Project Description

Medical Insurance is a contract that requires an insurer to pay some or all of a person's healthcare costs in exchange for a premium, with the insurance typicallly pays for medical expenses incured by the insured. Insurance cost differs from each individual, certain variable may affect a person insurance cost to be higher or lower.

This project will analyze medical insurance cost in a dataset and exploring variables that may affect it. Further, this project will showcase data analysis skills to provide meaningful insight.

Dataset Overview

The dataset used in this analysis contains information about health insurance policies, coverage, premiums, and demographic details such as age, income, and region.

Columns Description

1. `age` : age of primary beneficiary.
2. `sex` : insurance contractor gender, male or female.
3. `bmi` : body mass index, providing an understanding of body, weights that are relatively high or low relative to height, objective index of body weight (kg/m2) using the ratio of height to weight.
4. `children` : number of children covered by health insurance/number of dependents.
5. `smoker` : is the beneficiary smoking? yes or no.
6. `region` : the beneficiary's residential area in the US; northeast, southeast, southwest, northwest.
7. `charges` : individual medical costs billed by health insurance, in USD.

Environment set-up

In [1]:
```
# import required libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings

# Suppress all warnings
warnings.filterwarnings('ignore')
```

Data Wrangling

We'd load our desired data from the flat csv file `insurance.  csv` to a dataframe using `pandas`, and display its first 5 records. here, we want to check for:

- Missingness in our dataframe.
- Inconsistent data types.
- NaNs.
- Duplicated rows.

In [2]:
```
#Load Data
df = pd.read_csv(r'D:\Projects\My portofolio - Completed projects\Python\2024 10 US Health Insurance EDA\insura

#cheack top rows
df.head()
```

Out[2]:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

In [3]:
```
# display the number of rows and columns in the dataset
df.shape
```

Out[3]:
```
(1338, 7)
```

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

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

-We need change data types of `sex` , `children` , `region` and `smoker` column have inconsistant data type.

In [5]:
```
# exploring age column distribution
df.age.describe()
```

Out[5]:
```
count    1338.000000
mean       39.207025
std        14.049960
min        18.000000
25%        27.000000
50%        39.000000
75%        51.000000
max        64.000000
Name: age, dtype: float64
```

- `age` column data is Logical.

In [6]:
```
df.describe().transpose()
```

Out[6]:

	count	mean	std	min	25%	50%	75%	max
age	1338.0	39.207025	14.049960	18.0000	27.00000	39.000	51.000000	64.00000
bmi	1338.0	30.663397	6.098187	15.9600	26.29625	30.400	34.693750	53.13000
children	1338.0	1.094918	1.205493	0.0000	0.00000	1.000	2.000000	5.00000
charges	1338.0	13270.422265	12110.011237	1121.8739	4740.28715	9382.033	16639.912515	63770.42801

In [7]:
```
# checking for duplicates
df.duplicated().sum()
```

Out[7]:
```
1
```

In [8]:
```
# checking for duplicates
duplicates = df[df.duplicated(keep=False)]

duplicates
```

Out[8]:

	age	sex	bmi	children	smoker	region	charges
195	19	male	30.59	0	no	northwest	1639.5631
581	19	male	30.59	0	no	northwest	1639.5631

- our dataset has one duplicated row.

In [9]:
```
# exploring the unique values of each column
df.nunique()
```

Out[9]:
```
age           47
sex            2
bmi          548
children       6
smoker         2
region         4
charges     1337
dtype: int64
```

Exploration Summery

1. our dataset consists of 1338 rows with 7 columns, and has no NaNs values.
2. `sex`, `region`, `children` and `smoker` coulmns needs to be casted into a categoy type
3. We need to drop one duplicate row.

Data Cleaning

in this section, we'd perform some operations on our dataset based on the previous findings to make our analysis more accurate and clear.

Dropping one duplicate row

In [10]:
```
# drop duplicate
df = df.drop_duplicates()
```

In [11]:
```
# checking data
df.duplicated().sum()
```

Out[11]:
```
0
```

Handling `date` data type

In [12]:
```
df.sex.unique()
```

Out[12]:
```
array(['female', 'male'], dtype=object)
```

In [13]:
```
df.smoker.unique()
```

Out[13]:
```
array(['yes', 'no'], dtype=object)
```

In [14]:
```
# changing data type
df['sex'] = df['sex'].astype('category')
df['smoker'] = df['smoker'].astype('category')
df['region'] = df['region'].astype('category')
df['children'] = df['children'].astype('category')

# cecking changes
print(df[['sex', 'smoker']].dtypes)
```

```
sex       category
smoker    category
dtype: object
```

In [15]:
```
#check data
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1337 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1337 non-null   int64
 1   sex       1337 non-null   category
 2   bmi       1337 non-null   float64
 3   children  1337 non-null   category
 4   smoker    1337 non-null   category
 5   region    1337 non-null   category
 6   charges   1337 non-null   float64
dtypes: category(4), float64(2), int64(1)
memory usage: 47.7 KB
```

We endded up with a datafram of 1337 rows and 7 columns, and everything looks tidy and clean. We'd proceed in visualizing it to extract meaningful insights from it.

Data Visualization and EDA

Now that our data is clean, we'd perform some EDA on it in order to extract useful insights from it.

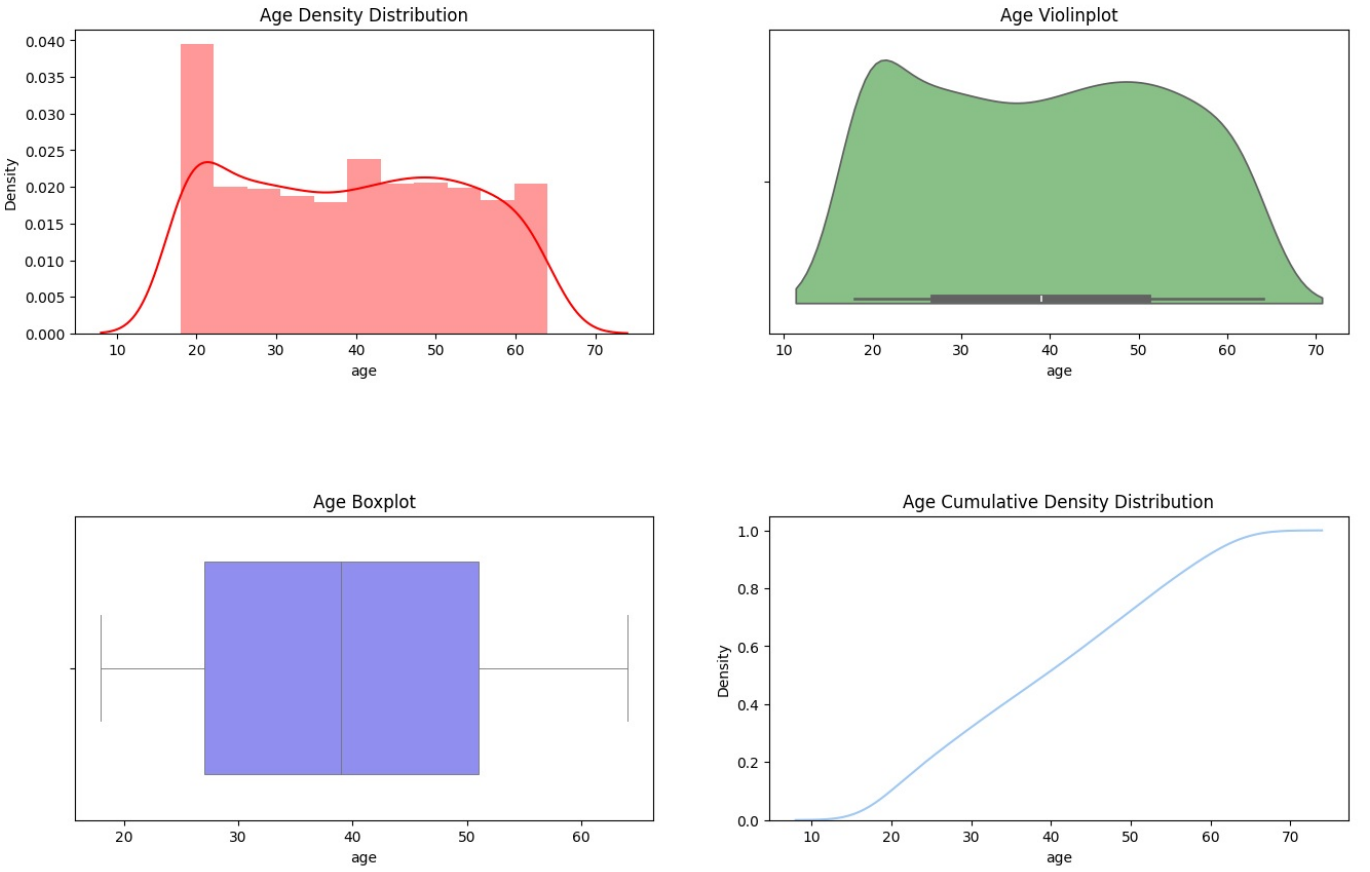
What Can We Infer from the Age Analysis?

In [16]:
```
#Let's construct a function that shows the summary and density distribution of a numerical attribute:
def summary(x):
    x_min = df[x].min()
    x_max = df[x].max()
```

```
    Q1 = df[x].quantile(0.25)
    Q2 = df[x].quantile(0.50)
    Q3 = df[x].quantile(0.75)
    print(f'5 Point Summary of {x.capitalize()} Attribute:\n'
          f'{x.capitalize()}(min) : {x_min}\n'
          f'Q1                    : {Q1}\n'
          f'Q2(Median)            : {Q2}\n'
          f'Q3                    : {Q3}\n'
          f'{x.capitalize()}(max) : {x_max}')

    fig = plt.figure(figsize=(16, 10))
    plt.subplots_adjust(hspace = 0.6)
    sns.set_palette('pastel')

    plt.subplot(221)
    ax1 = sns.distplot(df[x], color = 'r')
    plt.title(f'{x.capitalize()} Density Distribution')

    plt.subplot(222)
    ax2 = sns.violinplot(x = df[x], palette = 'Accent', split = True)
    plt.title(f'{x.capitalize()} Violinplot')

    plt.subplot(223)
    ax2 = sns.boxplot(x=df[x], palette = 'cool', width=0.7, linewidth=0.6)
    plt.title(f'{x.capitalize()} Boxplot')

    plt.subplot(224)
    ax3 = sns.kdeplot(df[x], cumulative=True)
    plt.title(f'{x.capitalize()} Cumulative Density Distribution')

    plt.show()
```

In [17]:
```
summary('age')
```

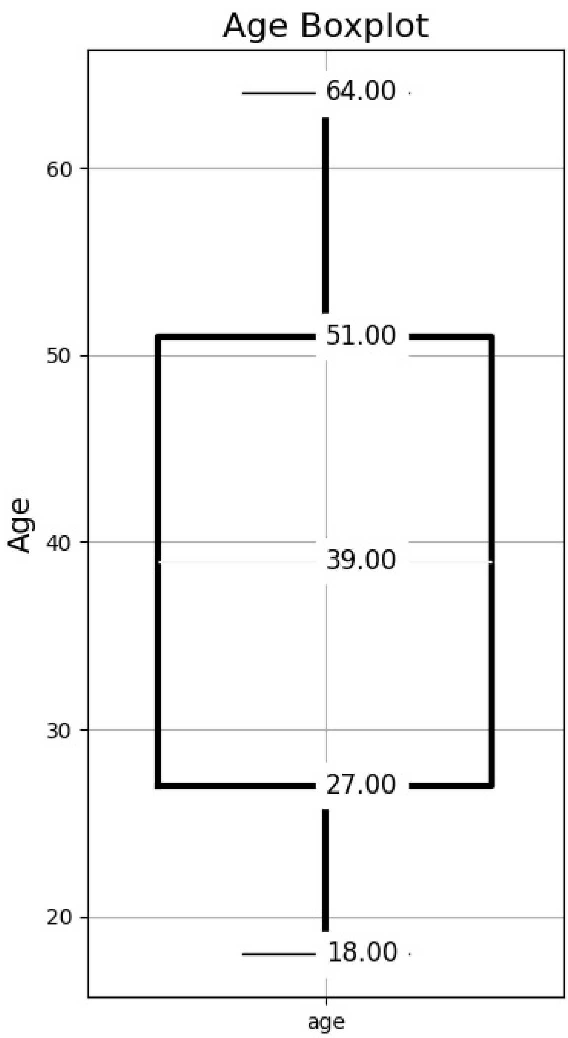
```
5 Point Summary of Age Attribute:
Age(min) : 18
Q1                    : 27.0
Q2(Median)            : 39.0
Q3                    : 51.0
Age(max) : 64
```



In [18]:
```
#Let's take a closer look at the Boxplot, and calculate the measure of skewness and totalnumber of outlier valu
def box_plot(x = 'bmi'):
    def add_values(bp, ax):
        """ This actually adds the numbers to the various points of the boxplots"""
        for element in ['whiskers', 'medians', 'caps']:
            for line in bp[element]:
                # Get the position of the element. y is the label you want
                (x_l, y),(x_r, _) = line.get_xydata()
                # Make sure datapoints exist
```

```python
            # (I've been working with intervals, should not be problem for this case)
            if not np.isnan(y):
                x_line_center = x_l + (x_r - x_l)/2
                y_line_center = y  # Since it's a line and it's horisontal
                # overlay the value:  on the line, from center to right
                ax.text(x_line_center, y_line_center, # Position
                        '%.2f' % y, # Value (3f = 3 decimal float)
                        verticalalignment='center', # Centered vertically with line
                        fontsize=12, backgroundcolor="white")

    fig, axes = plt.subplots(1, figsize=(4, 8))

    red_diamond = dict(markerfacecolor='r', marker='D')

    bp_dict = df.boxplot(column = x,
                         grid=True,
                         figsize=(4, 8),
                         ax=axes,
                         vert = True,
                         notch=False,
                         widths = 0.7,
                         showmeans = True,
                         whis = 1.5,
                         flierprops = red_diamond,
                         boxprops= dict(linewidth=3.0, color='black'),
                         whiskerprops=dict(linewidth=3.0, color='black'),
                         return_type = 'dict')

    add_values(bp_dict, axes)

    plt.title(f'{x.capitalize()} Boxplot', fontsize=16)
    plt.ylabel(f'{x.capitalize()}', fontsize=14)
    plt.show()

    skew = df[x].skew()
    Q1 = df[x].quantile(0.25)
    Q3 = df[x].quantile(0.75)
    IQR = Q3 - Q1
    total_outlier_num = ((df[x] < (Q1 - 1.5 * IQR)) | (df[x] > (Q3 + 1.5 * IQR))).sum()
    print(f'Mean {x.capitalize()} = {df[x].mean()}')
    print(f'Median {x.capitalize()} = {df[x].median()}')
    print(f'Skewness of {x}: {skew}.')
    print(f'Total number of outliers in {x} distribution: {total_outlier_num}.')
```

In [19]:
```python
box_plot('age')
```




```
Mean Age = 39.222139117427076
Median Age = 39.0
Skewness of age: 0.054780773126998195.
Total number of outliers in age distribution: 0.
```

In [20]:
```
# How many of the insured have the age of 64?
df_age_64 = df[df['age'] == 64]
print(f'Total number of insured people with the age of 64: ({len(df_age_64)}).')
```

```
Total number of insured people with the age of 64: (22).
```

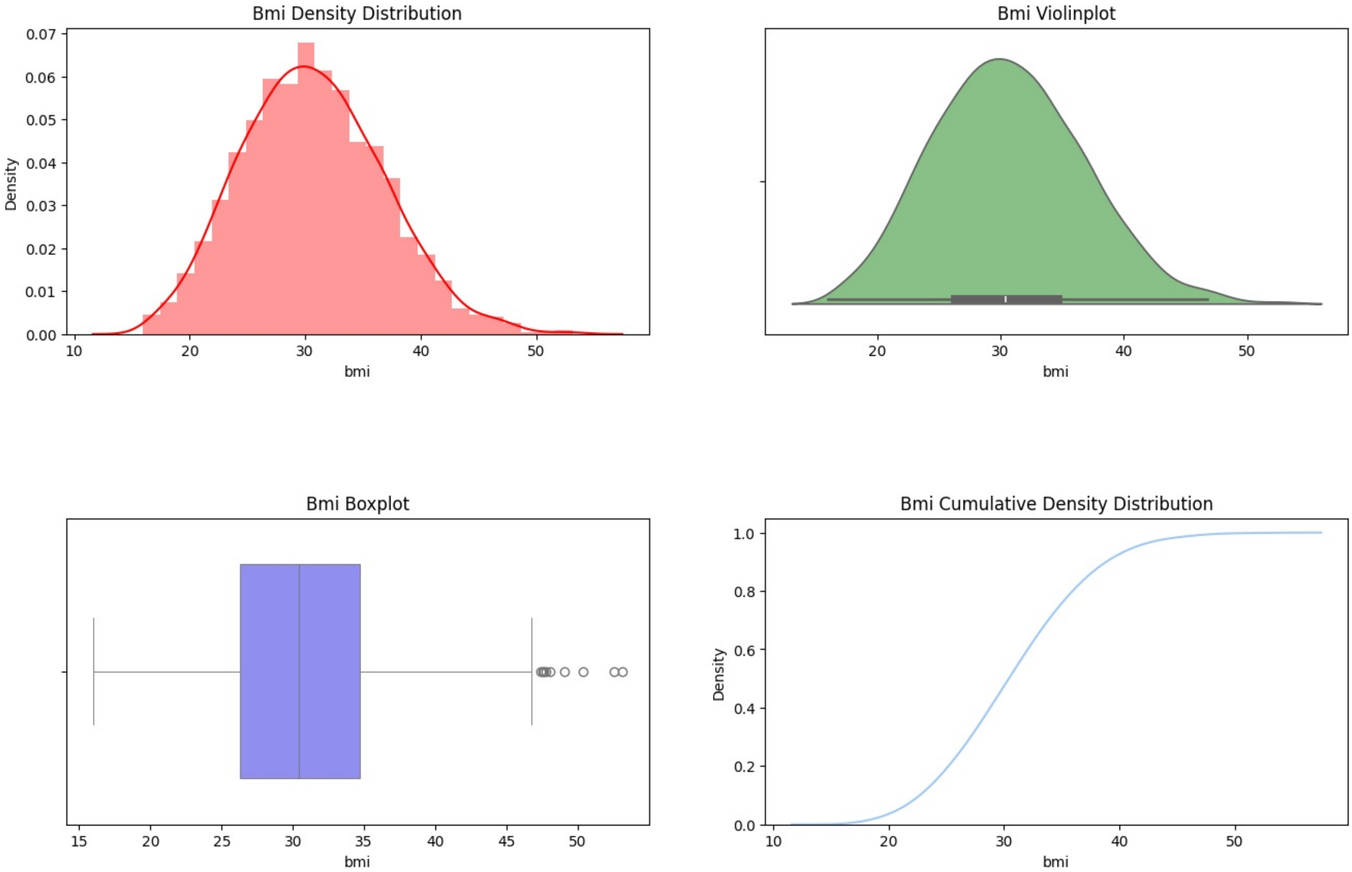
We can notice that:

- The age distribution ranges from a minimum of 18 to a maximum of 64 years.
- The median age is 39 and the Mean is 39.2 , indicating that half of the insured individuals are younger than this age.
- The first quartile (27 years) and the third quartile (51 years) suggest that the majority of insured individuals are between their late twenties and early fifties.
- Notably, there are 22 insured individuals who are exactly 64 years old, indicating a significant presence of this age group within the insured population.
- There are no outlier values in the Age distribution in the data.

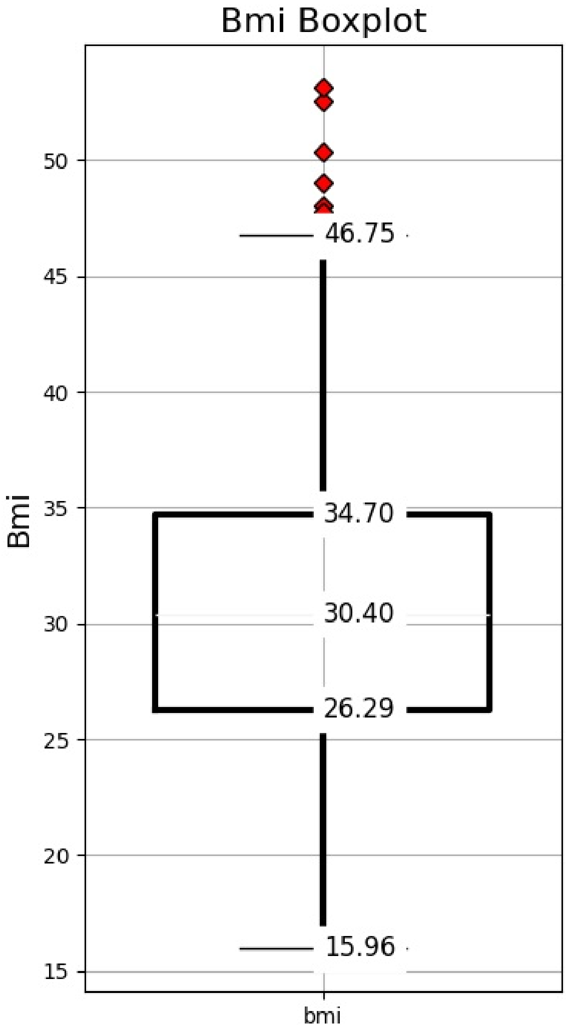
What Can We Infer from the BMI Analysis?

In [21]:
```
summary('bmi')
```

```
5 Point Summary of Bmi Attribute:
Bmi(min) : 15.96
Q1                    : 26.29
Q2(Median)            : 30.4
Q3                    : 34.7
Bmi(max) : 53.13
```



In [22]:
```
box_plot('bmi')
```




```
Mean Bmi = 30.66345175766642
Median Bmi = 30.4
Skewness of bmi: 0.28391419385321137.
Total number of outliers in bmi distribution: 9.
```

Who is the insured with the highest BMI, and how does his charges compare to the rest?

In [23]:
```
df[df['bmi'] == df['bmi'].max()]
```

Out[23]:

	age	sex	bmi	children	smoker	region	charges
1317	18	male	53.13	0	no	southeast	1163.4627

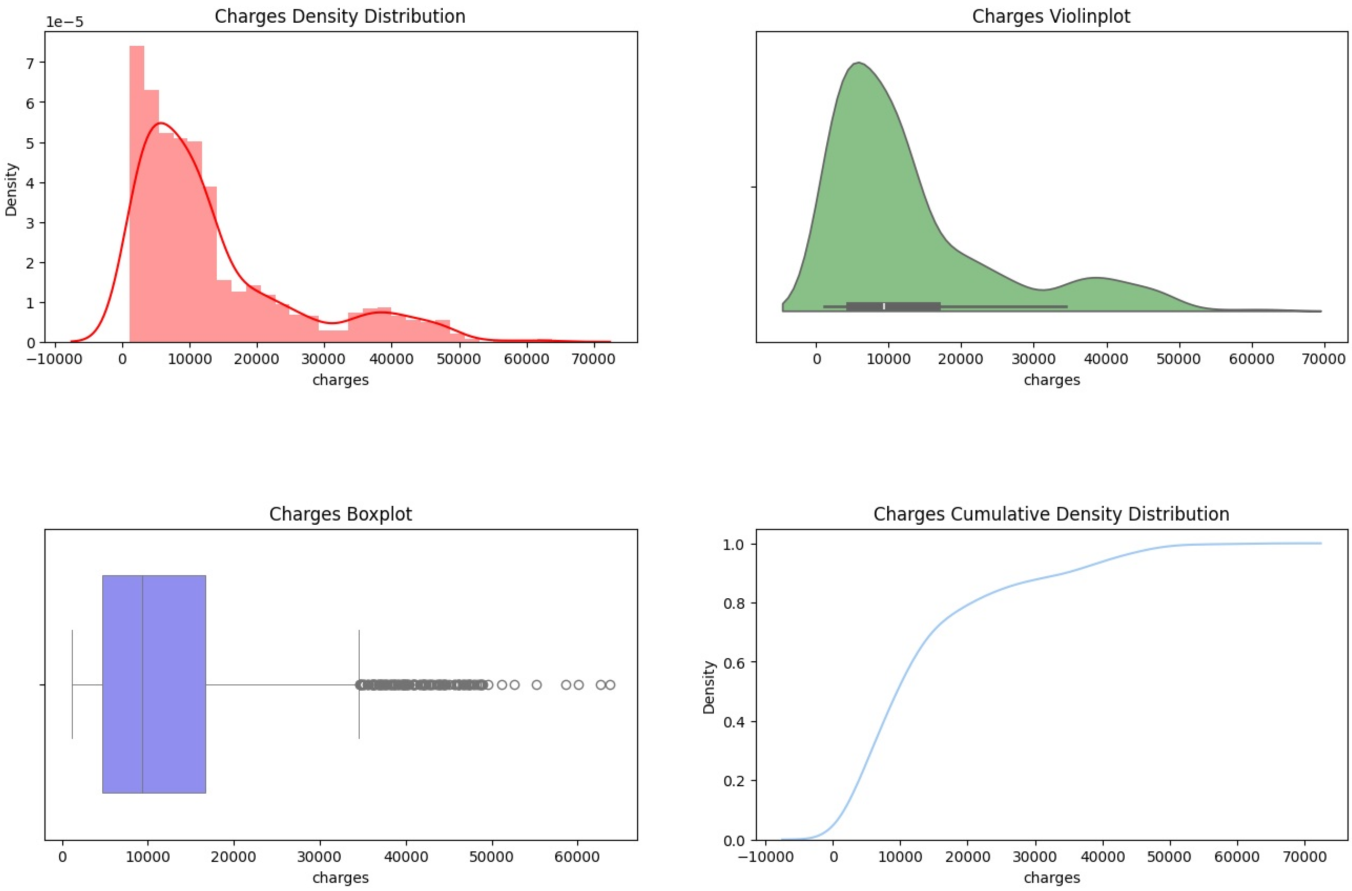
We can notice that:

- The BMI distribution of the Insured approximately follows a normal distribution with a Mean of 30.66 and Median of 30.4.
- There are a total of 9 outlier values in the BMI distribution, all in the higher side. The highest BMI observed is 53.13.
- The insured individual with the highest BMI (53.13) is 18 years old and doesn't have children, paying 1163.46 in charges. This reflects common underwriting practices where age and health metrics influence insurance premiums.

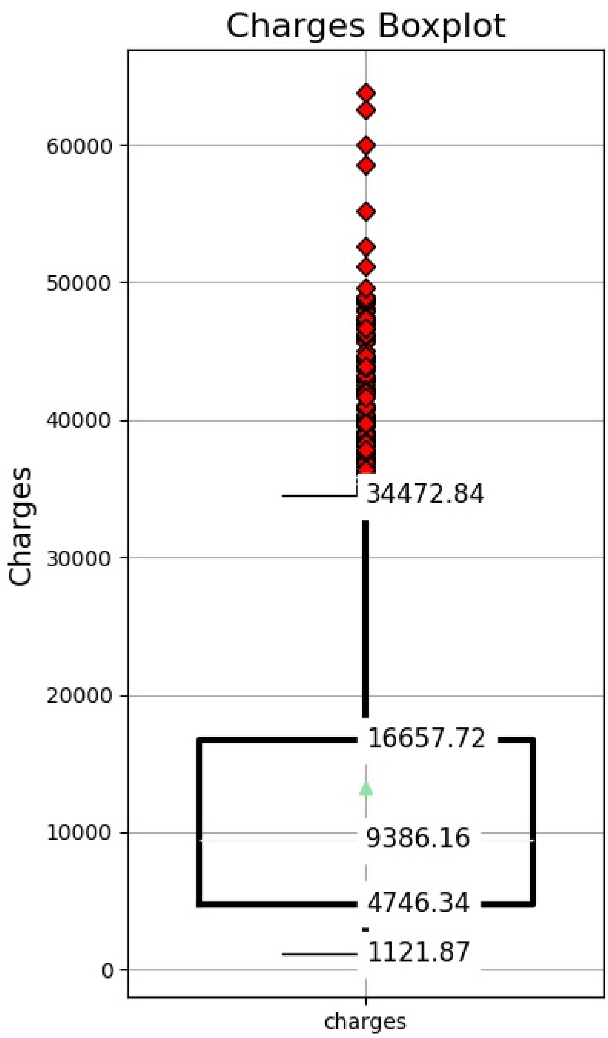
What Can We Infer from the Charges Analysis?

In [24]:
```
summary('charges')
```

```
5 Point Summary of Charges Attribute:
Charges(min) : 1121.8739
Q1                    : 4746.344
Q2(Median)            : 9386.1613
Q3                    : 16657.71745
Charges(max) : 63770.42801
```



In [25]:
```
box_plot('charges')
```




```
Mean Charges = 13279.121486655948
Median Charges = 9386.1613
Skewness of charges: 1.5153909108403483.
Total number of outliers in charges distribution: 139.
```

Who is paying the highest charges?

In [26]:
```
df[df['charges'] == df['charges'].max()]
```

Out[26]:

	age	sex	bmi	children	smoker	region	charges
543	54	female	47.41	0	yes	southeast	63770.42801

Who is the insured with the highest BMI, and how does his charges compare to the rest?

In [27]:
```
df[df['bmi'] == df['bmi'].max()]
```

Out[27]:

	age	sex	bmi	children	smoker	region	charges
1317	18	male	53.13	0	no	southeast	1163.4627

In [28]:
```
df[df['charges'] == df['charges'].median()]
```

Out[28]:

	age	sex	bmi	children	smoker	region	charges
782	51	male	35.97	1	no	southeast	9386.1613

We can notice that:

- The distribution of charges is heavily right-skewed (mean > median), with a mean of 13,279.12 and a median of 9,386.16, indicating a few individuals with very high charges significantly influence the average.
- The lowest charge recorded is 1,121.87, while the highest is a substantial 63,770.43, highlighting a significant range in premium payments.
- Out of 1,337 data points, there are 139 outlier values, all of which are in the higher end of the distribution, indicating a few individuals face exceptionally high charges.
- The insured individual with the highest charges (63,770.43) is a 54-year-old female No-smoker with a high BMI.
- The person with the highest BMI (obese, or least healthy, based on available data) is also one of the youngest (male, 18, non-smoker.) He is paying less premium charges than the mean(which, we note, is affected by extreme outlier values of charges like the

person above), but significantly more than the median. This is in line with our basic understanding of underwriting rules.

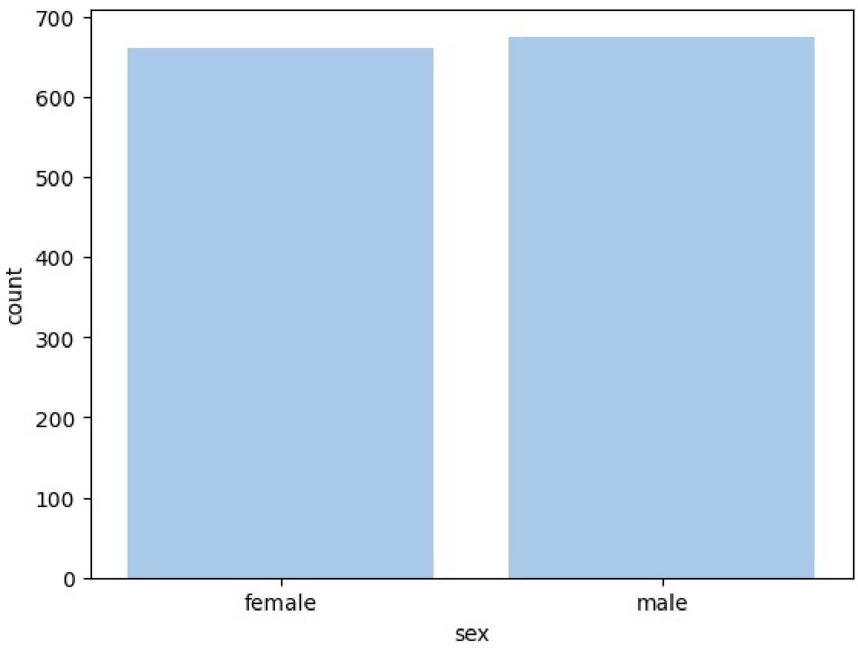
Explore the distribution of categorical variables: `Sex` , `Smoker` , `Region` and `Children`

In [29]:
```
# Create a function that returns a Pie chart for categorical variable:
def pie_chart(x = 'smoker'):
    """
    Function creates a Pie chart for categorical variables.
    """
    fig, ax = plt.subplots(figsize=(8, 6), subplot_kw=dict(aspect="equal"))

    s = df.groupby(x).size()

    mydata_values = s.values.tolist()
    mydata_index = s.index.tolist()

    def func(pct, allvals):
        absolute = int(pct/100.*np.sum(allvals))
        return "{:.1f}%\n({:d})".format(pct, absolute)


    wedges, texts, autotexts = ax.pie(mydata_values, autopct=lambda pct: func(pct, mydata_values),
                                      textprops=dict(color="w"))

    ax.legend(wedges, mydata_index,
              title="Index",
              loc="center left",
              bbox_to_anchor=(1, 0, 0.5, 1))

    plt.setp(autotexts, size=12, weight="bold")

    ax.set_title(f'{x.capitalize()} Piechart')

    plt.show()
```

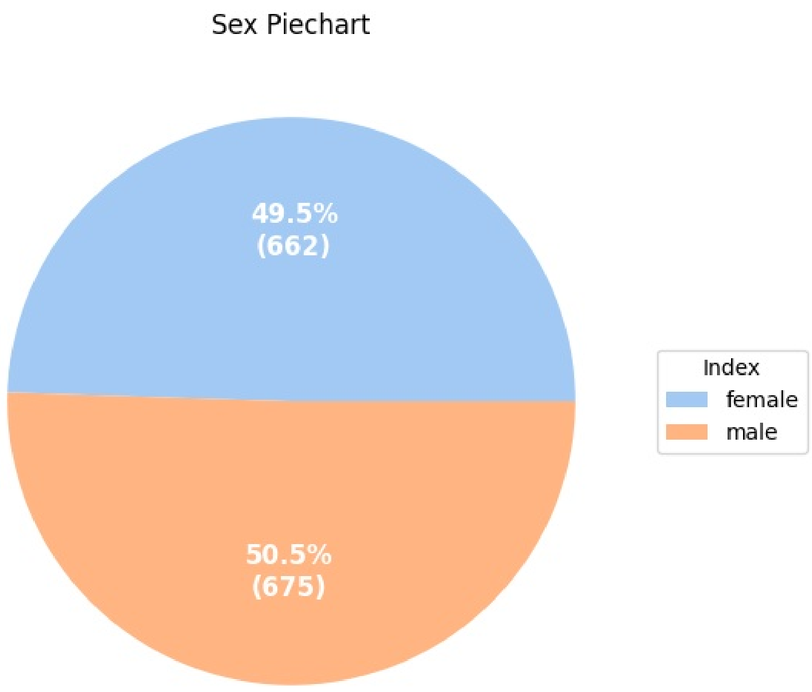
Sex Distribution

In [30]:
```
sns.countplot(x = 'sex', data = df)
```

Out[30]: <Axes: xlabel='sex', ylabel='count'>



In [31]:
```
pie_chart('sex')
```



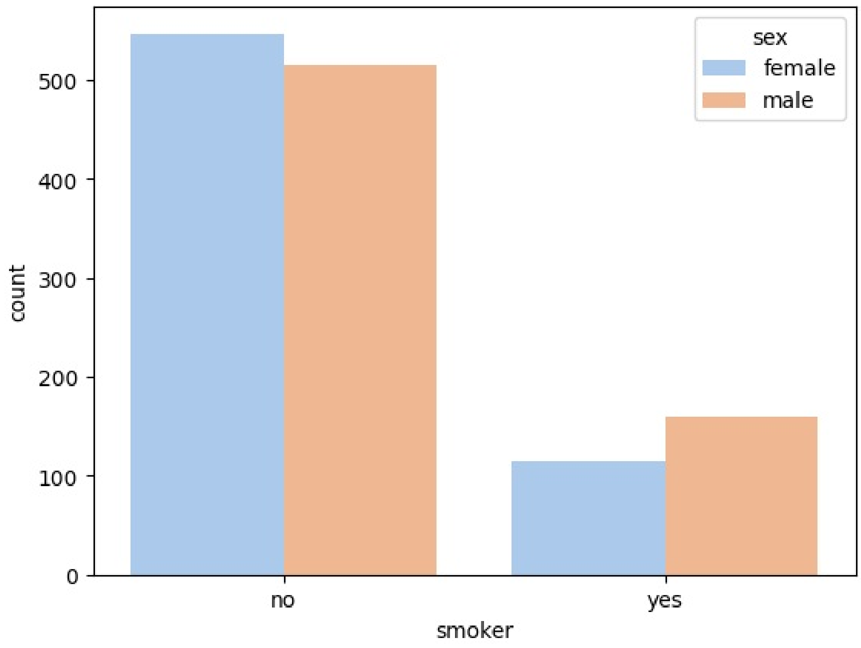
We can notice that:

- The dataset is almost evenly distributed among genders, with 675 Males (50.5%) and 662 Females (49.5%).

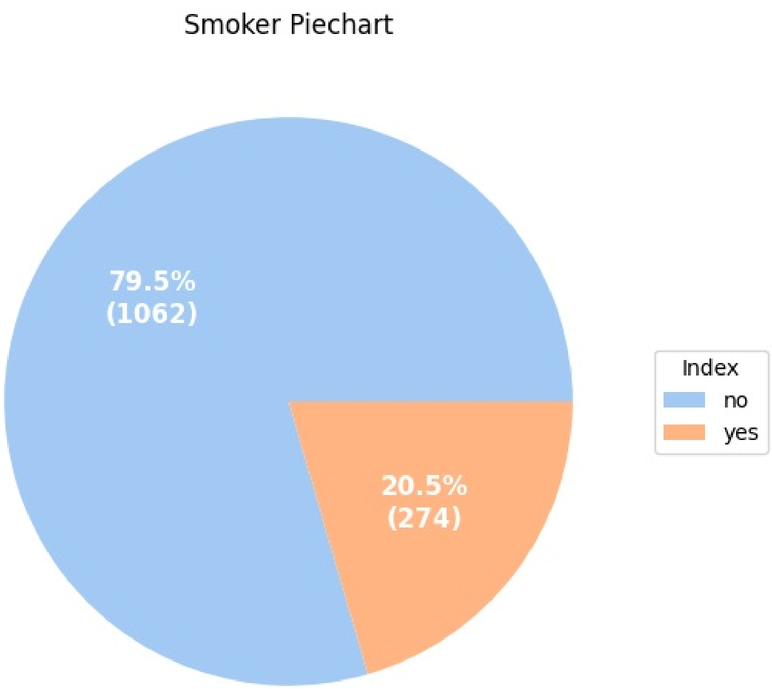
Smokers Distribution

In [32]:
```
sns.countplot(x = 'smoker', hue = 'sex', data = df)
```

Out[32]:
```
<Axes: xlabel='smoker', ylabel='count'>
```



In [33]:
```
pie_chart('smoker')
```



Are average premium charges for smokers significantly higher than non-smokers?

In [34]:
```
df['charges'].groupby(df['smoker']).mean()
```

Out[34]:
```
smoker
no      8440.660307
yes    32050.231832
Name: charges, dtype: float64
```

In [35]:
```
df.groupby(['smoker', 'sex']).agg('count')
```

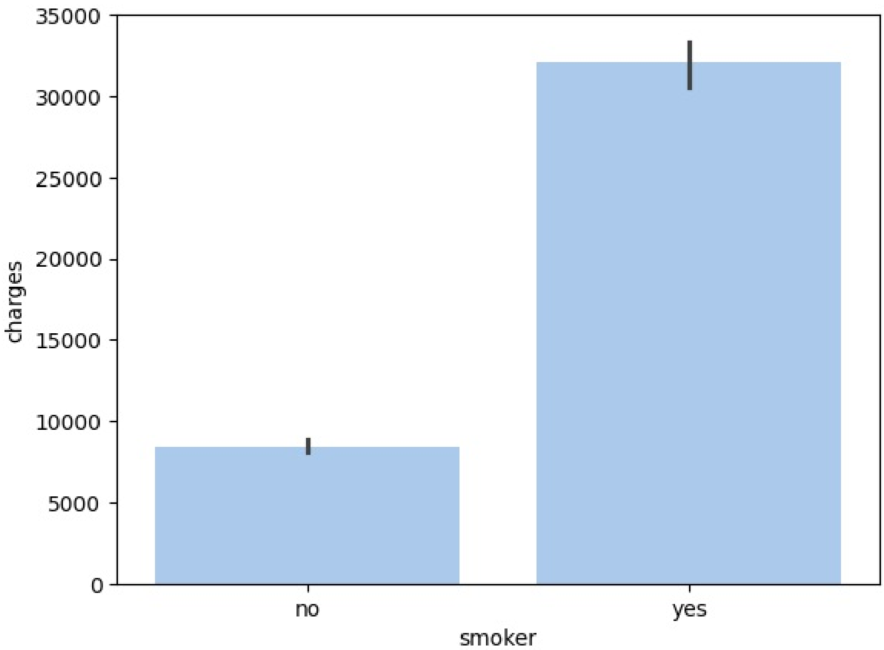
Out[35]:

smoker	sex	age	bmi	children	region	charges
no	female	547	547	547	547	547
	male	516	516	516	516	516
yes	female	115	115	115	115	115
	male	159	159	159	159	159

- yes, average premium charges for smokers are indeed significantly higher than non-smokers.

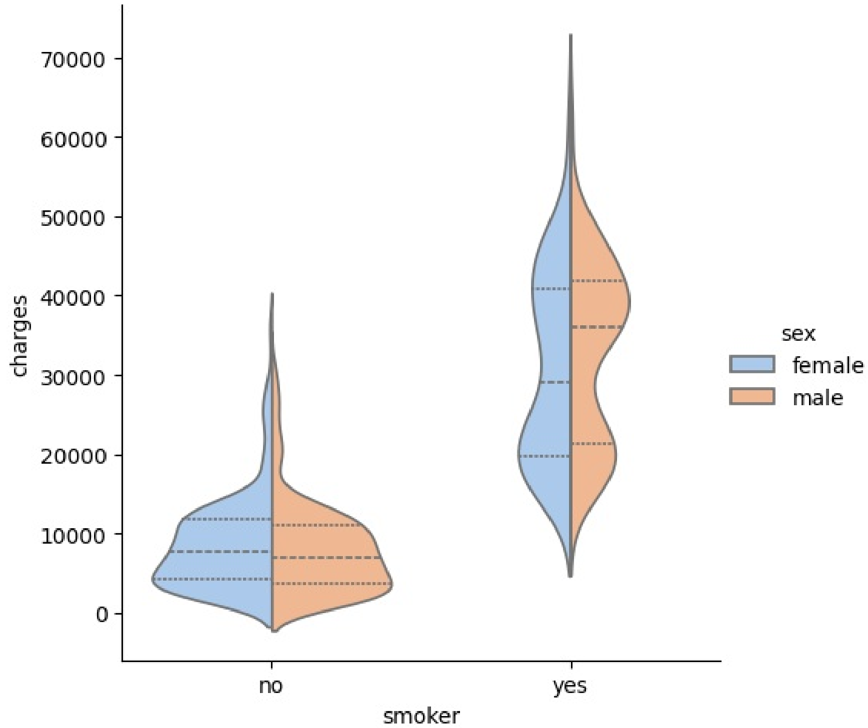
In [36]:
```
sns.barplot(x = "smoker", y = "charges", data = df)
```

Out[36]:
```
<Axes: xlabel='smoker', ylabel='charges'>
```



In [37]:
```
sns.catplot(x="smoker", y="charges", hue="sex",
            kind="violin", inner="quartiles", split=True,
            palette="pastel", data=df)
```

Out[37]: <seaborn.axisgrid.FacetGrid at 0x22dc7867050>



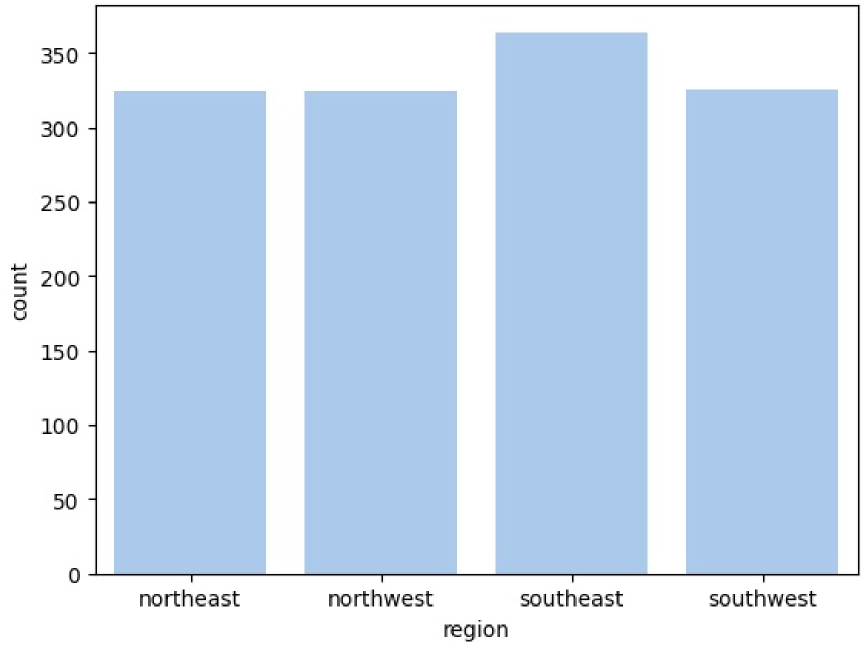
We can notice that:

- Of the total 1337 insured, 274 (20.5%) are smokers and the rest are non-smokers.
- Among 274 smokers, proportion of males (159) are higher than females (115).
- The average insurance premium for smokers are significantly higher than non-smokers.

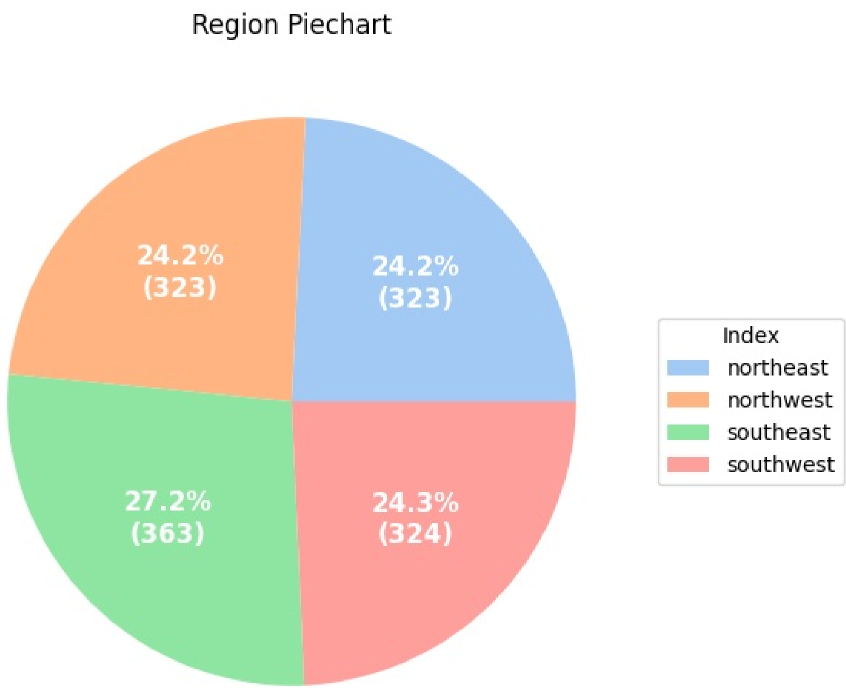
Regions Distribution

In [38]:
```
sns.countplot(x = 'region', data = df)
```

Out[38]:
```
<Axes: xlabel='region', ylabel='count'>
```




```
In [39]: pie_chart('region')
```



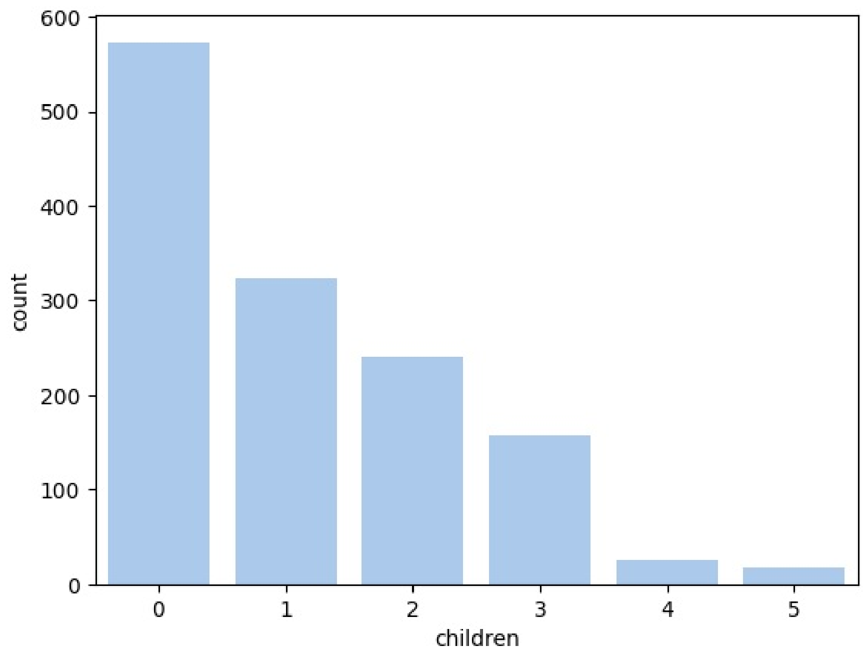
We can notice that:

- All four regions are represented approximately evenly in the dataset.

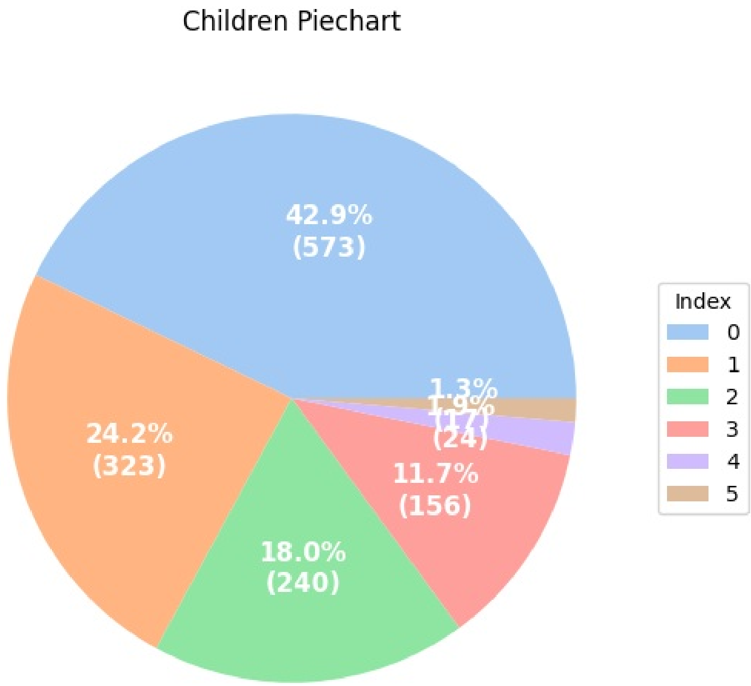
Number of children

```
In [40]: sns.countplot(x = 'children', data = df)
```

```
Out[40]: <Axes: xlabel='children', ylabel='count'>
```



In [41]:
```
pie_chart('children')
```



In [42]:
```
df.groupby(['children']).agg('count')['age']
```

Out[42]:
```
children
0    573
1    324
2    240
3    157
4     25
5     18
Name: age, dtype: int64
```

We can notice that:

- In the dataset, approximately 85% (1137/1337) of the insured have less than 3 children.

In [43]:
```
# Next, we select all columns of the dataFrame with datatype = category:
cat_columns = df.select_dtypes(['category']).columns
cat_columns
```

Out[43]:
```
Index(['sex', 'children', 'smoker', 'region'], dtype='object')
```

In [44]:
```
# Finally, we transform the original columns by replacing the elements with their category codes:
```

```
df[cat_columns] = df[cat_columns].apply(lambda x: x.cat.codes)
df.head()
```

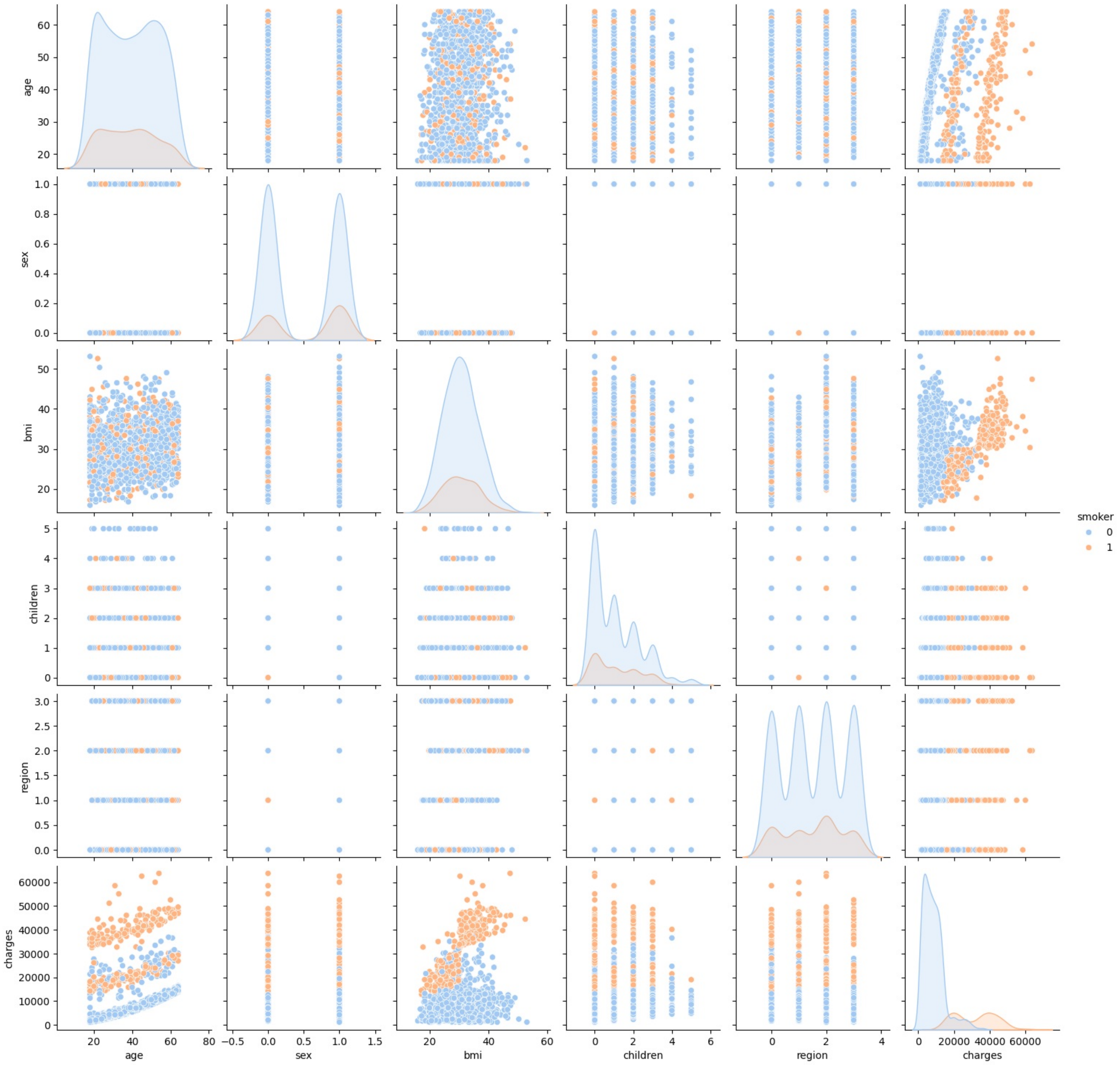
Out[44]:

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.900	0	1	3	16884.92400
1	18	1	33.770	1	0	2	1725.55230
2	28	1	33.000	3	0	2	4449.46200
3	33	1	22.705	0	0	1	21984.47061
4	32	1	28.880	0	0	1	3866.85520

In [45]:

```
# Now we can plot all columns of our dataset in a pairplot!
sns.pairplot(df, hue  = 'smoker')
```

Out[45]: `<seaborn.axisgrid.PairGrid at 0x22dc7a41bb0>`

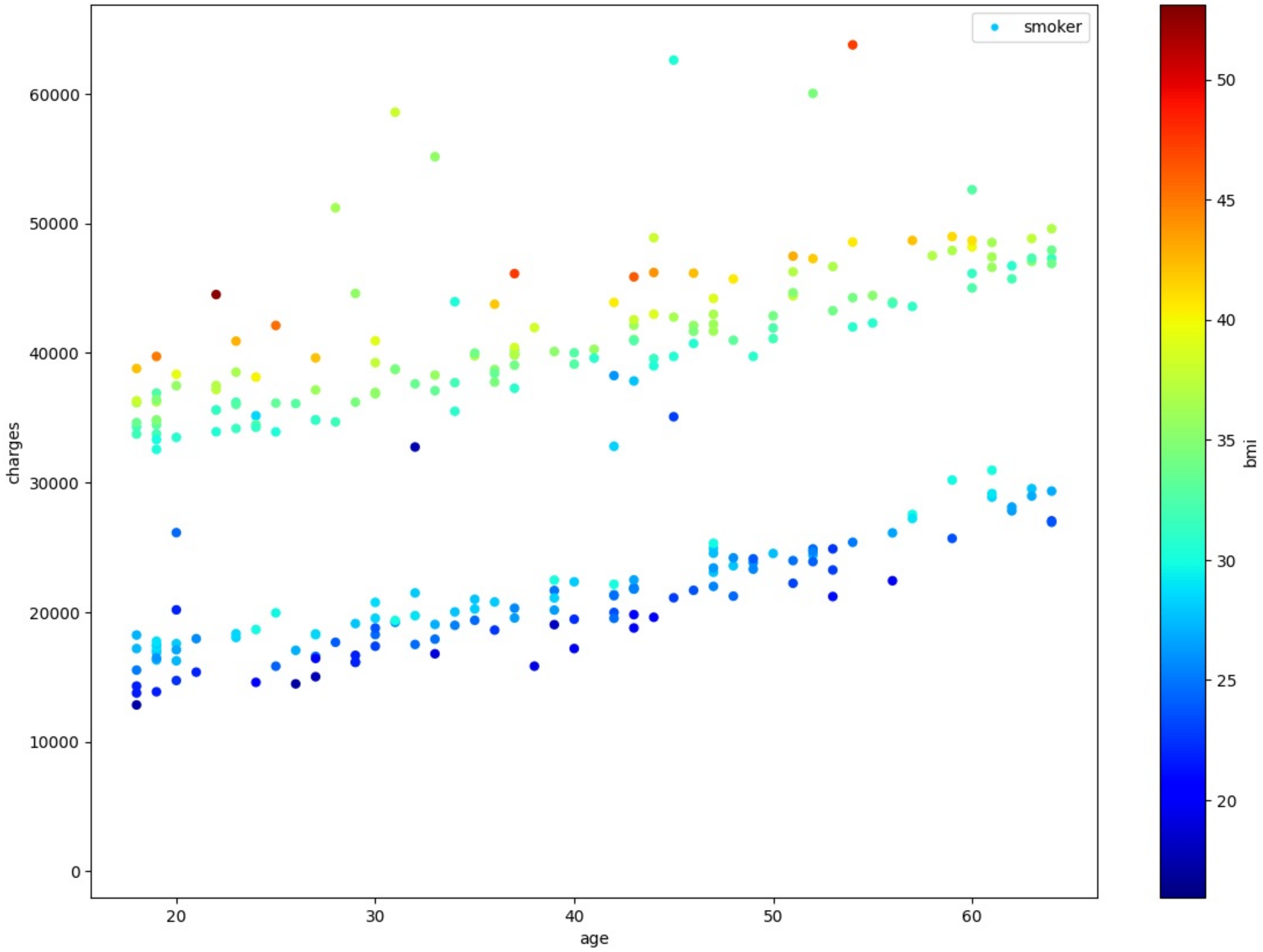


A particularly interesting relationship between insurance premium charges, BMI and smoking status(Smoker/Non-smoker) can be seen in this graph:

In [46]:

```
df.plot(kind="scatter", x="age", y="charges",
    s=df["smoker"]*25, label="smoker", figsize=(14,10),
    c='bmi', cmap=plt.get_cmap("jet"), colorbar=True,
    sharex=False)
plt.legend()
```

Out[46]: `<matplotlib.legend.Legend at 0x22dc72a20f0>`



In [47]:
```
corr = df.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(corr, annot=True, cmap = 'summer_r')
```

Out[47]: <Axes: >

- From the correlation heatmap, we can conclude that the premium charges show a weak positive correlation with Age and BMI of the insured, and a strong positive correlation with smoking habit.

Conclusion

- our dataset consists of 1337 rows with 7 columns, and has no NaNs values.
- The age distribution ranges from a minimum of 18 to a maximum of 64 years.
- The median age is 39 and the Mean is 39.2 , indicating that half of the insured individuals are younger than this age.
- The first quartile (27 years) and the third quartile (51 years) suggest that the majority of insured individuals are between their late twenties and early fifties.
- Notably, there are 22 insured individuals who are exactly 64 years old, indicating a significant presence of this age group within the insured population.
- There are no outlier values in the Age distribution in the data.
- The BMI distribution of the Insured approximately follows a normal distribution with a Mean of 30.66 and Median of 30.4.
- There are a total of 9 outlier values in the BMI distribution, all in the higher side. The highest BMI observed is 53.13.
- The insured individual with the highest BMI (53.13) is 18 years old and doesn't have children, paying 1163.46 in charges. This reflects common underwriting practices where age and health metrics influence insurance premiums.
- The distribution of charges is heavily right-skewed (mean > median), with a mean of 13,279.12 and a median of 9,386.16, indicating a few individuals with very high charges significantly influence the average.
- The lowest charge recorded is 1,121.87, while the highest is a substantial 63,770.43, highlighting a significant range in premium payments.
- Out of 1,337 data points, there are 139 outlier values, all of which are in the higher end of the distribution, indicating a few individuals face exceptionally high charges.
- The insured individual with the highest charges (63,770.43) is a 54-year-old female No-smoker with a high BMI.
- The person with the highest BMI (obese, or least healthy, based on available data) is also one of the youngest (male, 18, non-smoker.) He is paying less premium charges than the mean(which, we note, is affected by extreme outlier values of charges like the person above), but significantly more than the median. This is in line with our basic understanding of underwriting rules.
- The dataset is almost evenly distributed among genders, with 675 Males (50.5%) and 662 Females (49.5%).
- Average premium charges for smokers are indeed significantly higher than non-smokers.
- Of the total 1337 insured, 274 (20.5%) are smokers and the rest are non-smokers.
- Among 274 smokers, proportion of males (159) are higher than females (115).
- The average insurance premium for smokers are significantly higher than non-smokers.
- All four regions are represented approximately evenly in the dataset.
- In the dataset, approximately 85% (1137/1337) of the insured have less than 3 children.
- We can conclude that the premium charges show a weak positive correlation with Age and BMI of the insured, and a strong positive correlation with smoking habit.

> **"This project was entirely developed by **Bassam El-Shoraa"****.