EDA

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
data = pd.read_csv('./insurance.csv')
```

In [3]:
```
data.head()
```

Out[3]:

	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 [4]:
```
data.tail()
```

Out[4]:

	age	sex	bmi	children	smoker	region	charges
1333	50	male	30.97	3	no	northwest	10600.5483
1334	18	female	31.92	0	no	northeast	2205.9808
1335	18	female	36.85	0	no	southeast	1629.8335
1336	21	female	25.80	0	no	southwest	2007.9450
1337	61	female	29.07	0	yes	northwest	29141.3603

In [5]:
```
data.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
```

In [6]:
```
data.isnull().mean()*100
```

Out[6]:
```
age         0.0
sex         0.0
bmi         0.0
children    0.0
smoker      0.0
region      0.0
charges     0.0
dtype: float64
```

In [7]:
```
data.duplicated().sum()
```

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

In [8]:
```
data.drop_duplicates(inplace=True)
```

In [9]:
```
data.describe()
```

Out[9]:

	age	bmi	children	charges
count	1337.000000	1337.000000	1337.000000	1337.000000
mean	39.222139	30.663452	1.095737	13279.121487
std	14.044333	6.100468	1.205571	12110.359656
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.290000	0.000000	4746.344000
50%	39.000000	30.400000	1.000000	9386.161300
75%	51.000000	34.700000	2.000000	16657.717450
max	64.000000	53.130000	5.000000	63770.428010

In [10]:
```
data.shape
```

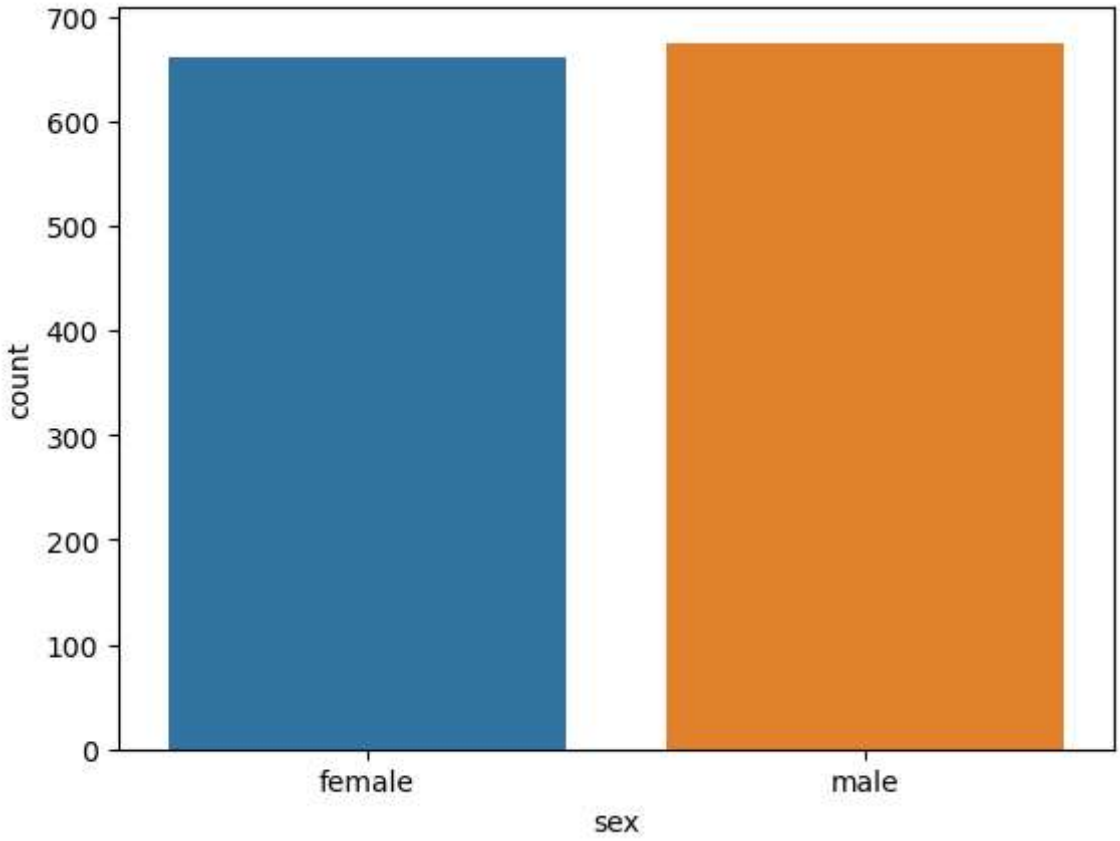
Out[10]:
```
(1337, 7)
```

In [11]:
```
data.columns
```

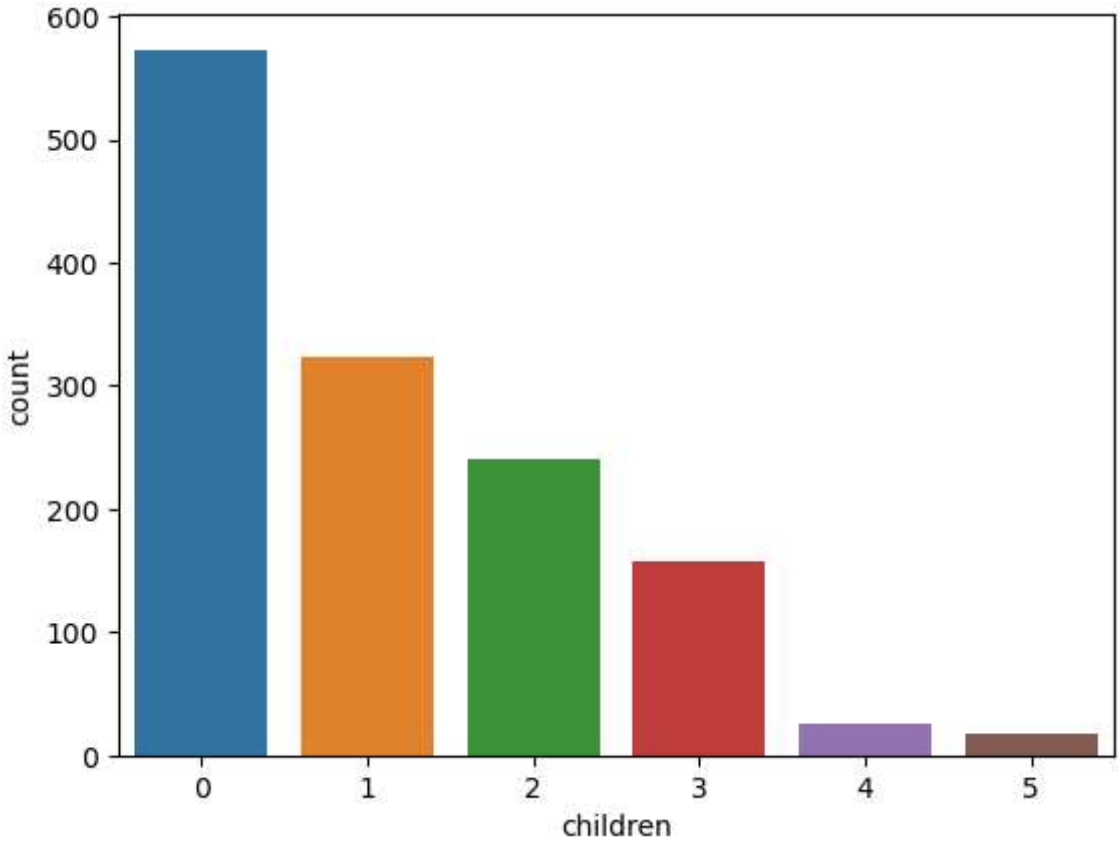
Out[11]:
```
Index(['age', 'sex', 'bmi', 'children', 'smoker', 'region', 'charges'], dtype
='object')
```

EDA

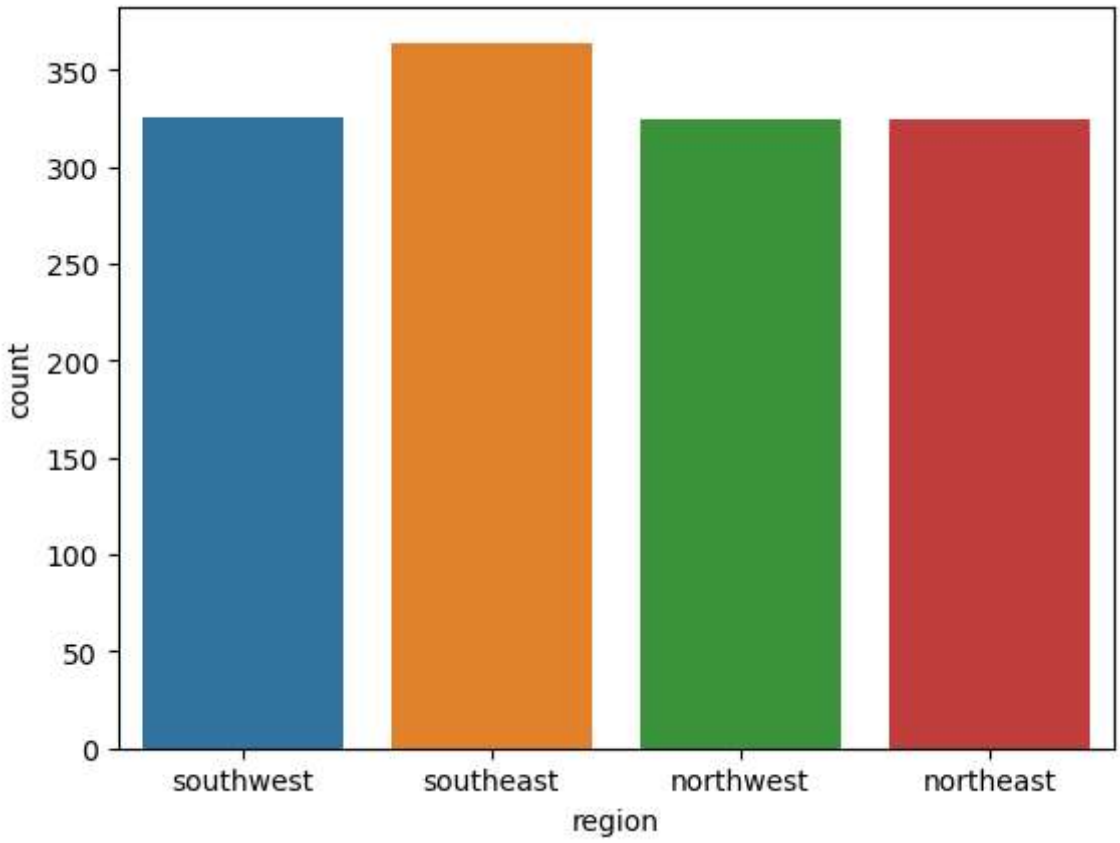
In [12]:
```
sns.countplot(x='sex', data=data);
```



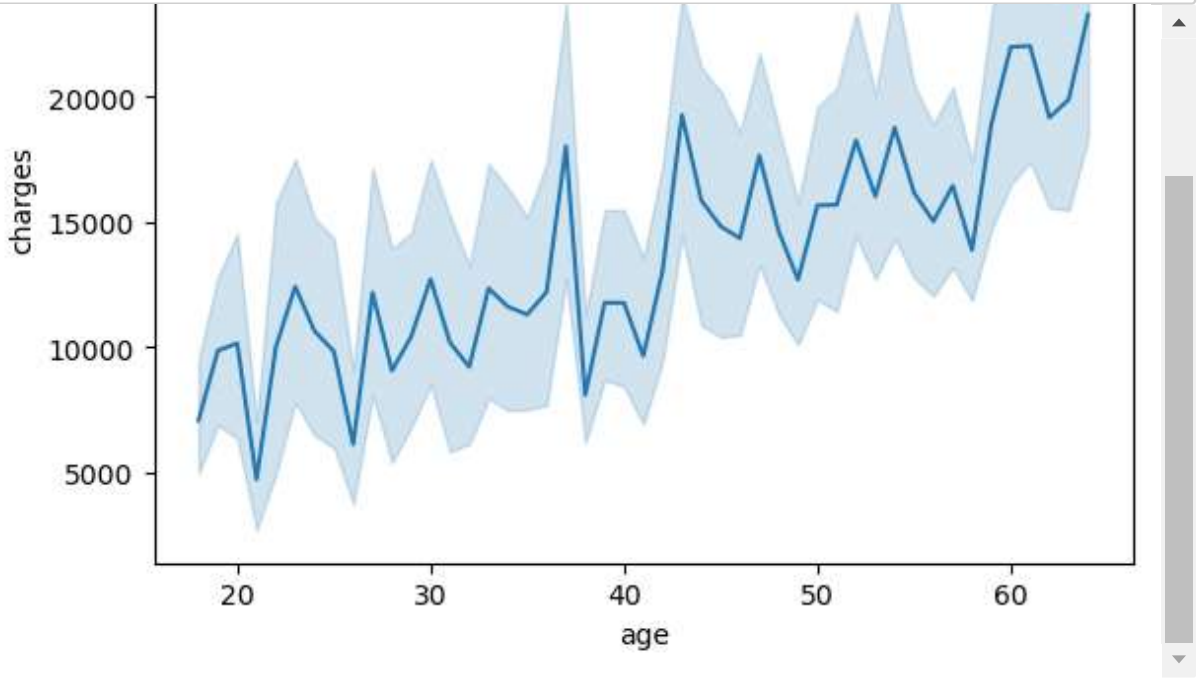
In [13]:
```
sns.countplot(x='children', data=data);
```



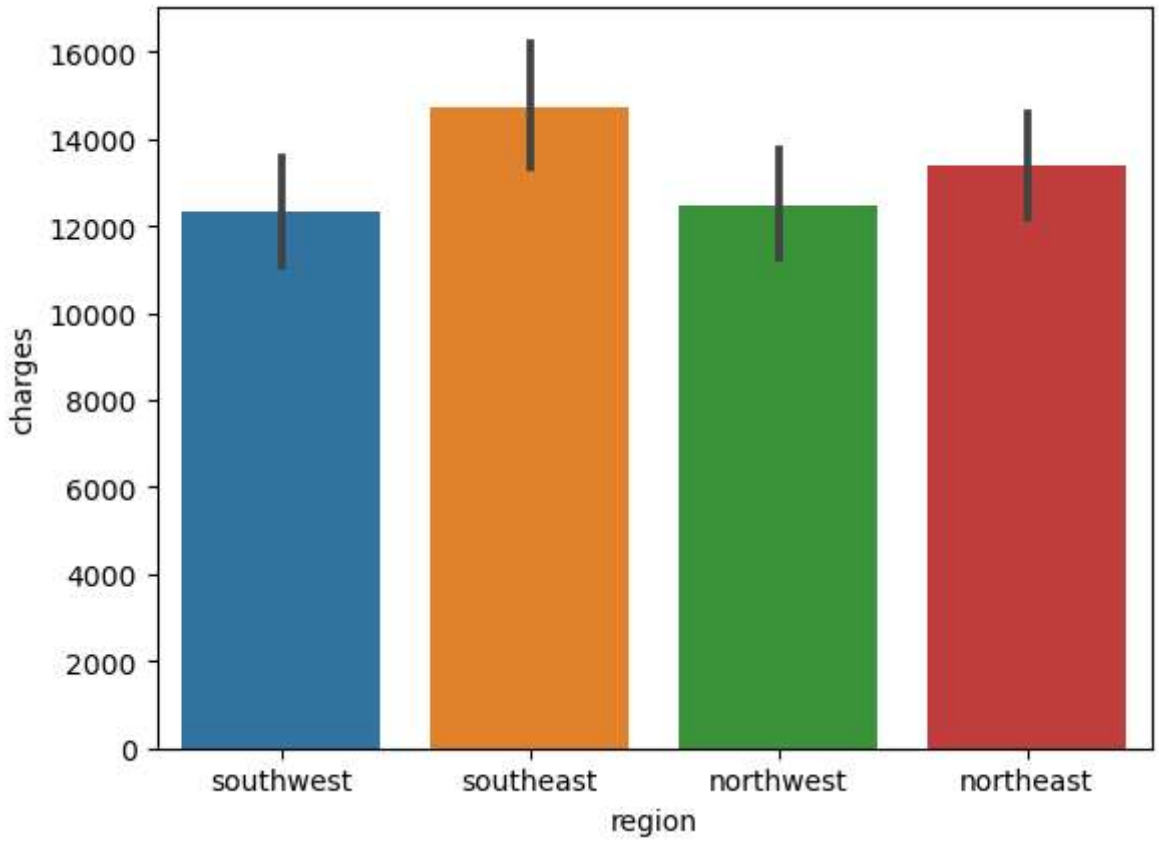
In [14]:
```
sns.countplot(x='region', data=data);
```



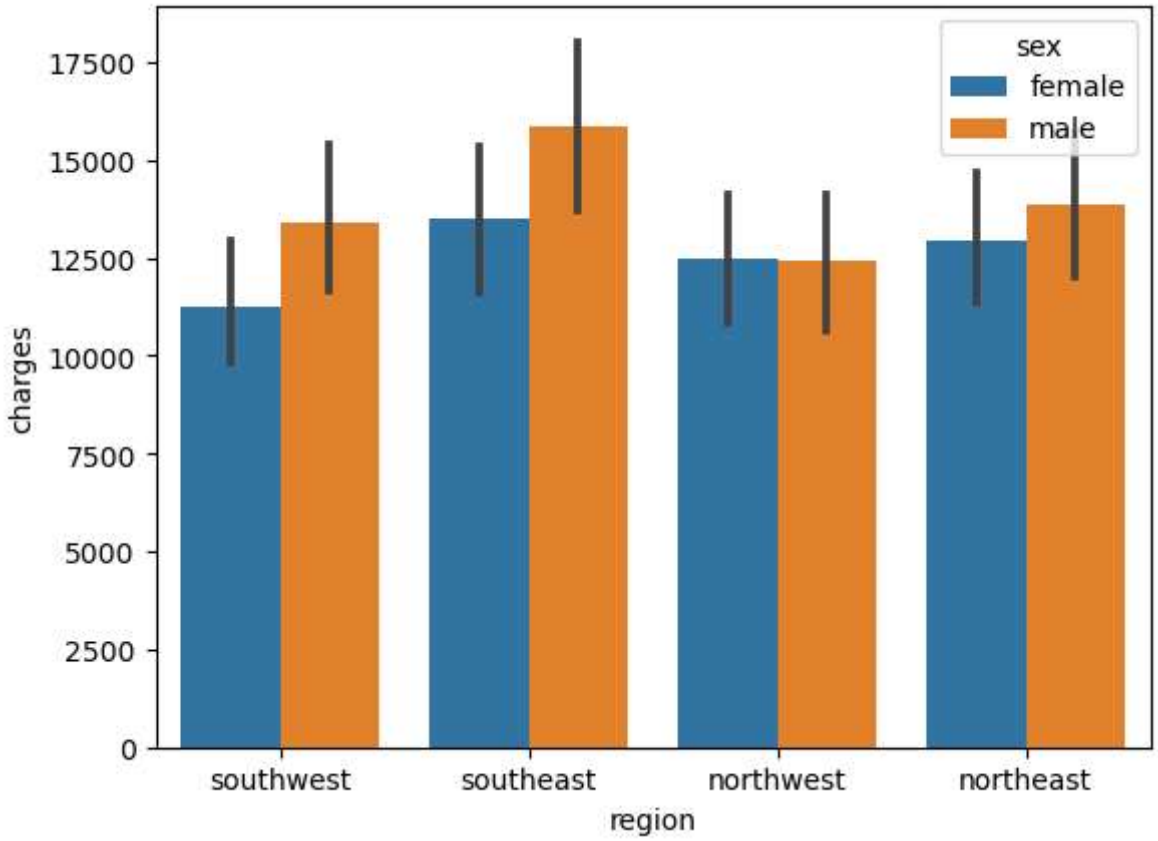
In [15]:
```
sns.lineplot(x='age', y = 'charges', data=data);
```



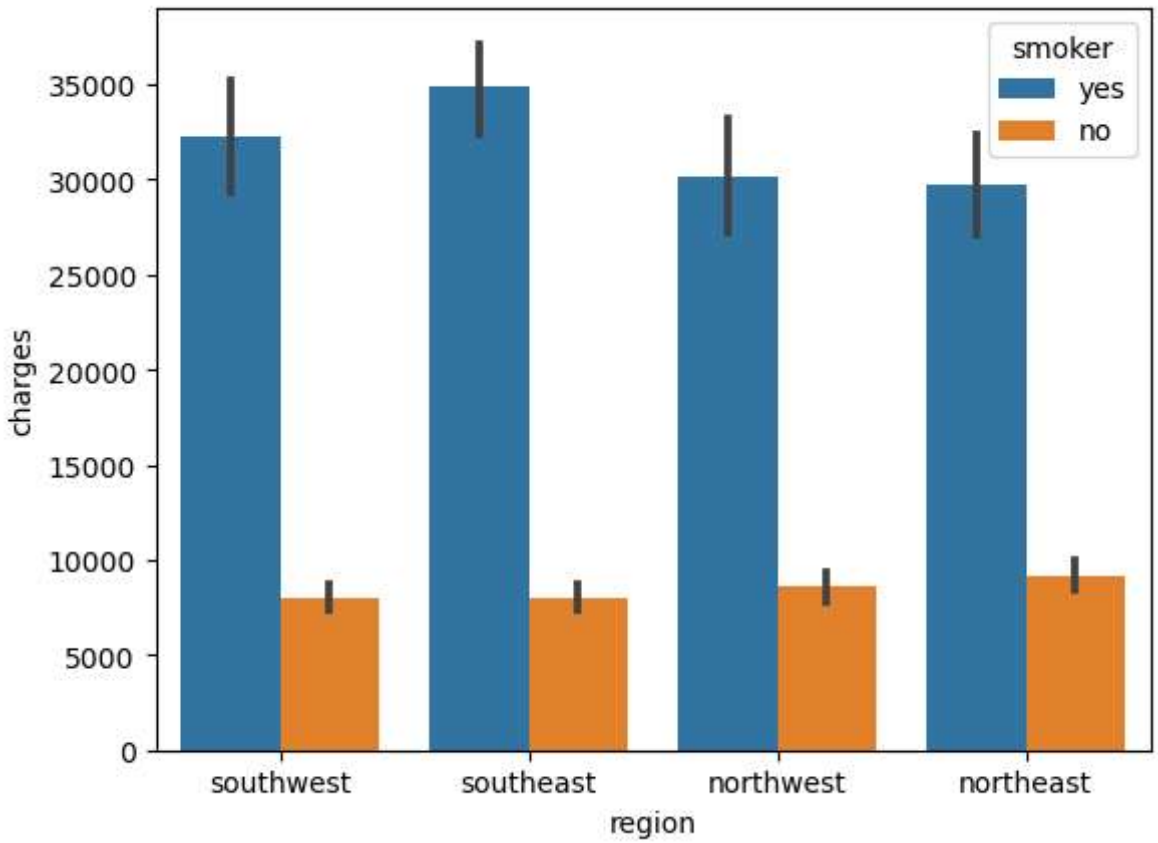
In [16]:
```
sns.barplot(x='region', y='charges', data=data);
```



In [17]:
```
sns.barplot(x='region', y='charges',hue='sex', data=data);
```



In [18]:
```
sns.barplot(x='region', y='charges',hue='smoker', data=data);
```



In [19]:
```
data
```

	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
...	...	...	...	...	...	...	...
1333	50	male	30.970	3	no	northwest	10600.54830
1334	18	female	31.920	0	no	northeast	2205.98080
1335	18	female	36.850	0	no	southeast	1629.83350
1336	21	female	25.800	0	no	southwest	2007.94500
1337	61	female	29.070	0	yes	northwest	29141.36030

1337 rows × 7 columns

Descriptive method

Ask question and find the answers from the dataset

1. In which region there is maximum amount of smokers present ?

To find out the answer lets sort the data frame by only smokers, then take the count of region.

In [20]:
```
data_smoker = data.query("smoker == 'yes'")
```

In [21]:
```
data_smoker.region.value_counts()
```

Out[21]:
```
southeast    91
northeast    67
southwest    58
northwest    58
Name: region, dtype: int64
```