In [1]:
```
# im.show()
from IPython.display import display, Image
display(Image(filename="/Users/LuckyDog/Downloads/cost-of-health-coverage1.jpg"))
```

Medical Cost Personal Datasets - Case Study

In [2]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib as mpl
from matplotlib.dates import DateFormatter
import statsmodels.api as sm
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
%matplotlib inline
import matplotlib.dates as md
pd.set_option("display.precision", 2)
```

In [3]:
```
df = pd.read_csv('/Users/LuckyDog/Downloads/Capstone 2/insurance.csv',header=0)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
age         1338 non-null int64
sex         1338 non-null object
bmi         1338 non-null float64
children    1338 non-null int64
smoker      1338 non-null object
region      1338 non-null object
charges     1338 non-null float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

In [5]:
```
df.describe()
```

Out[5]:

	age	bmi	children	charges
count	1338.00	1338.00	1338.00	1338.00
mean	39.21	30.66	1.09	13270.42
std	14.05	6.10	1.21	12110.01
min	18.00	15.96	0.00	1121.87
25%	27.00	26.30	0.00	4740.29
50%	39.00	30.40	1.00	9382.03
75%	51.00	34.69	2.00	16639.91
max	64.00	53.13	5.00	63770.43

In [6]:
```
df=df.dropna()
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1338 entries, 0 to 1337
Data columns (total 7 columns):
age         1338 non-null int64
sex         1338 non-null object
bmi         1338 non-null float64
children    1338 non-null int64
smoker      1338 non-null object
region      1338 non-null object
charges     1338 non-null float64
dtypes: float64(2), int64(2), object(3)
memory usage: 83.6+ KB
```

Data Pre-Processing

Started the Data processing Encoding non categorical data

In [7]:
```
num_col=df.select_dtypes(include = np.number).columns
print('Numerical Column Number: \n',num_col)

cat_col=df.select_dtypes(exclude = np.number).columns
print('Categorical Column: \n', num_col)
```

```
Numerical Column Number:
 Index(['age', 'bmi', 'children', 'charges'], dtype='object')
Categorical Column:
 Index(['age', 'bmi', 'children', 'charges'], dtype='object')
```

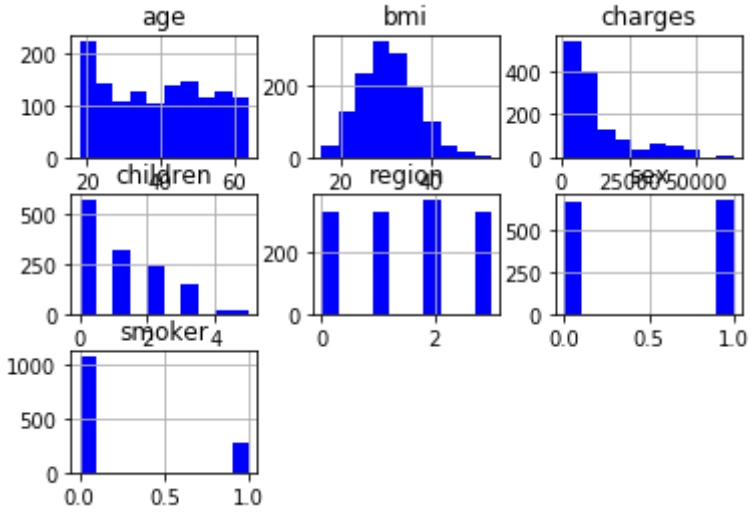
In [8]:
```
#import Label Encoder:
from sklearn import preprocessing

#Label Encoder knows how to understand 'word' variables
label_encoder = preprocessing.LabelEncoder()

#Encode Sex label
df['sex'] = label_encoder.fit_transform(df['sex'])
df['smoker'] = label_encoder.fit_transform(df['smoker'])
df['region'] = label_encoder.fit_transform(df['region'])


df.head()
```

Out[8]:

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.90	0	1	3	16884.92
1	18	1	33.77	1	0	2	1725.55
2	28	1	33.00	3	0	2	4449.46
3	33	1	22.70	0	0	1	21984.47
4	32	1	28.88	0	0	1	3866.86

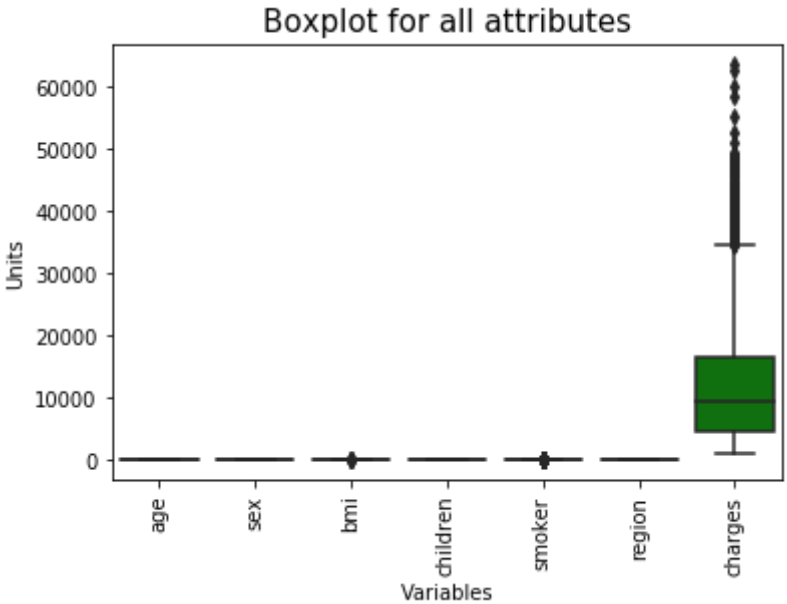
Exploratory Data analysis

In [9]:
```
# Univariate Histograms
data = df

data.hist(color='blue')
plt.figure(figsize=(6,10))
plt.show()
```

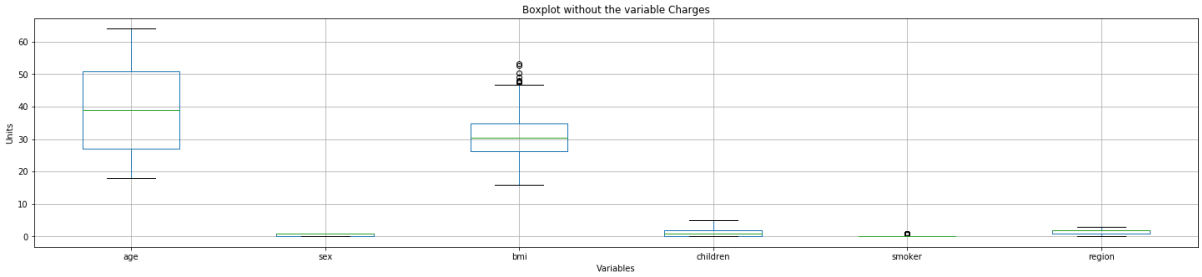



```
<Figure size 432x720 with 0 Axes>
```

In [10]:
```
_=sns.boxplot(data=df,color='green')
plt.title('Boxplot for all attributes', fontsize=15)
plt.xlabel('Variables')
plt.ylabel('Units')
plt.rcParams['figure.figsize'] = (25,5)
plt.xticks(rotation=90)
plt.show()
```



In [11]:
```
_ = df.boxplot(column=['age', 'sex', 'bmi',
       'children', 'smoker',
       'region'] )
plt.title('Boxplot without the variable Charges')
plt.xlabel('Variables')
plt.ylabel('Units')
plt.show()
```

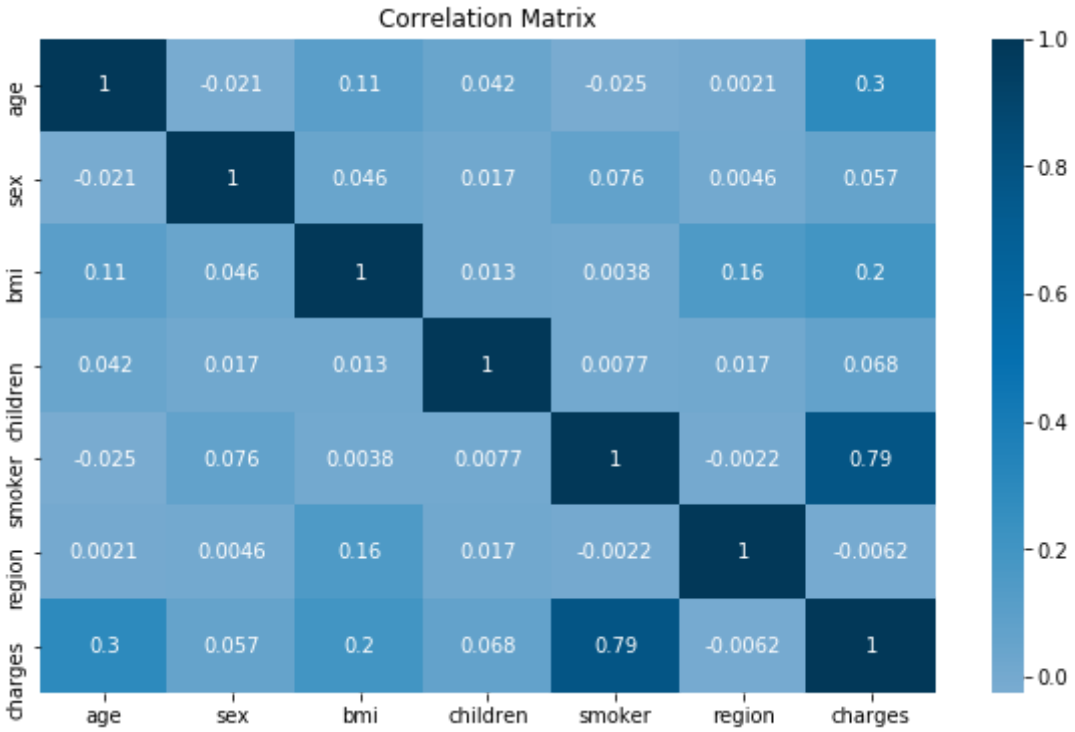


Bmi had a few outliers but it is not seen as an error, so I am not going to delete it

Heatmap

In [12]:
```
fig, ax = plt.subplots(figsize=(10,6))
sns.heatmap(df.corr(), center=0, cmap='PuBu',annot=True)
ax.set_title("Correlation Matrix")
bottom, top = ax.get_ylim()
ax.set_ylim(bottom + 0.5, top - 0.5)
```

Out[12]: (7.0, 0.0)



In [13]:
```
pd.options.display.float_format = '{:.4f}'.format
df.corr()
```

Out[13]:

	age	sex	bmi	children	smoker	region	charges
age	1.0000	-0.0209	0.1093	0.0425	-0.0250	0.0021	0.2990
sex	-0.0209	1.0000	0.0464	0.0172	0.0762	0.0046	0.0573
bmi	0.1093	0.0464	1.0000	0.0128	0.0038	0.1576	0.1983
children	0.0425	0.0172	0.0128	1.0000	0.0077	0.0166	0.0680
smoker	-0.0250	0.0762	0.0038	0.0077	1.0000	-0.0022	0.7873
region	0.0021	0.0046	0.1576	0.0166	-0.0022	1.0000	-0.0062
charges	0.2990	0.0573	0.1983	0.0680	0.7873	-0.0062	1.0000

SMOKING

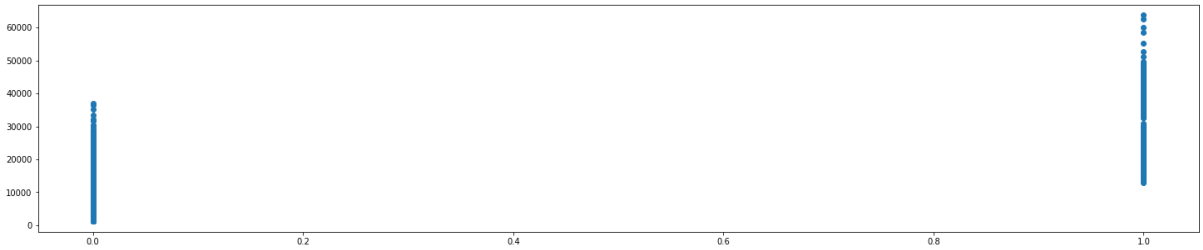
Target Correlation with CHARGES

In [14]:
```
pd.options.display.float_format = '{:.4f}'.format

df[df.columns[1:]].corr()['charges'][:]
```

Out[14]:
```
sex         0.0573
bmi         0.1983
children    0.0680
smoker      0.7873
region     -0.0062
charges     1.0000
Name: charges, dtype: float64
```

Going deeper on the Smokers x Charges Analyses

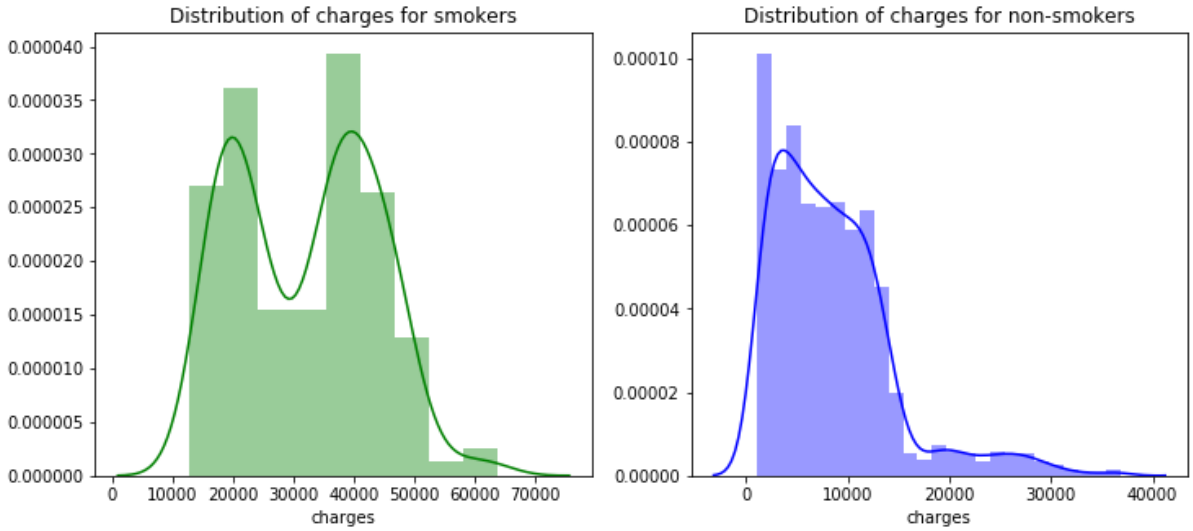
In [15]:
```
plt.scatter(df['smoker'], df['charges'])
plt.show()
```



Smokers pay more for the medical cost

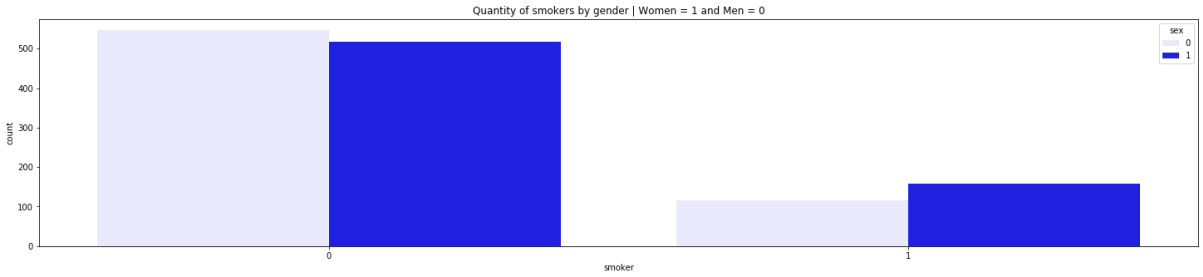
In [16]:
```
f= plt.figure(figsize=(12,5))

ax=f.add_subplot(121)
sns.distplot(df[(df.smoker == 1)]['charges'],color='g',ax=ax)
ax.set_title('Distribution of charges for smokers')

ax=f.add_subplot(122)
sns.distplot(df[(df.smoker == 0)]['charges'],color='b',ax=ax)
ax.set_title('Distribution of charges for non-smokers')
```

Out[16]: Text(0.5, 1.0, 'Distribution of charges for non-smokers')



In [17]:
```
sns.countplot(x="smoker", hue="sex", data=df, color='blue').set_title("Q
uantity of smokers by gender | Women = 1 and Men = 0")
```

Out[17]: Text(0.5, 1.0, 'Quantity of smokers by gender | Women = 1 and Men = 0')



There are more male smokers than women

Chart above show that we have more Male that Smokes, But the question is: Are there more Male that smokes or there are more males pacient?

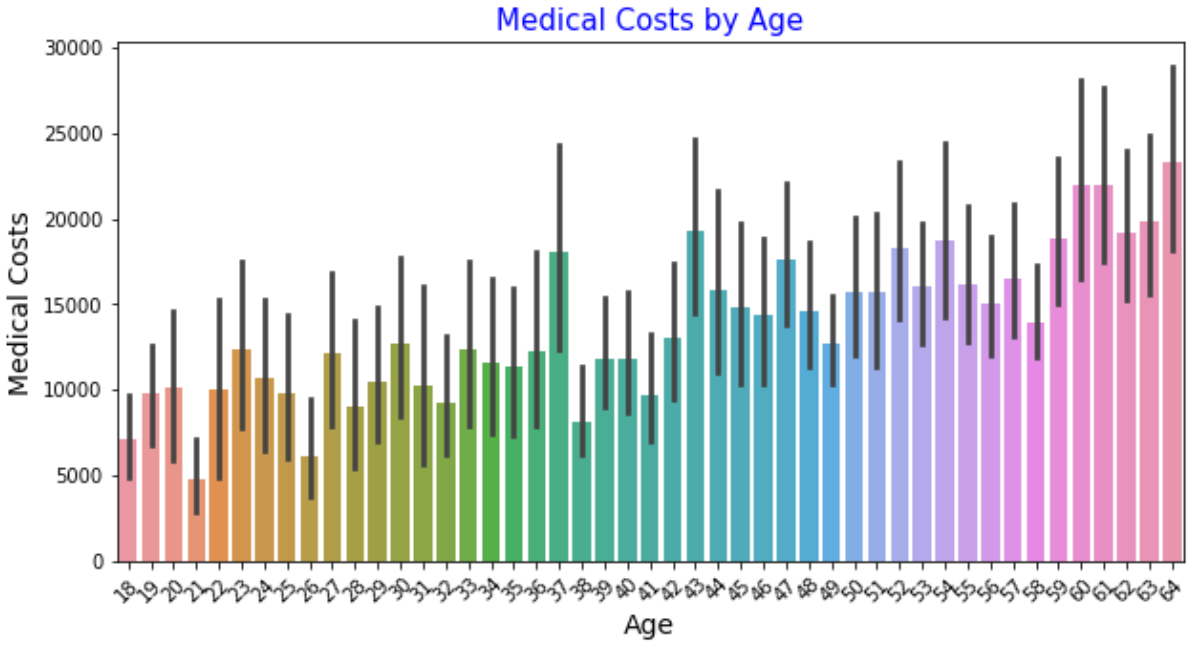
In [18]:
```
df['sex'].value_counts()
```

Out[18]:
```
1    676
0    662
Name: sex, dtype: int64
```

There are a slightly less women than men, but nothing too significative

AGE

In [19]:
```
plt.figure(figsize=(10,5))
sns.barplot(x=df.age, y=df.charges);
plt.xticks(rotation= 45)
plt.xlabel('Age', fontsize=14)
plt.ylabel('Medical Costs', fontsize=14)
plt.title('Medical Costs by Age', color = 'blue', fontsize=15)
plt.show()
```



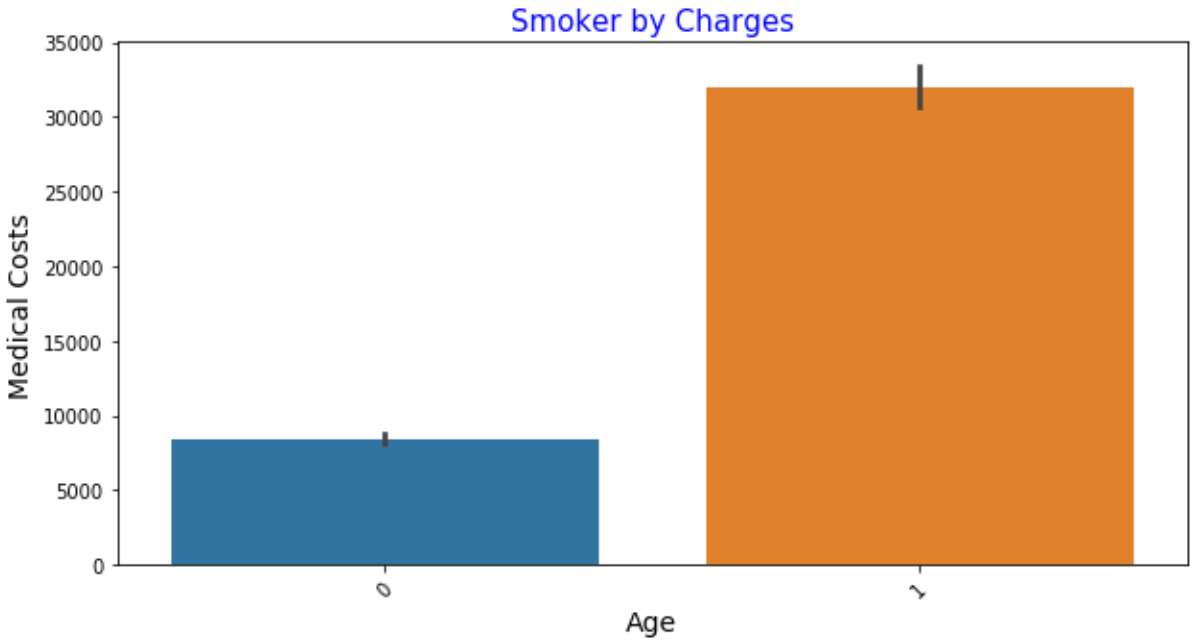
How old you get, higher you pay for Medical Cost

In [20]:
```
data['charges'].mean()
```

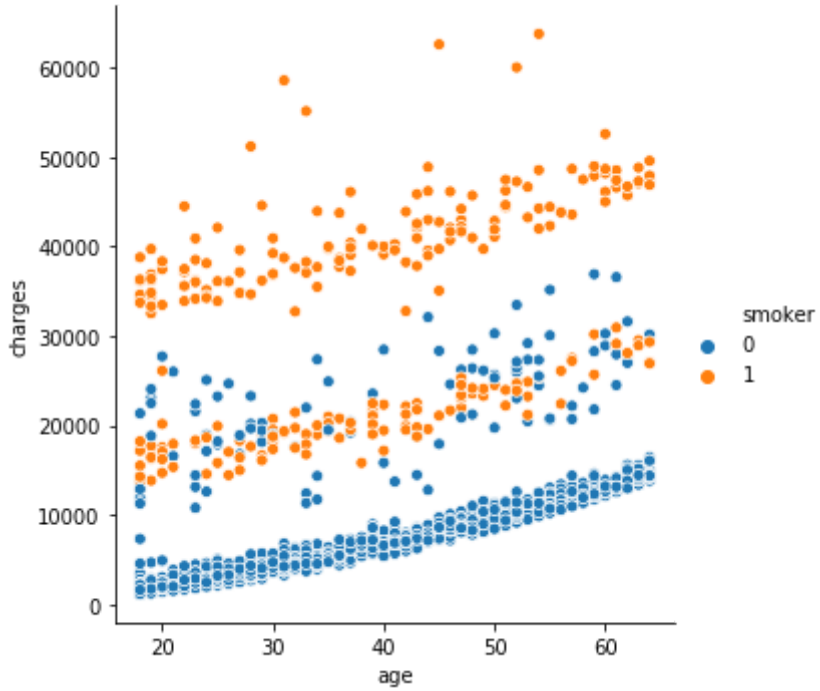
Out[20]:
```
13270.422265141257
```

Every pacient pays in average $13K for medical bills

In [21]:
```
plt.figure(figsize=(10,5))
sns.barplot(x=df.smoker, y=df.charges);
plt.xticks(rotation= 45)
plt.xlabel('Age', fontsize=14)
plt.ylabel('Medical Costs', fontsize=14)
plt.title('Smoker by Charges', color = 'blue', fontsize=15)
plt.show()
```



In [26]:
```
sns.relplot(x="age", y="charges", hue="smoker", data=df);
```

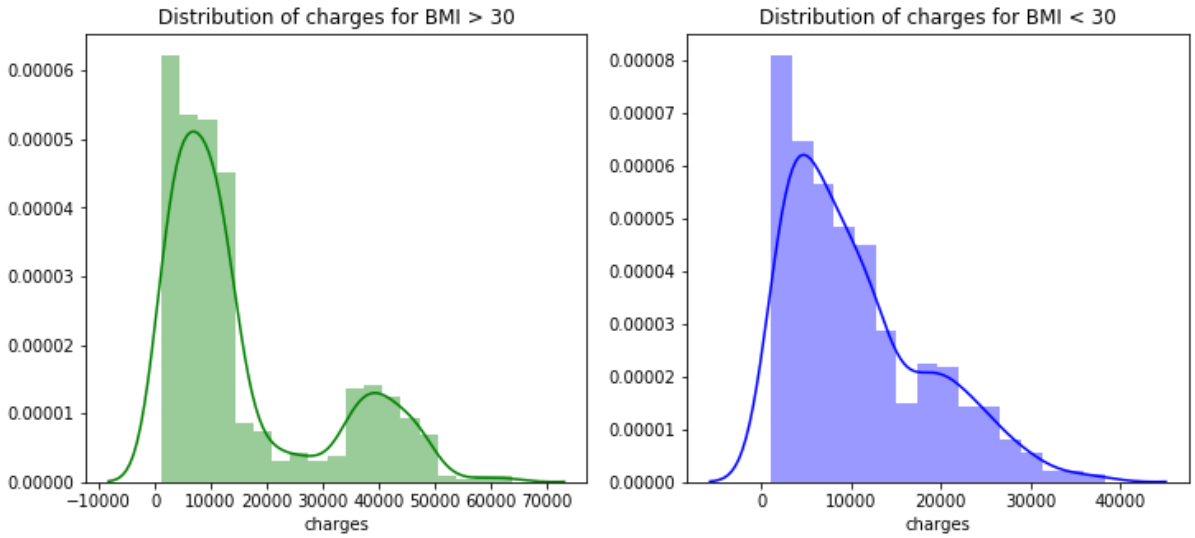


Older and non-smoker == Higher medical cost

BMI

In [37]:
```
f= plt.figure(figsize=(12,5))

ax=f.add_subplot(121)
sns.distplot(df[(df.bmi >= 30)]['charges'],color='g',ax=ax)
ax.set_title('Distribution of charges for BMI > 30')

ax=f.add_subplot(122)
sns.distplot(df[(df.bmi <= 30)]['charges'],color='b',ax=ax)
ax.set_title('Distribution of charges for BMI < 30')
```

Out[37]: Text(0.5, 1.0, 'Distribution of charges for BMI < 30')



BMI > 30 pays more for medical Cost

In [ ]:

In [ ]:

In [ ]: