

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
In [2]: df = pd.read_csv(r'F:\Python\medical_cost.csv')
```

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

Out[3]:		Id	age	sex	bmi	children	smoker	region	charges
	0	1	19	female	27.900	0	yes	southwest	16884.92400
	1	2	18	male	33.770	1	no	southeast	1725.55230
	2	3	28	male	33.000	3	no	southeast	4449.46200
	3	4	33	male	22.705	0	no	northwest	21984.47061
	4	5	32	male	28.880	0	no	northwest	3866.85520

Converting ages into categorical group based on certain age group using pd.cut method

```
In [4]: bins = [12, 20, 59, 100] # Age group boundaries
labels = ['Teen', 'Adult', 'Senior'] # Age group labels
```

```
In [5]: df['Age Group'] = pd.cut(df['age'], bins=bins, labels=labels, right=False)
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  --
0   Id           1338 non-null    int64
1   age          1338 non-null    int64
2   sex          1338 non-null    object
3   bmi          1338 non-null    float64
4   children     1338 non-null    int64
5   smoker       1338 non-null    object
6   region       1338 non-null    object
7   charges      1338 non-null    float64
8   Age Group    1338 non-null    category
dtypes: category(1), float64(2), int64(3), object(3)
memory usage: 85.2+ KB
```

```
In [7]: df.shape
```

```
Out[7]: (1338, 9)
```

```
In [8]: df.head()
```

Out[8]:		Id	age	sex	bmi	children	smoker	region	charges	Age Group
	0	1	19	female	27.900	0	yes	southwest	16884.92400	Teen
	1	2	18	male	33.770	1	no	southeast	1725.55230	Teen
	2	3	28	male	33.000	3	no	southeast	4449.46200	Adult
	3	4	33	male	22.705	0	no	northwest	21984.47061	Adult
	4	5	32	male	28.880	0	no	northwest	3866.85520	Adult

```
In [9]: df['smoker'].value_counts()
```

```
Out[9]: no      1064
yes       274
Name: smoker, dtype: int64
```

```
In [10]: df['sex'].value_counts()
```

```
Out[10]: male      676
female    662
Name: sex, dtype: int64
```

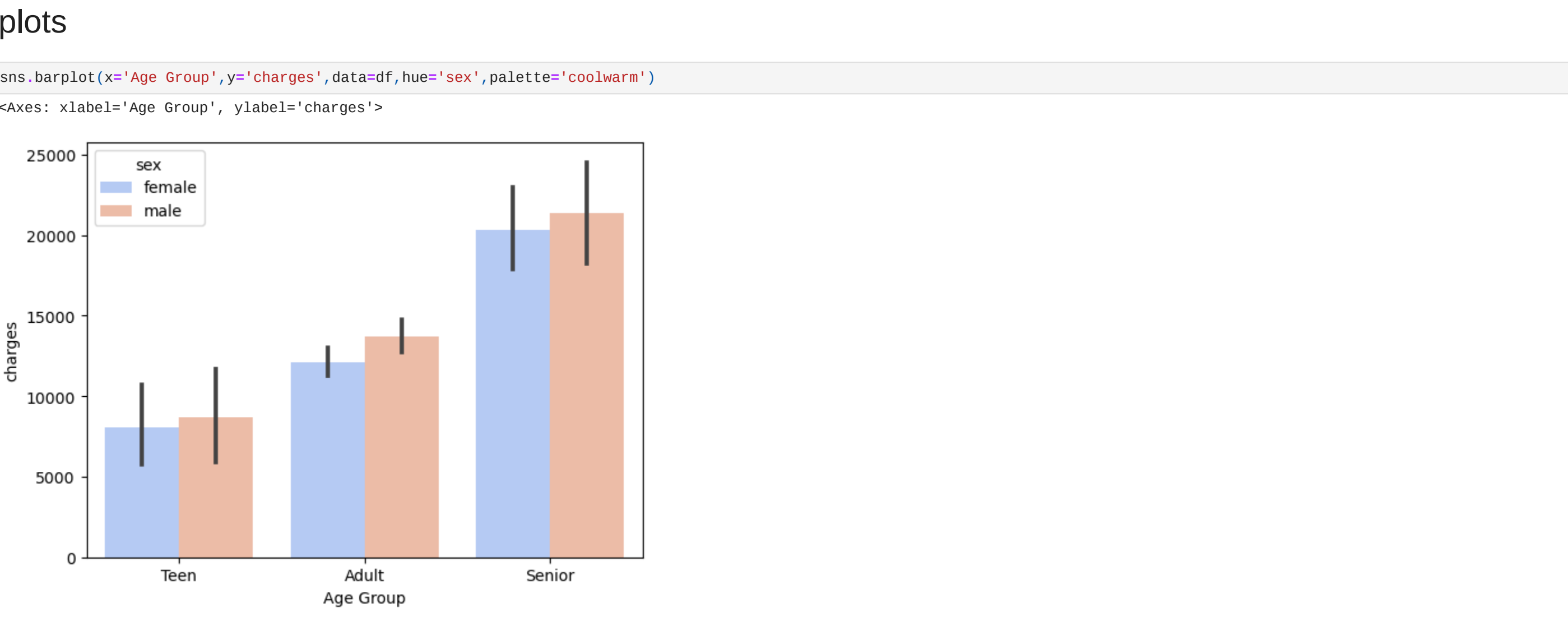
```
In [11]: df['Age Group'].value_counts()
```

```
Out[11]: Adult     1062
Senior      139
Teen        137
Name: Age Group, dtype: int64
```

plots

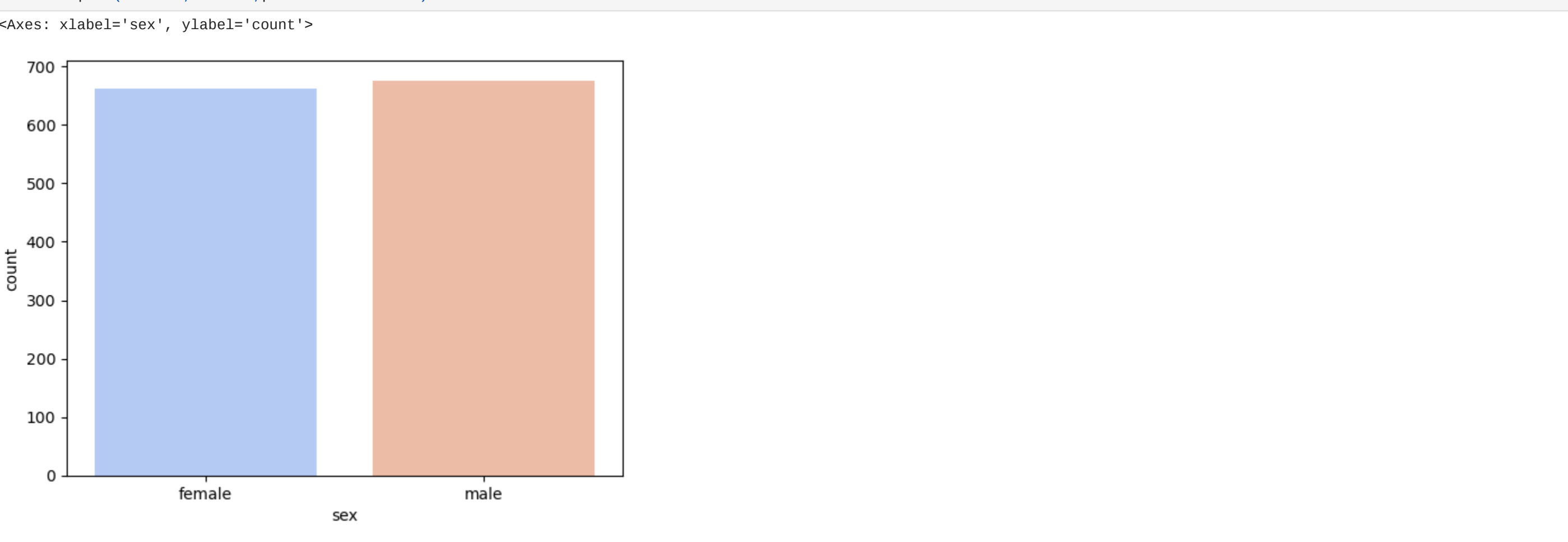
```
In [12]: sns.barplot(x='Age Group',y='charges',data=df,hue='sex',palette='coolwarm')
```

```
Out[12]: <Axes: xlabel='Age Group', ylabel='charges'>
```



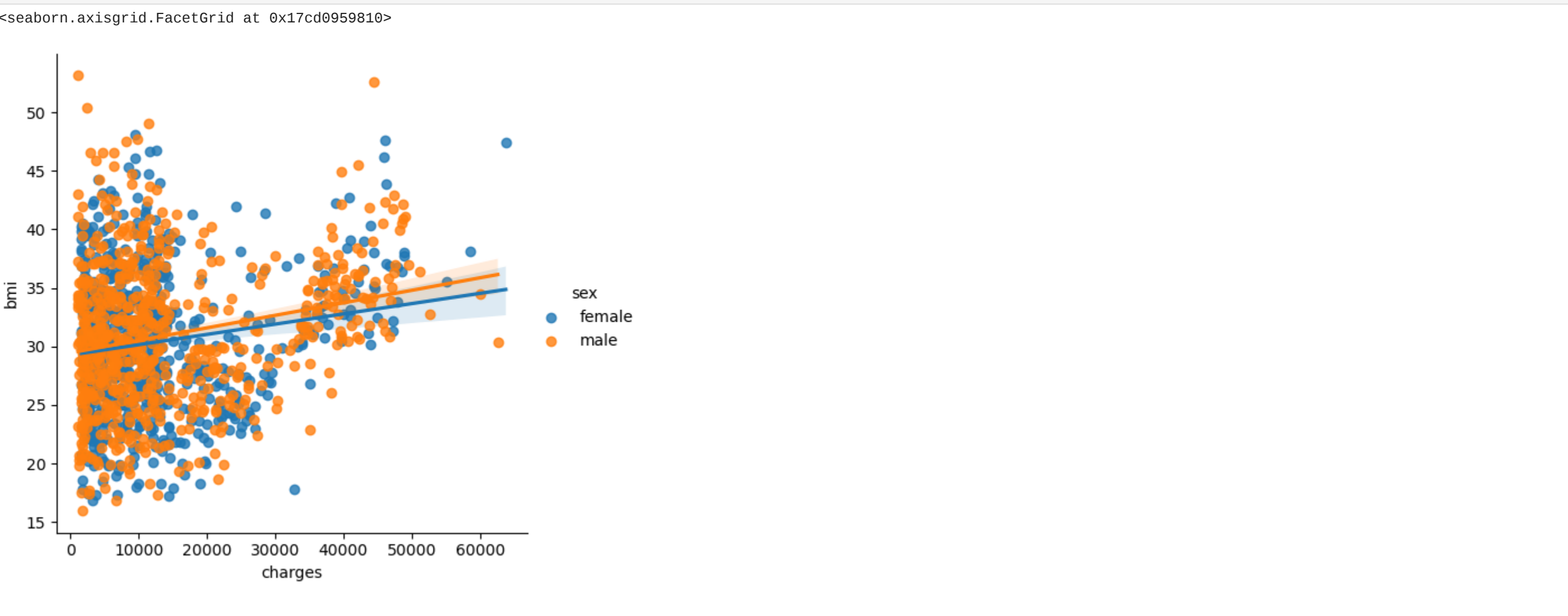
```
In [13]: sns.countplot(x='sex',data=df,palette='coolwarm')
```

```
Out[13]: <Axes: xlabel='sex', ylabel='count'>
```



```
In [14]: sns.lmplot(x='charges',y='bmi',data=df,hue='sex')
```

```
Out[14]: <seaborn.axisgrid.FacetGrid at 0x17cd0959810>
```



Converting BMI into a categorical group based on certain age group using a function

```
In [15]: def bmi_category(bmi):
if bmi < 18.5:
return 'Underweight'
elif 18.5 <= bmi < 24.9:
return 'Normal weight'
elif 24.9 <= bmi < 29.9:
return 'Overweight'
else:
return 'Obese'

df['bmi_category']=df['bmi'].apply(bmi_category )
```

We can convert BMI into a categorical group based on certain age group using pd.cut method also

```
In [16]: ## bins= [0,18.5,24.9,29.9,100]
## labels=['underweight','normal weight','overweight','obese']
```

```
In [17]: ## df['category']=pd.cut(df['bmi'], bins=bins, labels=labels, right=False)
```

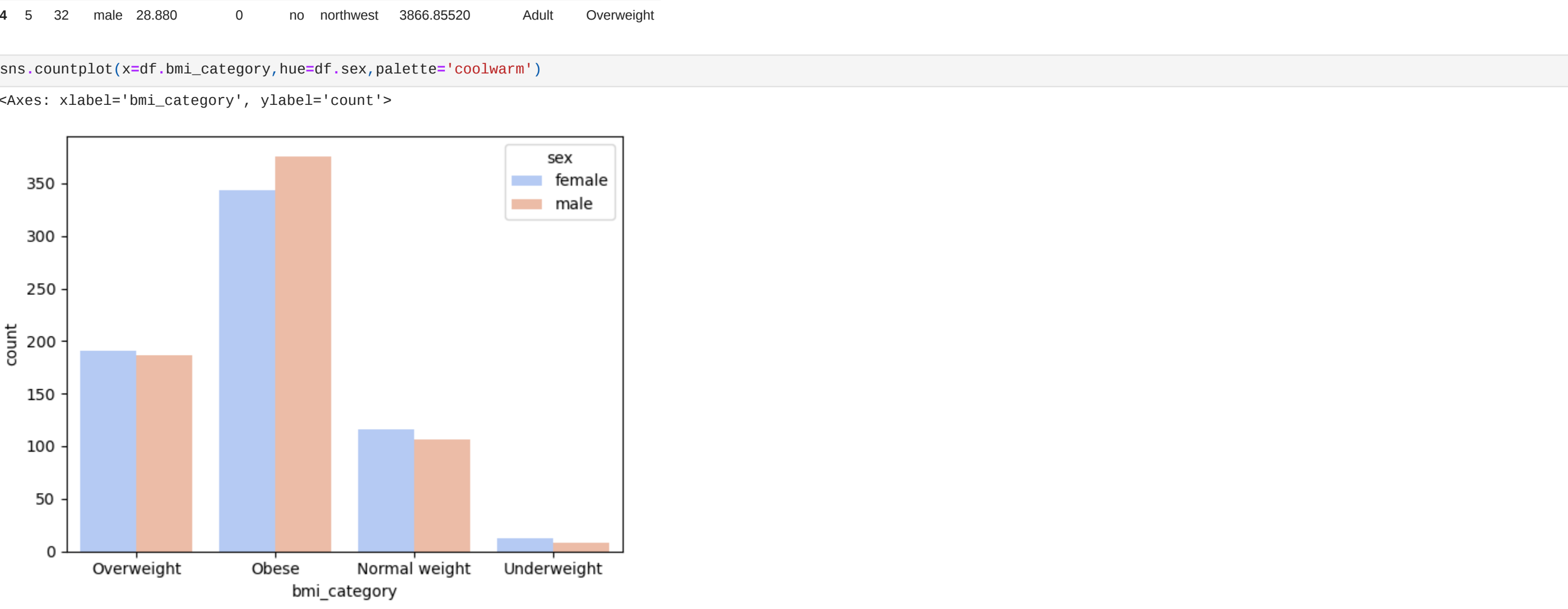
```
In [18]: ## df.drop('category',axis=1,inplace=True)
```

```
In [19]: df.head()
```

Out[19]:		Id	age	sex	bmi	children	smoker	region	charges	Age Group	bmi_category
	0	1	19	female	27.900	0	yes	southwest	16884.92400	Teen	Overweight
	1	2	18	male	33.770	1	no	southeast	1725.55230	Teen	Obese
	2	3	28	male	33.000	3	no	southeast	4449.46200	Adult	Obese
	3	4	33	male	22.705	0	no	northwest	21984.47061	Adult	Normal weight
	4	5	32	male	28.880	0	no	northwest	3866.85520	Adult	Overweight

```
In [20]: sns.countplot(x=df.bmi_category,hue=df.sex,palette='coolwarm')
```

```
Out[20]: <Axes: xlabel='bmi_category', ylabel='count'>
```

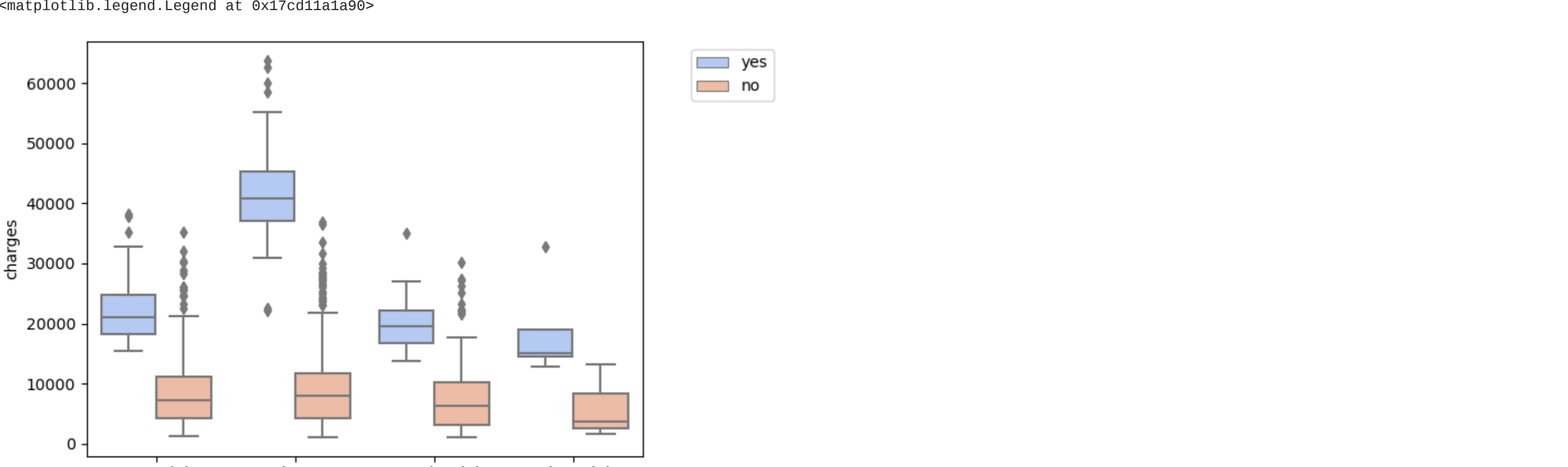


Finding the charges paid by different BMI\_Category based on smoking habits

```
In [21]: sns.boxplot(x='bmi_category',y='charges',data=df,hue='smoker',palette='coolwarm')
```

```
plt.legend(loc='upper right', bbox_to_anchor=(1.25, 1))
```

```
Out[21]: <matplotlib.legend.Legend at 0x17cd11a1a90>
```



```
In [22]: df.head()
```

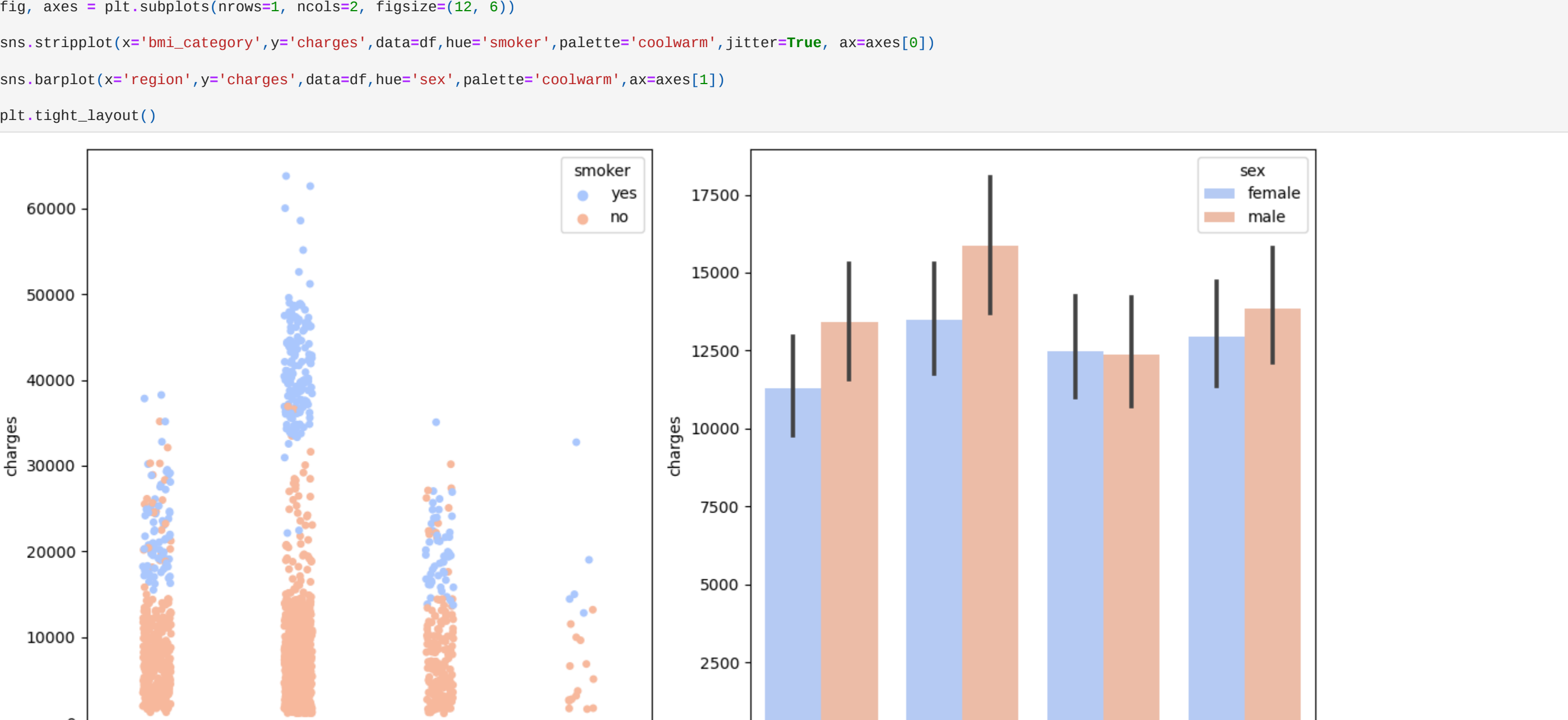
Out[22]:		Id	age	sex	bmi	children	smoker	region	charges	Age Group	bmi_category
	0	1	19	female	27.900	0	yes	southwest	16884.92400	Teen	Overweight
	1	2	18	male	33.770	1	no	southeast	1725.55230	Teen	Obese
	2	3	28	male	33.000	3	no	southeast	4449.46200	Adult	Obese
	3	4	33	male	22.705	0	no	northwest	21984.47061	Adult	Normal weight
	4	5	32	male	28.880	0	no	northwest	3866.85520	Adult	Overweight

```
In [23]: fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(12, 6))
```

```
sns.stripplot(x='bmi_category',y='charges',data=df,hue='smoker',palette='coolwarm',jitter=True, ax=axes[0])
```

```
sns.barplot(x='region',y='charges',data=df,hue='sex',palette='coolwarm',ax=axes[1])
```

```
plt.tight_layout()
```



```
In [24]: corr=df.corr()
```

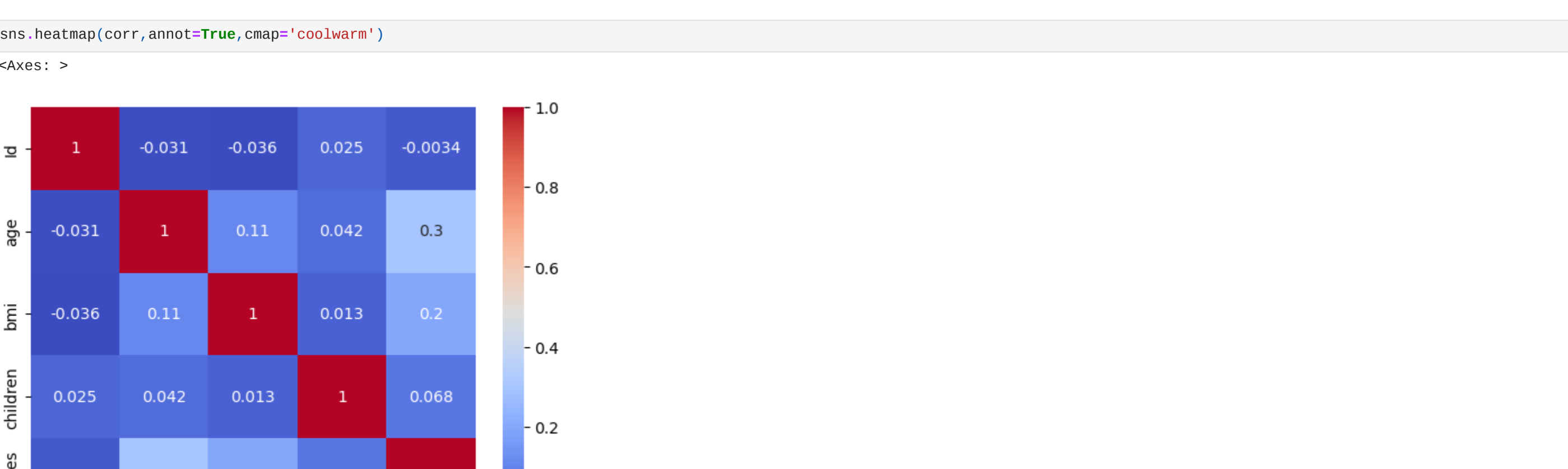
```
corr
```

```
Out[24]:
```

	Id	age	bmi	children	charges
Id	1.000000	-0.031467	-0.036169	0.025221	-0.003373
age	-0.031467	1.000000	0.109272	0.042469	0.299008
bmi	-0.036169	0.109272	1.000000	0.012759	0.198341
children	0.025221	0.042469	0.012759	1.000000	0.067998
charges	-0.003373	0.299008	0.198341	0.067998	1.000000

```
In [25]: sns.heatmap(corr,annot=True,cmap='coolwarm')
```

```
Out[25]: <Axes: >
```



```
In [ ]:
```