In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from IPython import get_ipython
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
data = pd.read_csv("lung_cancer.csv")
```

In [3]:

```
data.head()
```

Out[3]:

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	AI
0	M	69	1	2	2	1	1	2	
1	M	74	2	1	1	1	2	2	
2	F	59	1	1	1	2	1	2	
3	M	63	2	2	2	1	1	1	
4	F	63	1	2	1	1	1	1	

In [4]:

```
data.tail()
```

Out[4]:

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE
304	F	56	1	1	1	2	2	2
305	M	70	2	1	1	1	1	2
306	M	58	2	1	1	1	1	1
307	M	67	2	1	2	1	1	2
308	M	62	1	1	1	2	1	2

In [5]:

```
data.shape
```

Out[5]:

```
(309, 16)
```

In [6]:

```
data.columns
```

Out[6]:

```
Index(['GENDER', 'AGE', 'SMOKING', 'YELLOW_FINGERS', 'ANXIETY',
       'PEER_PRESSURE', 'CHRONIC DISEASE', 'FATIGUE ', 'ALLERGY ', 'WHEEZI
NG',
       'ALCOHOL CONSUMING', 'COUGHING', 'SHORTNESS OF BREATH',
       'SWALLOWING DIFFICULTY', 'CHEST PAIN', 'LUNG_CANCER'],
      dtype='object')
```

In [7]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 309 entries, 0 to 308
Data columns (total 16 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   GENDER                 309 non-null    object
 1   AGE                    309 non-null    int64
 2   SMOKING                309 non-null    int64
 3   YELLOW_FINGERS         309 non-null    int64
 4   ANXIETY                309 non-null    int64
 5   PEER_PRESSURE          309 non-null    int64
 6   CHRONIC DISEASE        309 non-null    int64
 7   FATIGUE                309 non-null    int64
 8   ALLERGY                309 non-null    int64
 9   WHEEZING               309 non-null    int64
 10  ALCOHOL CONSUMING      309 non-null    int64
 11  COUGHING               309 non-null    int64
 12  SHORTNESS OF BREATH    309 non-null    int64
 13  SWALLOWING DIFFICULTY  309 non-null    int64
 14  CHEST PAIN             309 non-null    int64
 15  LUNG_CANCER            309 non-null    object
dtypes: int64(14), object(2)
memory usage: 38.8+ KB
```

In [8]:

```
data.describe()
```

Out[8]:

	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	
count	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	30
mean	62.673139	1.563107	1.569579	1.498382	1.501618	1.504854	
std	8.210301	0.496806	0.495938	0.500808	0.500808	0.500787	
min	21.000000	1.000000	1.000000	1.000000	1.000000	1.000000	
25%	57.000000	1.000000	1.000000	1.000000	1.000000	1.000000	
50%	62.000000	2.000000	2.000000	1.000000	2.000000	2.000000	
75%	69.000000	2.000000	2.000000	2.000000	2.000000	2.000000	
max	87.000000	2.000000	2.000000	2.000000	2.000000	2.000000	

In [9]:

```
data.isnull().sum()
```

Out[9]:

```
GENDER                  0
AGE                     0
SMOKING                 0
YELLOW_FINGERS          0
ANXIETY                 0
PEER_PRESSURE           0
CHRONIC DISEASE         0
FATIGUE                 0
ALLERGY                 0
WHEEZING                0
ALCOHOL CONSUMING       0
COUGHING                0
SHORTNESS OF BREATH     0
SWALLOWING DIFFICULTY   0
CHEST PAIN              0
LUNG_CANCER             0
dtype: int64
```

In [12]:

```
data.duplicated().sum()
```

Out[12]:

```
33
```

In [14]:

```
data = data.drop_duplicates()
```

In [15]:

```
data.nunique()
```

Out[15]:

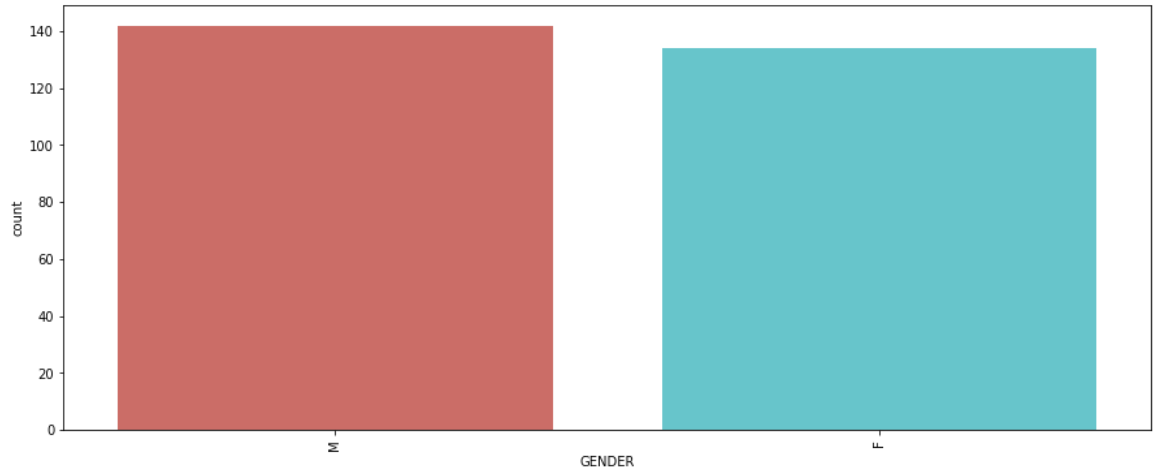
```
GENDER                     2
AGE                       39
SMOKING                    2
YELLOW_FINGERS             2
ANXIETY                    2
PEER_PRESSURE              2
CHRONIC DISEASE            2
FATIGUE                    2
ALLERGY                    2
WHEEZING                   2
ALCOHOL CONSUMING          2
COUGHING                   2
SHORTNESS OF BREATH        2
SWALLOWING DIFFICULTY      2
CHEST PAIN                 2
LUNG_CANCER                2
dtype: int64
```

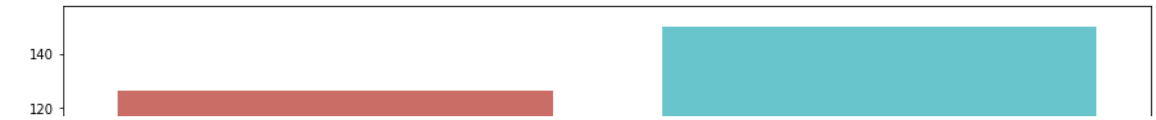
In [16]:

```
data1 = data[['GENDER', 'SMOKING', 'YELLOW_FINGERS', 'ANXIETY',
       'PEER_PRESSURE', 'CHRONIC DISEASE', 'FATIGUE ', 'ALLERGY ', 'WHEEZING',
       'ALCOHOL CONSUMING', 'COUGHING', 'SHORTNESS OF BREATH',
       'SWALLOWING DIFFICULTY', 'CHEST PAIN', 'LUNG_CANCER']]
```

In [17]:

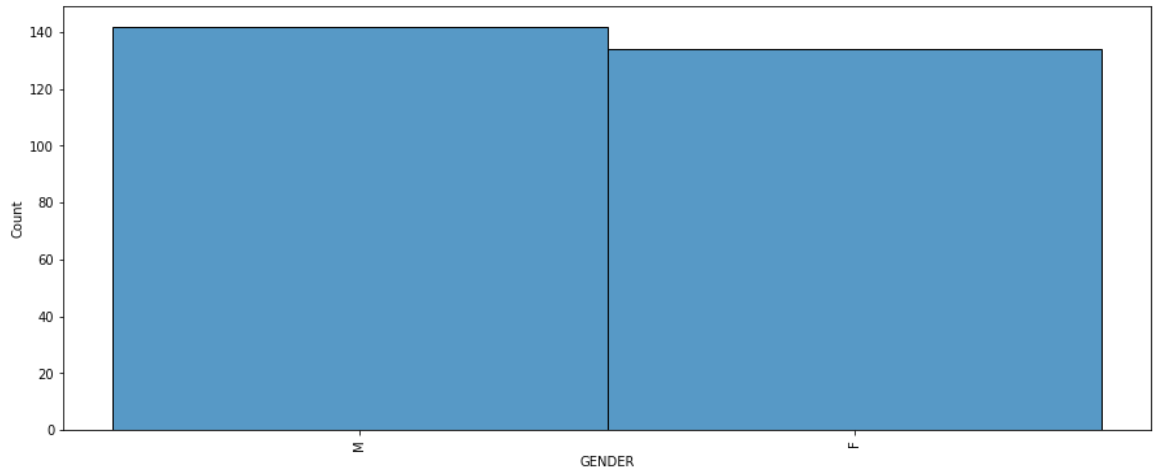
```
for i in data1.columns:
    plt.figure(figsize=(15,6))
    sns.countplot(data1[i], data = data1,
                  palette='hls')
    plt.xticks(rotation = 90)
    plt.show()
```





In [18]:

```
for i in data1.columns:
    plt.figure(figsize=(15,6))
    sns.histplot(data1[i])
    plt.xticks(rotation = 90)
    plt.show()
```



In [19]:

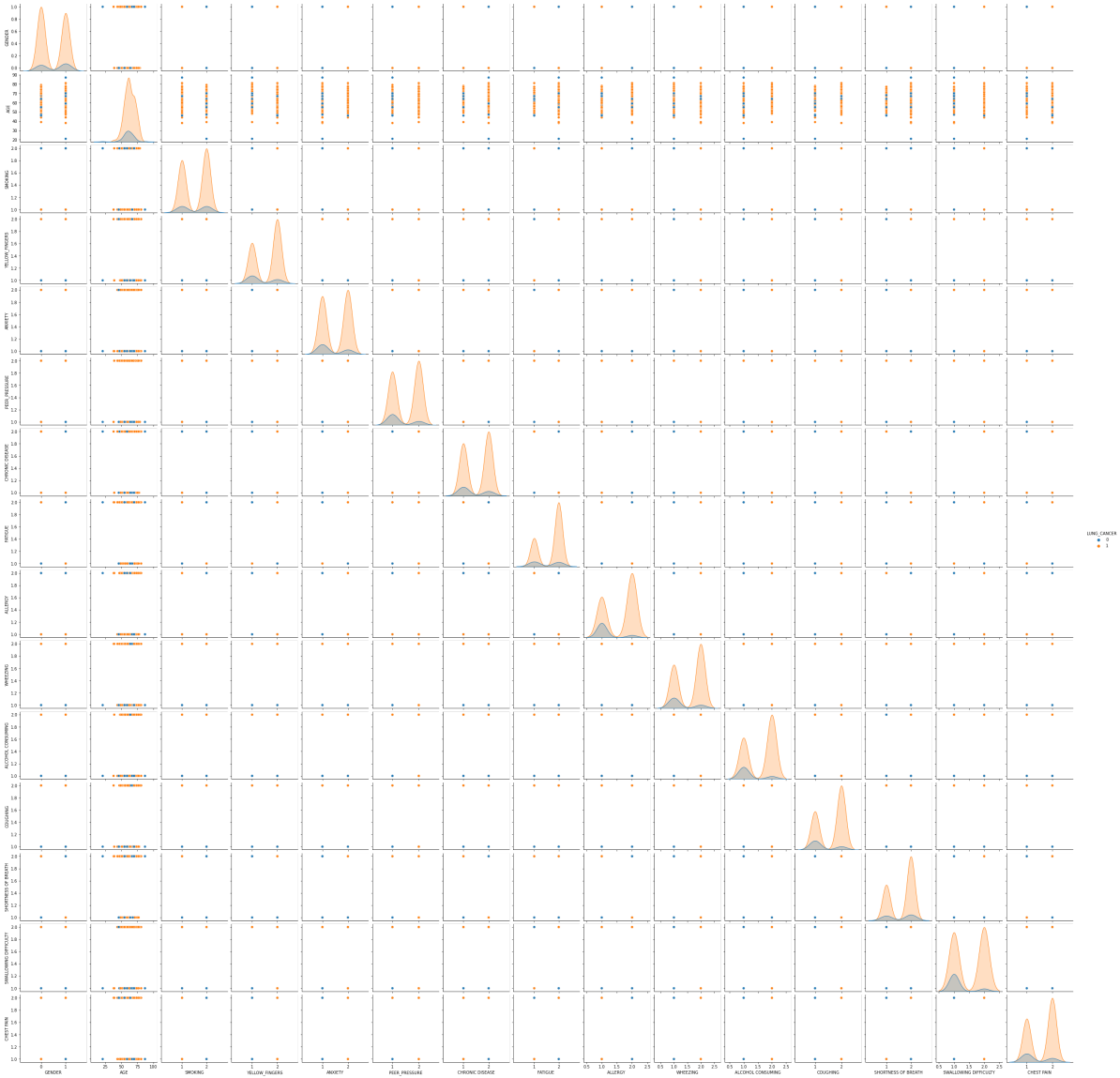
```
data2=data.replace({'NO': 0, 'YES': 1,'M':0,'F':1})
```

In [20]:

```
sns.pairplot(data2,hue='LUNG_CANCER')
```

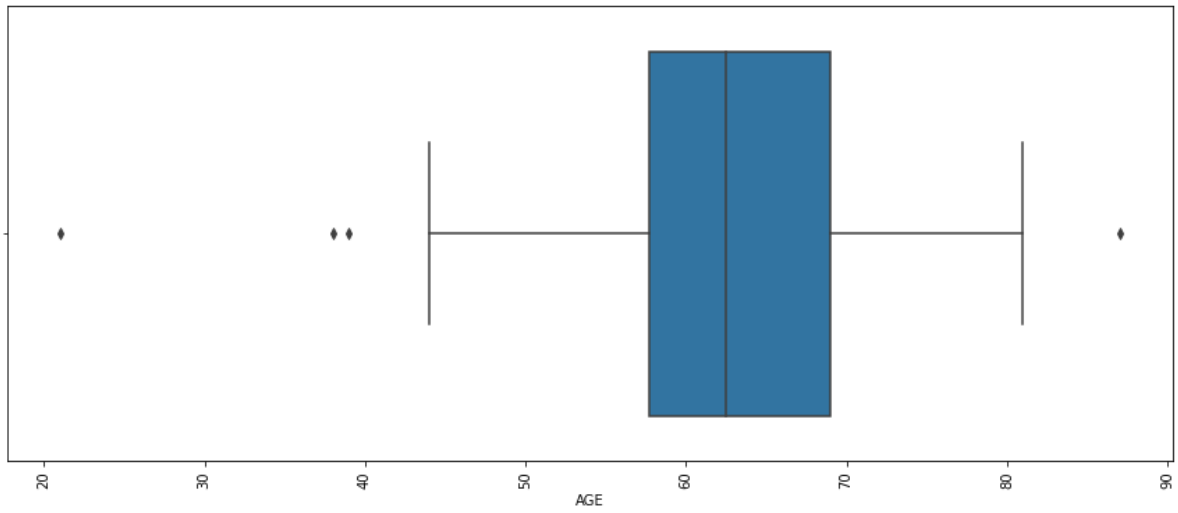
Out[20]:

```
<seaborn.axisgrid.PairGrid at 0x6c4e36f520>
```



In [25]:

```
plt.figure(figsize=(15,6))
sns.boxplot(data2['AGE'])
plt.xticks(rotation = 90)
plt.show()
```



In [27]:

```
data_age = data2['AGE']
Q3 = data_age.quantile(0.75)
Q1 = data_age.quantile(0.25)
IQR = Q3-Q1
lower_limit = Q1 -(1.5*IQR)
upper_limit = Q3 +(1.5*IQR)
age_outliers = data_age[(data_age <lower_limit) | (data_age >upper_limit)]
age_outliers
```

Out[27]:

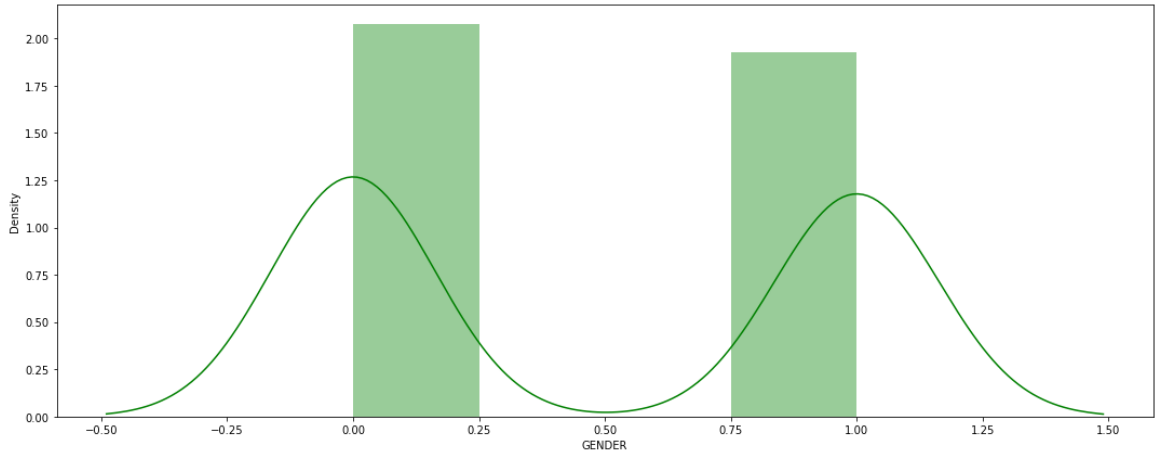
```
22     21
238    38
261    39
277    87
Name: AGE, dtype: int64
```

In [28]:

```
data3 = data2.drop([22, 238, 261, 277])
```

In [29]:

```
for i in data3.columns:
    plt.figure(figsize=(15,6))
    sns.distplot(data3[i], color='green')
    plt.tight_layout()
```



In [32]:

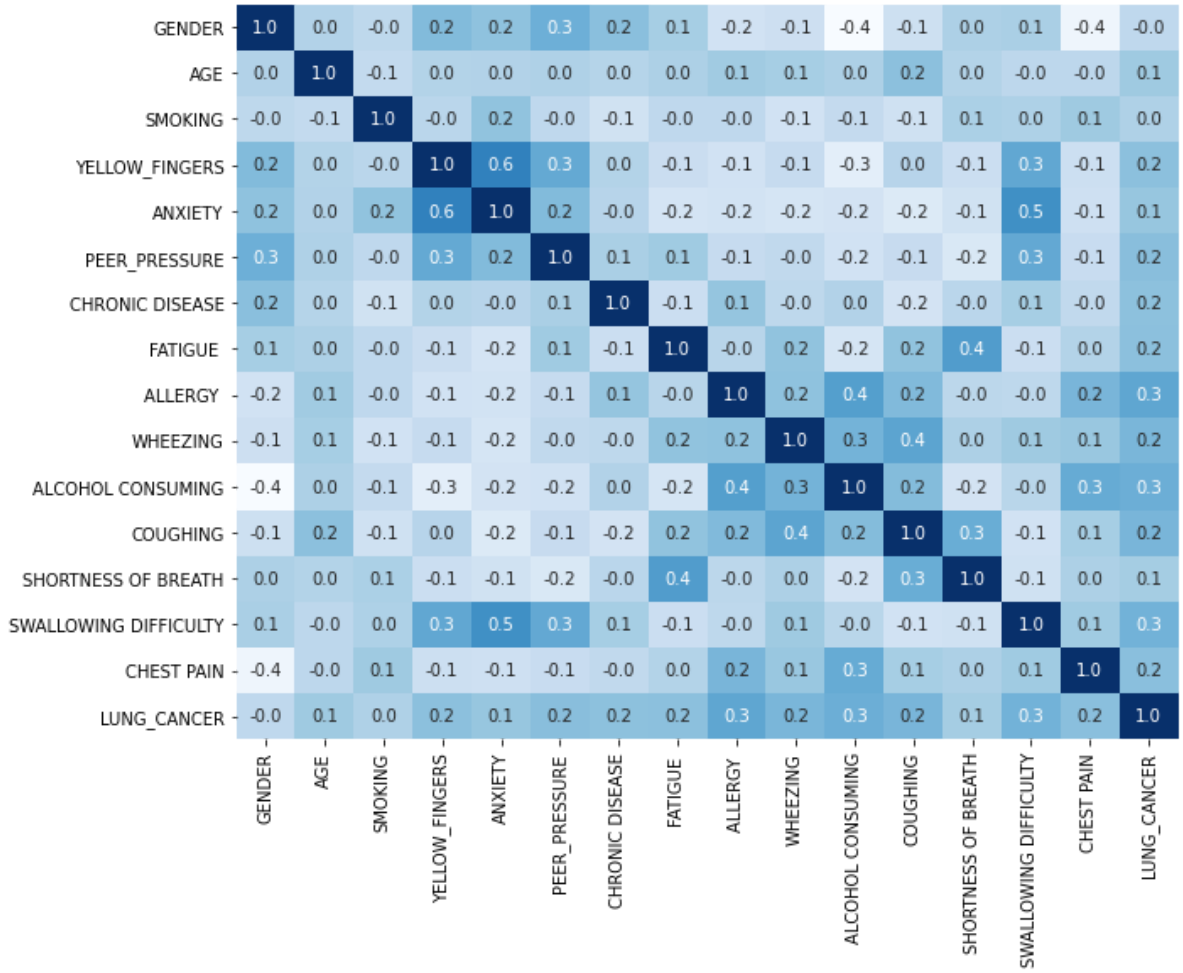
```
data3.corr()
```

Out[32]:

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSUR
GENDER	1.000000	0.024909	-0.033675	0.207346	0.162149	0.28048
AGE	0.024909	1.000000	-0.053188	0.015272	0.023998	0.03240
SMOKING	-0.033675	-0.053188	1.000000	-0.014520	0.154369	-0.03887
YELLOW_FINGERS	0.207346	0.015272	-0.014520	1.000000	0.557641	0.31685
ANXIETY	0.162149	0.023998	0.154369	0.557641	1.000000	0.20577
PEER_PRESSURE	0.280481	0.032400	-0.038872	0.316857	0.205776	1.00000
CHRONIC DISEASE	0.178063	0.004086	-0.143607	0.016338	-0.000271	0.05806
FATIGUE	0.074307	0.047199	-0.037302	-0.093901	-0.172967	0.10118
ALLERGY	-0.151862	0.094199	-0.037489	-0.149165	-0.155672	-0.06840
WHEEZING	-0.114735	0.070826	-0.149404	-0.068139	-0.177993	-0.04645
ALCOHOL CONSUMING	-0.425296	0.045367	-0.061819	-0.280579	-0.163578	-0.15044
COUGHING	-0.120723	0.174810	-0.134205	0.003331	-0.232195	-0.07069
SHORTNESS OF BREATH	0.047495	0.014714	0.053367	-0.103888	-0.146190	-0.21078
SWALLOWING DIFFICULTY	0.057045	-0.027002	0.041523	0.329270	0.471605	0.32517
CHEST PAIN	-0.358701	-0.028664	0.108440	-0.109458	-0.126749	-0.08408
LUNG_CANCER	-0.036082	0.107680	0.034597	0.173962	0.133064	0.18224

In [31]:

```
plt.figure(figsize = (10,8))
sns.heatmap(data3.corr(),annot=True, cbar=False, cmap='Blues', fmt='.1f')
plt.show()
```



In [33]:

```
def minmax_norm(df):
    return (data3 - data3.min()) / ( data3.max() - data3.min())

data4= minmax_norm(data3)
```

In [34]:

```
data4.head()
```

Out[34]:

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGU
0	0.0	0.675676	0.0	1.0	1.0	0.0	0.0	1.
1	0.0	0.810811	1.0	0.0	0.0	0.0	1.0	1.
2	1.0	0.405405	0.0	0.0	0.0	1.0	0.0	1.
3	0.0	0.513514	1.0	1.0	1.0	0.0	0.0	0.
4	1.0	0.513514	0.0	1.0	0.0	0.0	0.0	0.

In [35]:

```
x=data4.drop('LUNG_CANCER',axis=1)
y=data4['LUNG_CANCER']
```

In [36]:

```
x.shape
```

Out[36]:

```
(272, 15)
```

In [37]:

```
y.shape
```

Out[37]:

```
(272,)
```

In [38]:

```
from sklearn.model_selection import train_test_split
```

In [39]:

```
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2,
                                                    random_state=42)
```

In [40]:

```
from sklearn.linear_model import LogisticRegression
```

In [41]:

```
model1= LogisticRegression(random_state=0)
model1.fit(x_train, y_train)
```

Out[41]:

```
LogisticRegression(random_state=0)
```

In [42]:

```
print("Training Accuracy :", model1.score(x_train, y_train))
print("Testing Accuracy :", model1.score(x_test, y_test))
```

```
Training Accuracy : 0.9170506912442397
Testing Accuracy : 0.9454545454545454
```

In [43]:

```
from sklearn.tree import DecisionTreeClassifier
```

In [44]:

```
classifier_dt= DecisionTreeClassifier(criterion='entropy', random_state=0)
classifier_dt.fit(x_train, y_train)
```

Out[44]:

```
DecisionTreeClassifier(criterion='entropy', random_state=0)
```

In [45]:

```
print("Training Accuracy :", classifier_dt.score(x_train, y_train))
print("Testing Accuracy :", classifier_dt.score(x_test, y_test))
```

```
Training Accuracy : 0.9953917050691244
Testing Accuracy : 0.8545454545454545
```

In [53]:

```
data_allergy=data4[data4['ALLERGY ']==1]
data_allergy.groupby(['SMOKING','LUNG_CANCER'])['SMOKING'].count()
```

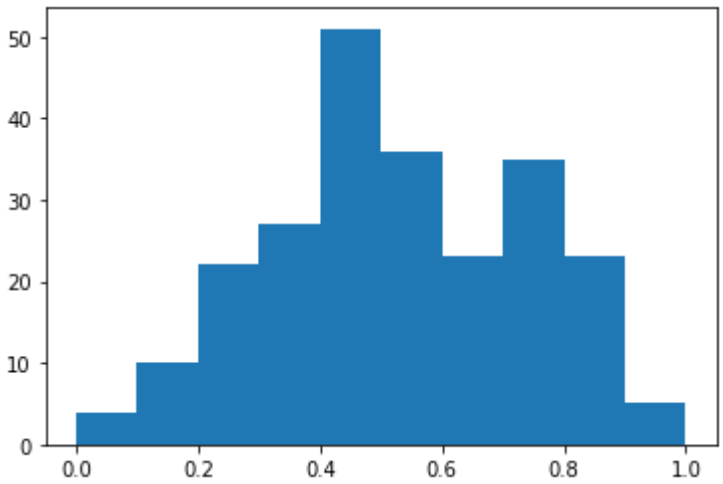
Out[53]:

```
SMOKING  LUNG_CANCER
0.0      1.0            70
1.0      0.0             4
         1.0            74
Name: SMOKING, dtype: int64
```

In [54]:

```
print(data4[data4['LUNG_CANCER']==1]['AGE'].describe())
plt.hist(data4[data4['LUNG_CANCER']==1]['AGE'],bins=10)
plt.show()
```

```
count    236.000000
mean       0.526340
std        0.211087
min        0.000000
25%        0.378378
50%        0.513514
75%        0.702703
max        1.000000
Name: AGE, dtype: float64
```



In [55]:

```
data4[['SMOKING','YELLOW_FINGERS','LUNG_CANCER']].corr()
```

Out[55]:

	SMOKING	YELLOW_FINGERS	LUNG_CANCER
SMOKING	1.000000	-0.014520	0.034597
YELLOW_FINGERS	-0.014520	1.000000	0.173962
LUNG_CANCER	0.034597	0.173962	1.000000