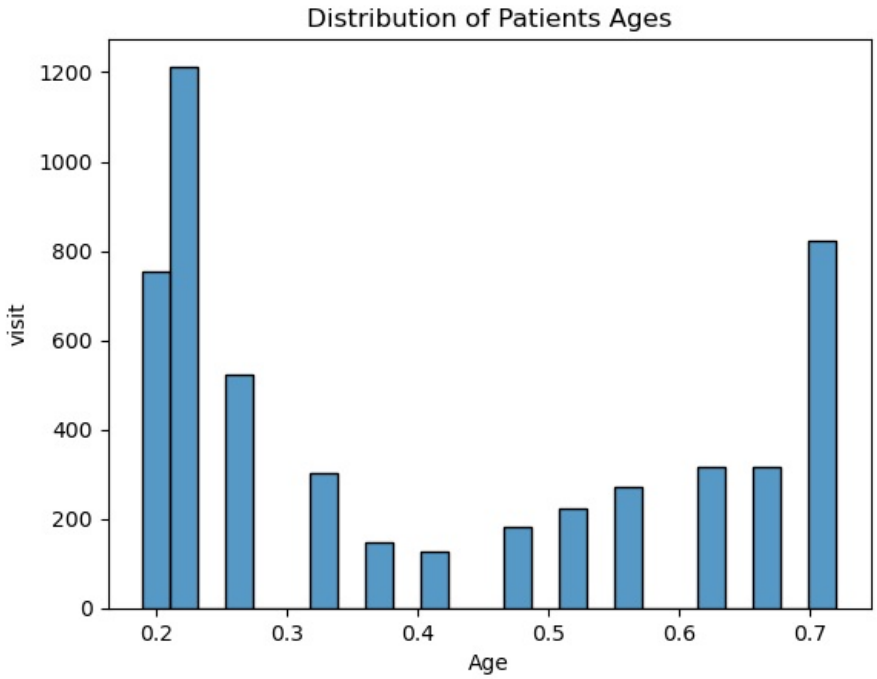
In [42]:
```
sns.histplot(dv['age'], bins=50)
```

Out[42]:
```
<AxesSubplot:xlabel='age', ylabel='Count'>
```

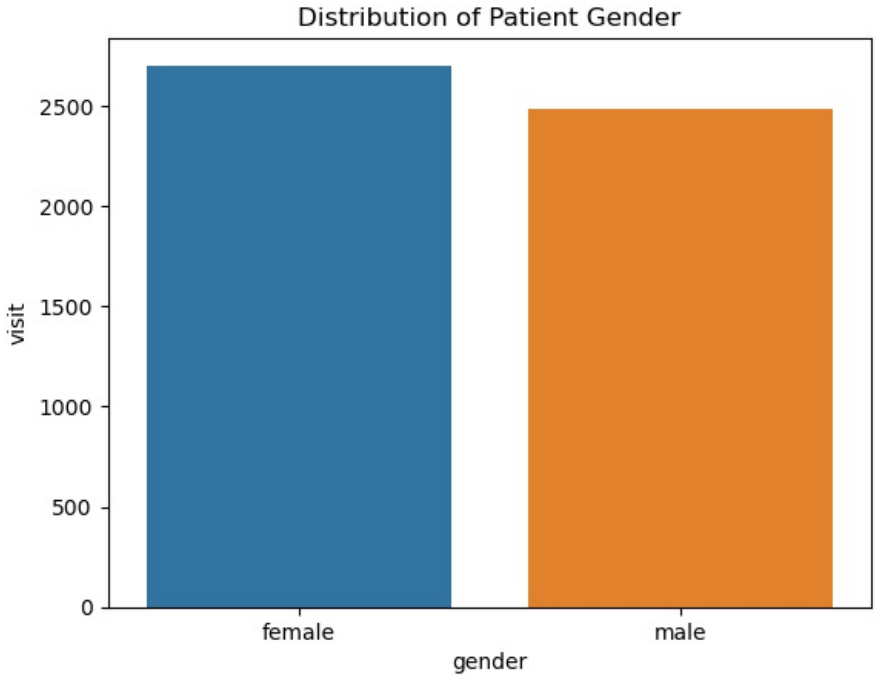


In [43]:
```
sns.histplot(dv['age'], bins=25)
plt.xlabel('Age')
plt.ylabel('visit')
plt.title('Distribution of Patients Ages')
plt.show
```

Out[43]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```

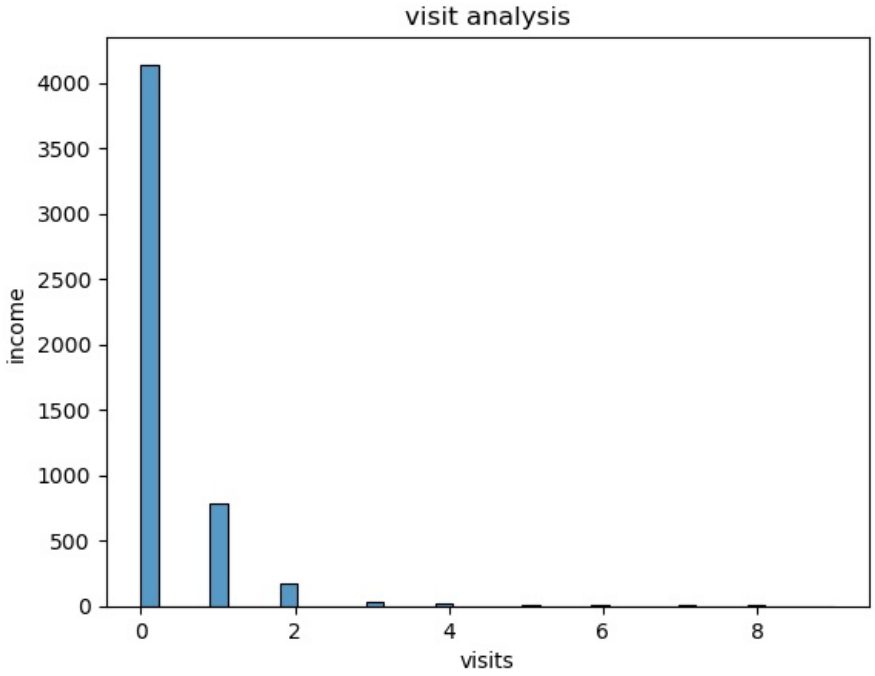


In [57]:
```
gender_counts = dv['gender'].value_counts()
sns.barplot(x=gender_counts.index,y=gender_counts.values)
plt.xlabel('gender')
plt.ylabel('visit')
plt.title('Distribution of Patient Gender')
plt.show()
```



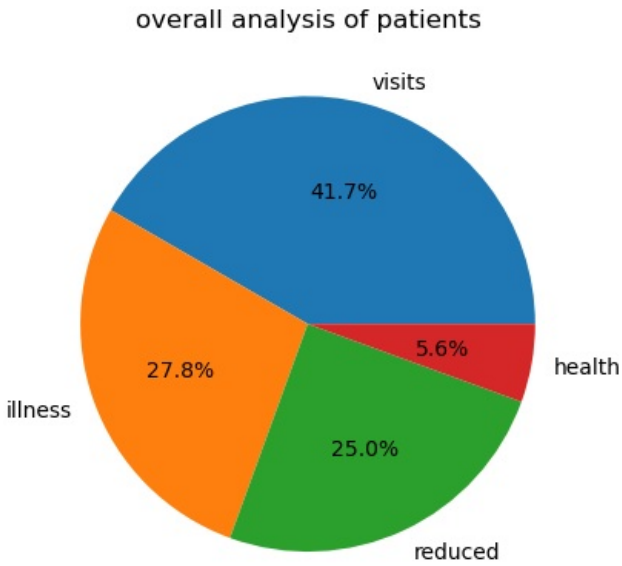
In [45]:
```
sns.histplot(dv['visits'], bins=40)
plt.xlabel('visits')
plt.ylabel('income')
plt.title('visit analysis')
plt.show
```

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

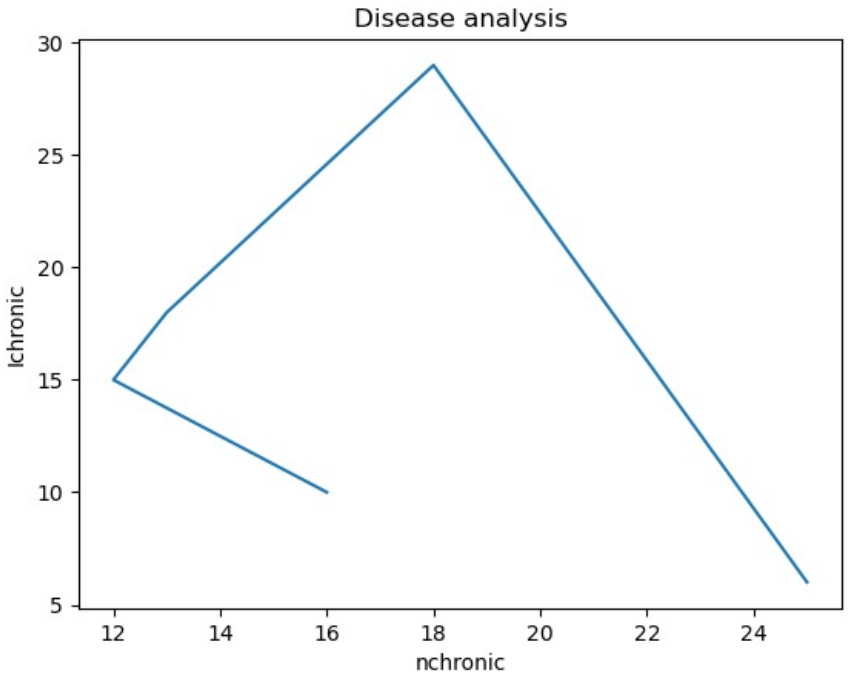


Observations

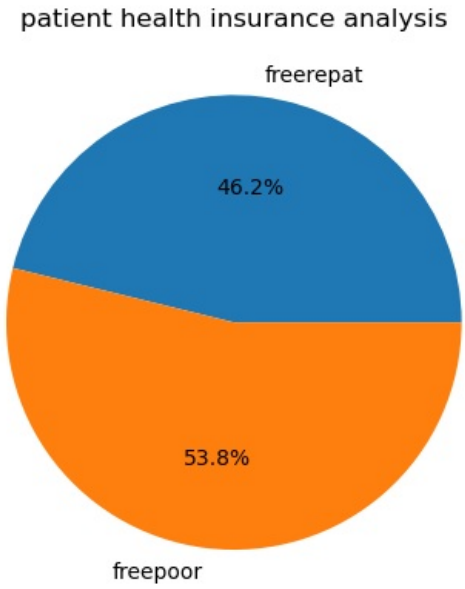
In [46]:
```
labels=['visits','illness','reduced','health']
sizes=[30,20,18,4]
plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
plt.title('overall analysis of patients')
plt.show()
```



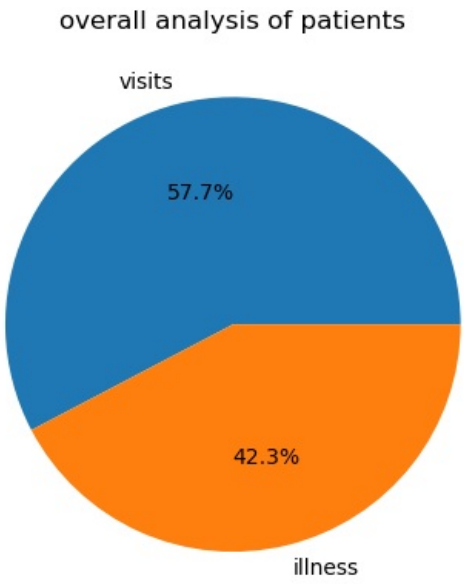
In [47]:
```
x = [16,12,13,18,25]
y = [10,15,18,29,6]
plt.plot(x,y)
plt.xlabel('nchronic')
plt.ylabel('Ichronic')
plt.title('Disease analysis')
plt.show()
```



In [48]:
```
labels=['freerepat','freepoor']
sizes=[60,70]
plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
plt.title('patient health insurance analysis')
plt.show()
```

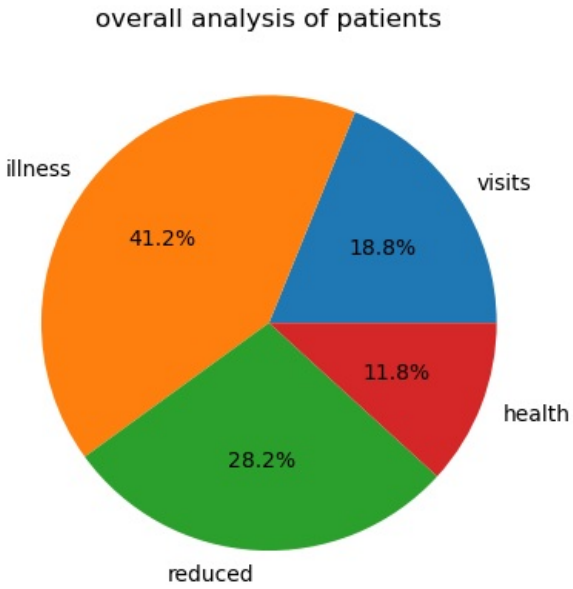


In [49]:
```
labels=['visits','illness']
sizes=[75,55]
plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
plt.title('overall analysis of patients')
plt.show()
```

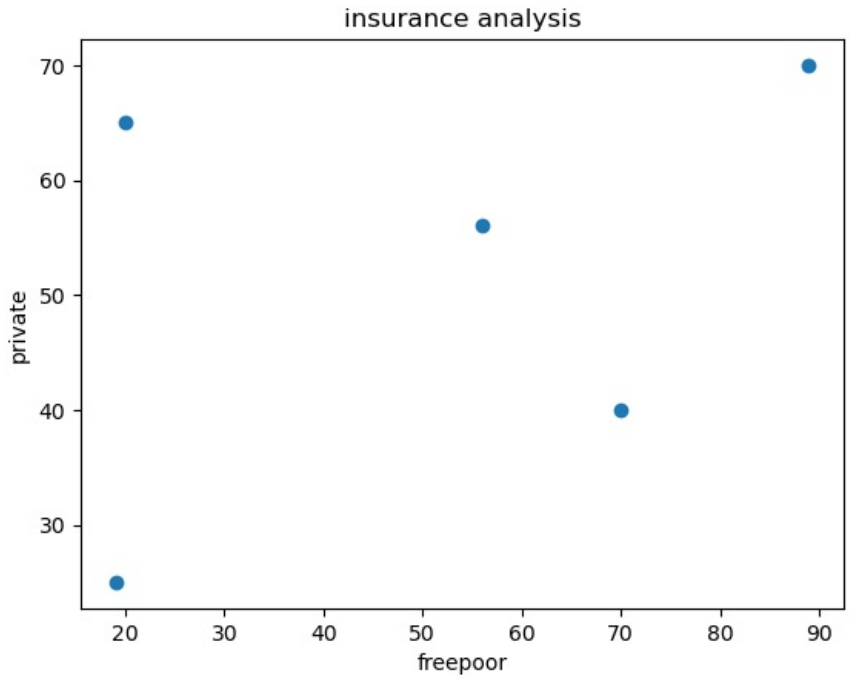


In [50]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [51]:
```
labels=['visits','illness','reduced','health']
sizes=[16,35,24,10]
plt.pie(sizes,labels=labels,autopct = '%1.1f%%')
plt.title('overall analysis of patients')
plt.show()
```



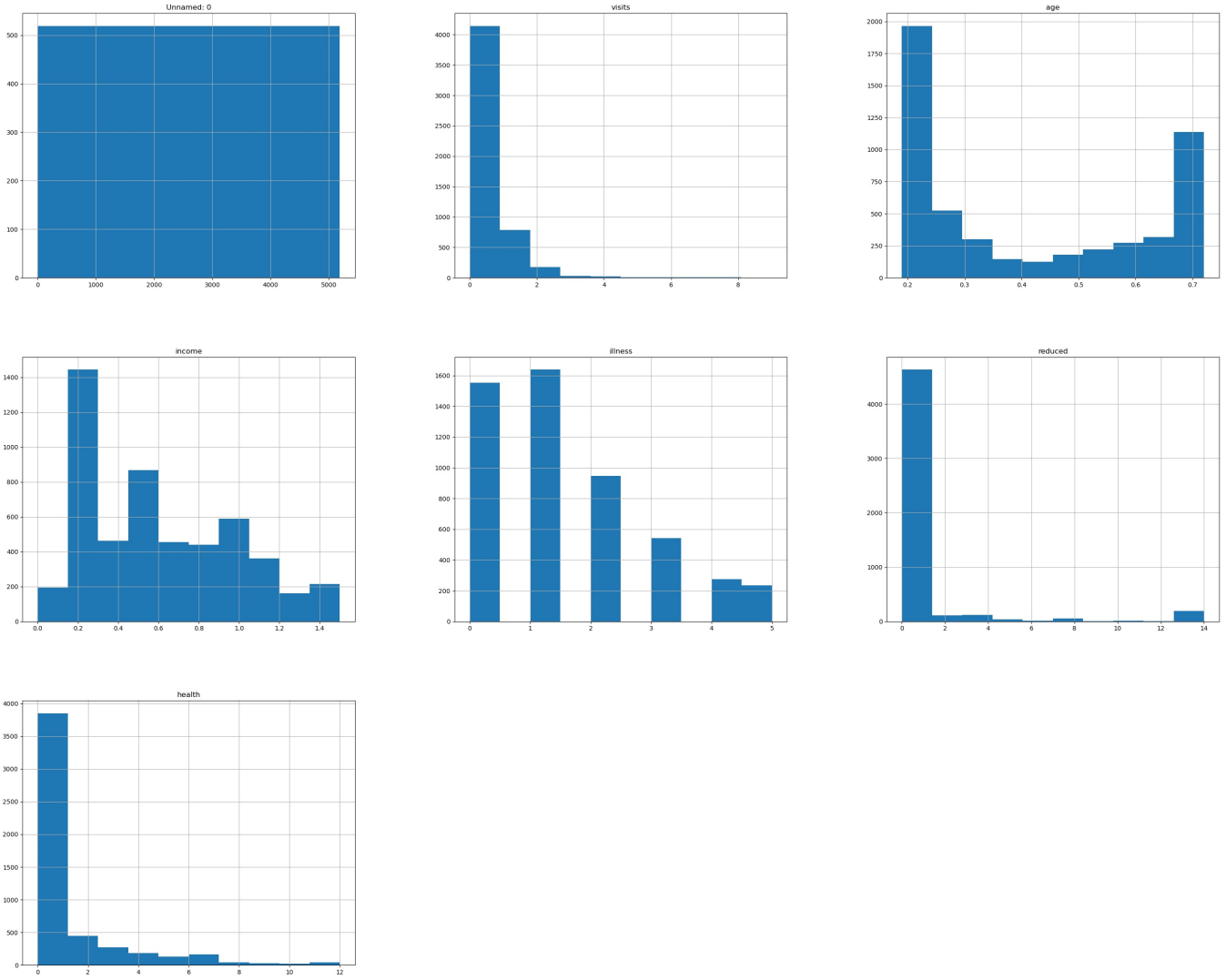
In [52]:
```
x = [19,70,56,89,20]
y = [25,40,56,70,65]
plt.scatter(x,y)
plt.xlabel('freepoor')
plt.ylabel('private')
plt.title('insurance analysis')
plt.show()
```



In [53]:
```
import pandas as pd
dv=pd.read_excel("DoctorVisits.xlsx")
```

In [54]:
```
dv.hist(figsize=(35,28))
```

Out[54]:
```
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 [55]:
```
x= (dv[['health']]==1).sum()
```

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

Out[55]:
```
health    64702.468174
dtype: float64
```

Conclusion

a) We investigated the patient doctor visits dataset.

b) Women outnumber men in terms of population. Income has no effect on the integrity of the dataset. Age and health status have a slightly higher influence on the analyses.

c) The data set's private data is not widely used.

d) When it comes to the factors of age and health, they are causing some kind of difference in the analytics.

e) The dataset's private data is not widely used.