In [60]:

```
import numpy as np
import pandas as pd
import seaborn as sns
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error
```

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.shape
```

Out[4]:

```
(1338, 7)
```

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.describe()
```

Out[6]:

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

In [7]:

```
data.isna().sum()
```

Out[7]:

```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

In [8]:

```
data.head()
```

Out[8]:

	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 [9]:

```
Sex = pd.get_dummies(data["sex"] , drop_first = True)
data["gender"] = Sex.astype(int)

data = pd.concat([data ,Sex ] ,axis = 1)
```

In [10]:

```
data.drop(["sex" , "male"] , axis = 1 , inplace = True)
```

In [11]:

```
data["region"].unique()
```

Out[11]:

```
array(['southwest', 'southeast', 'northwest', 'northeast'], dtype=object)
```

In [12]:

```
data.head()
```

Out[12]:

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

In [13]:

```
smoker = pd.get_dummies(data["smoker"] , drop_first = True)
```

In [14]:

```
smoker = smoker.astype(int)
data = pd.concat([data , smoker] , axis = 1)
```

In [15]:

```
data["Smoker"] = data["yes"]
```

In [16]:

```
data = data.drop(["yes" , "smoker"] , axis = 1)
```

In [17]:

```
data
```

Out[17]:

	age	bmi	children	region	charges	gender	Smoker
0	19	27.900	0	southwest	16884.92400	0	1
1	18	33.770	1	southeast	1725.55230	1	0
2	28	33.000	3	southeast	4449.46200	1	0
3	33	22.705	0	northwest	21984.47061	1	0
4	32	28.880	0	northwest	3866.85520	1	0
...	...	...	...	...	...	...	...
1333	50	30.970	3	northwest	10600.54830	1	0
1334	18	31.920	0	northeast	2205.98080	0	0
1335	18	36.850	0	southeast	1629.83350	0	0
1336	21	25.800	0	southwest	2007.94500	0	0
1337	61	29.070	0	northwest	29141.36030	0	1

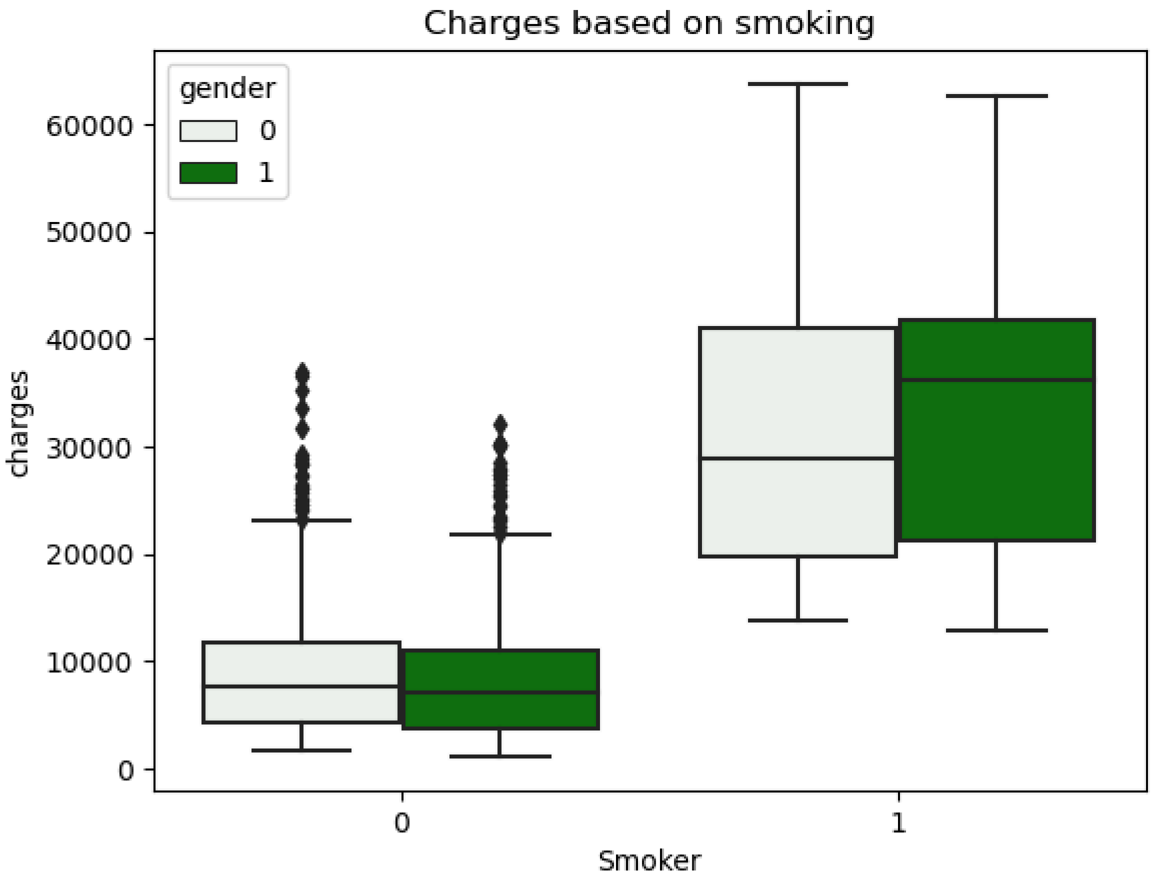
1338 rows × 7 columns

In [18]:

```
sns.boxplot(data = data ,x = "Smoker" , y = "charges" , hue = "gender" , color = "g
plt.title("Charges based on smoking")
```

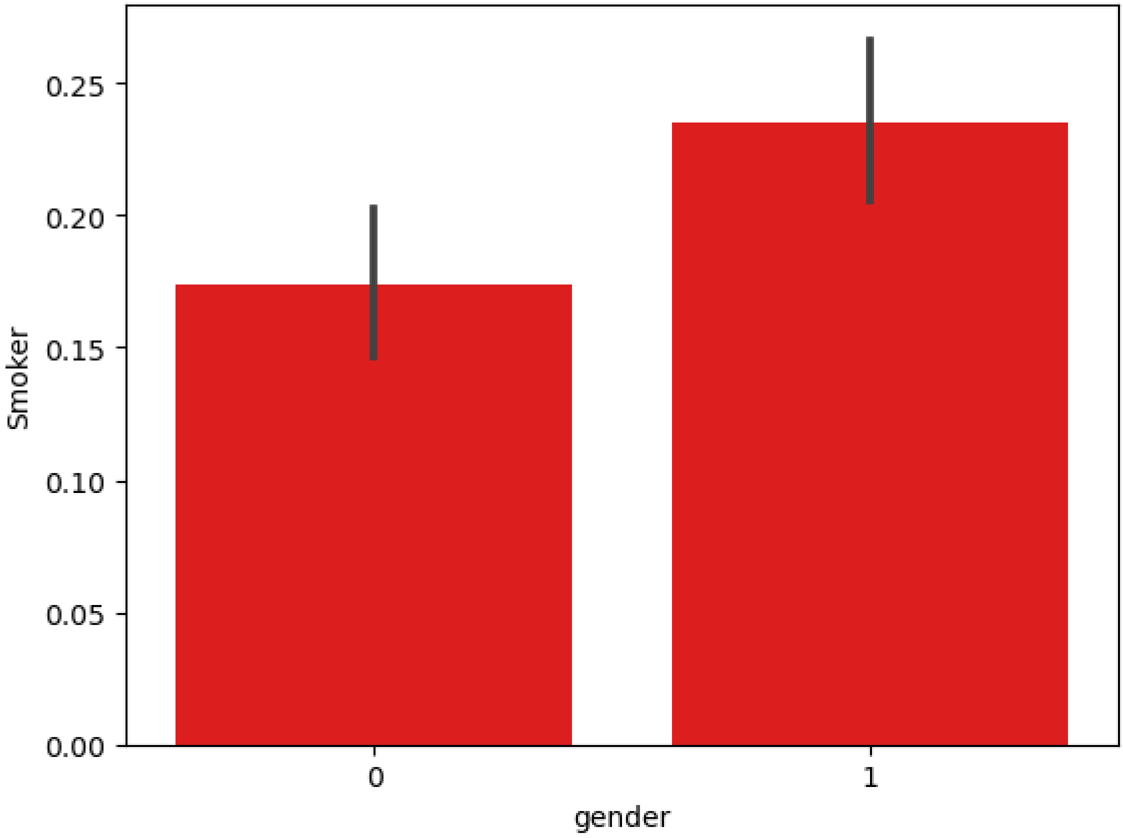
Out[18]:

```
Text(0.5, 1.0, 'Charges based on smoking')
```



In [19]:
```
sns.barplot(data = data , x = "gender" , y = "Smoker" , color = "red")
```

Out[19]:
```
<Axes: xlabel='gender', ylabel='Smoker'>
```



In [20]:
```
age_bins = data.value_counts()
```

In [21]:
```
data
```

Out[21]:

	age	bmi	children	region	charges	gender	Smoker
0	19	27.900	0	southwest	16884.92400	0	1
1	18	33.770	1	southeast	1725.55230	1	0
2	28	33.000	3	southeast	4449.46200	1	0
3	33	22.705	0	northwest	21984.47061	1	0
4	32	28.880	0	northwest	3866.85520	1	0
...	...	...	...	...	...	...	...
1333	50	30.970	3	northwest	10600.54830	1	0
1334	18	31.920	0	northeast	2205.98080	0	0
1335	18	36.850	0	southeast	1629.83350	0	0
1336	21	25.800	0	southwest	2007.94500	0	0
1337	61	29.070	0	northwest	29141.36030	0	1

1338 rows × 7 columns

In [22]:
```
data = data.drop("region" , axis = 1)
```

In [23]:
```
data
```

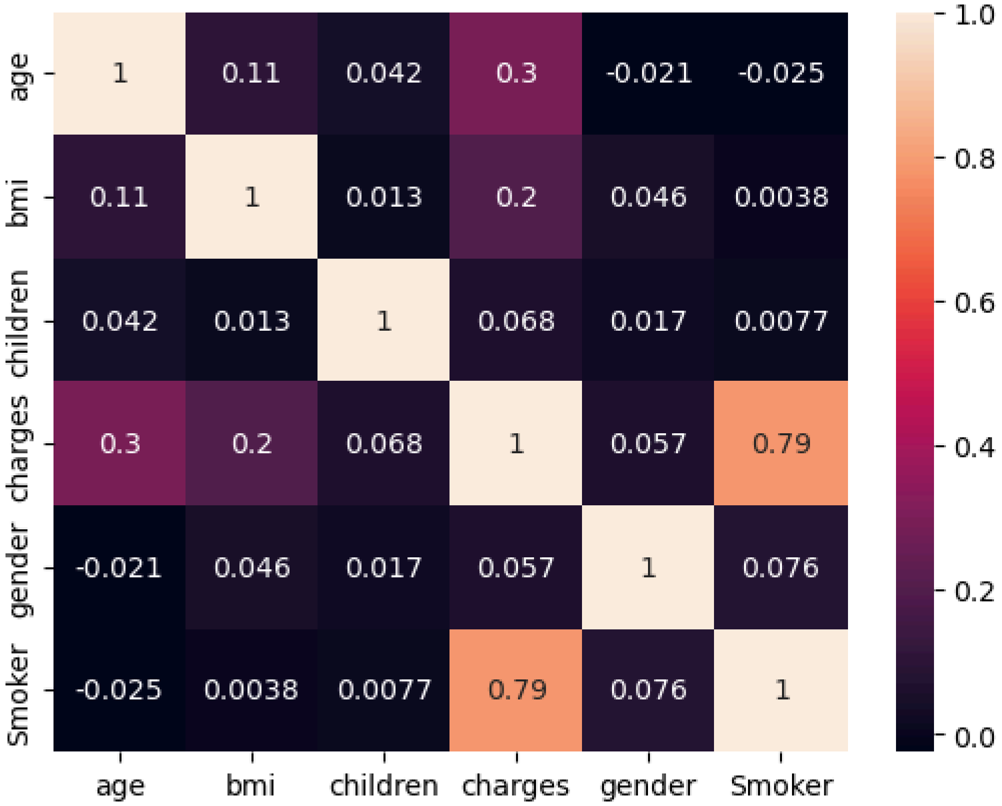
Out[23]:

	age	bmi	children	charges	gender	Smoker
0	19	27.900	0	16884.92400	0	1
1	18	33.770	1	1725.55230	1	0
2	28	33.000	3	4449.46200	1	0
3	33	22.705	0	21984.47061	1	0
4	32	28.880	0	3866.85520	1	0
...	...	...	...	...	...	...
1333	50	30.970	3	10600.54830	1	0
1334	18	31.920	0	2205.98080	0	0
1335	18	36.850	0	1629.83350	0	0
1336	21	25.800	0	2007.94500	0	0
1337	61	29.070	0	29141.36030	0	1

1338 rows × 6 columns

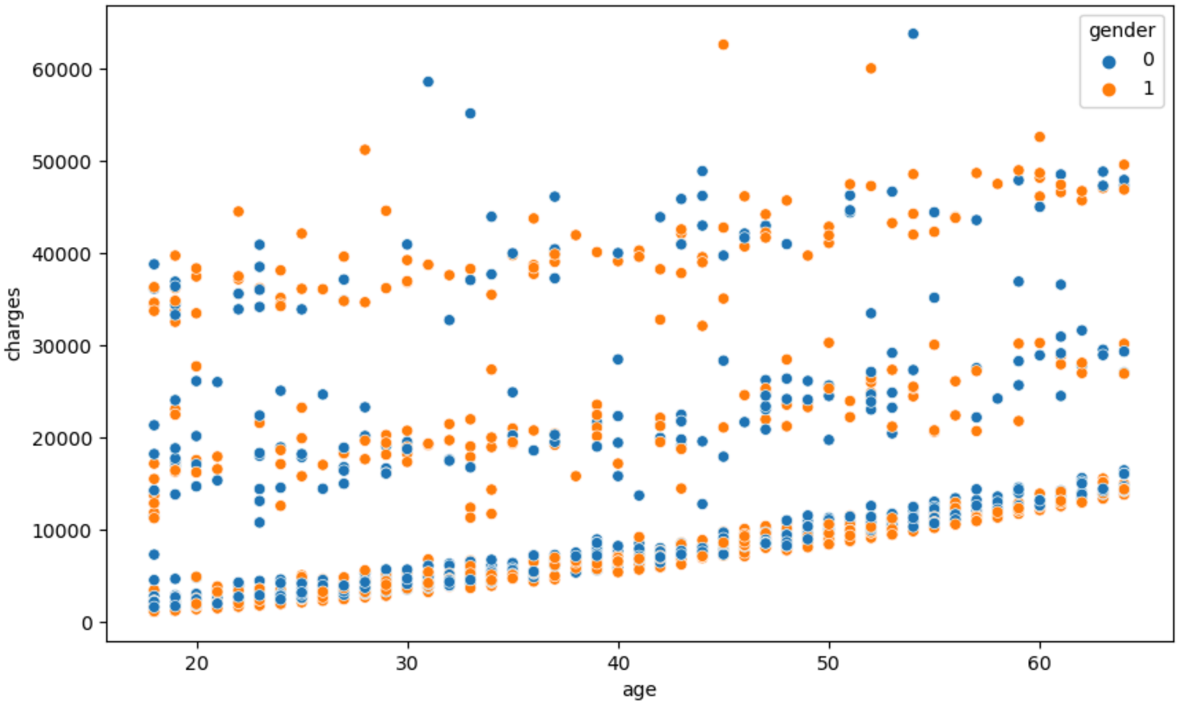
In [26]:
```
sns.heatmap(data.corr() ,annot = True , color = "green")
```

Out[26]:
```
<Axes: >
```



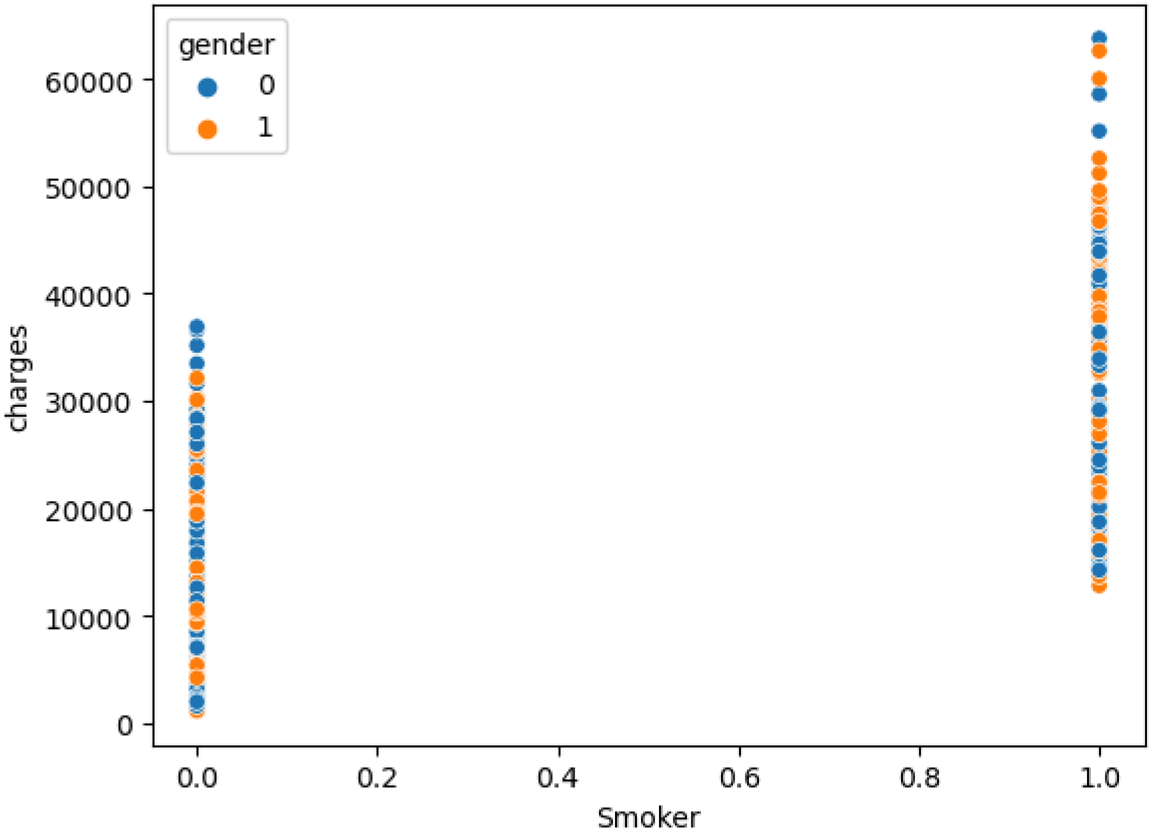
```
In [33]: plt.figure(figsize = (10,6))
         sns.scatterplot(data = data , x = "age" , y = "charges" , hue = "gender")
```

```
Out[33]: <Axes: xlabel='age', ylabel='charges'>
```



```
In [35]: sns.scatterplot(data = data , x = "Smoker" , y = "charges" , hue = "gender")
```

```
Out[35]: <Axes: xlabel='Smoker', ylabel='charges'>
```



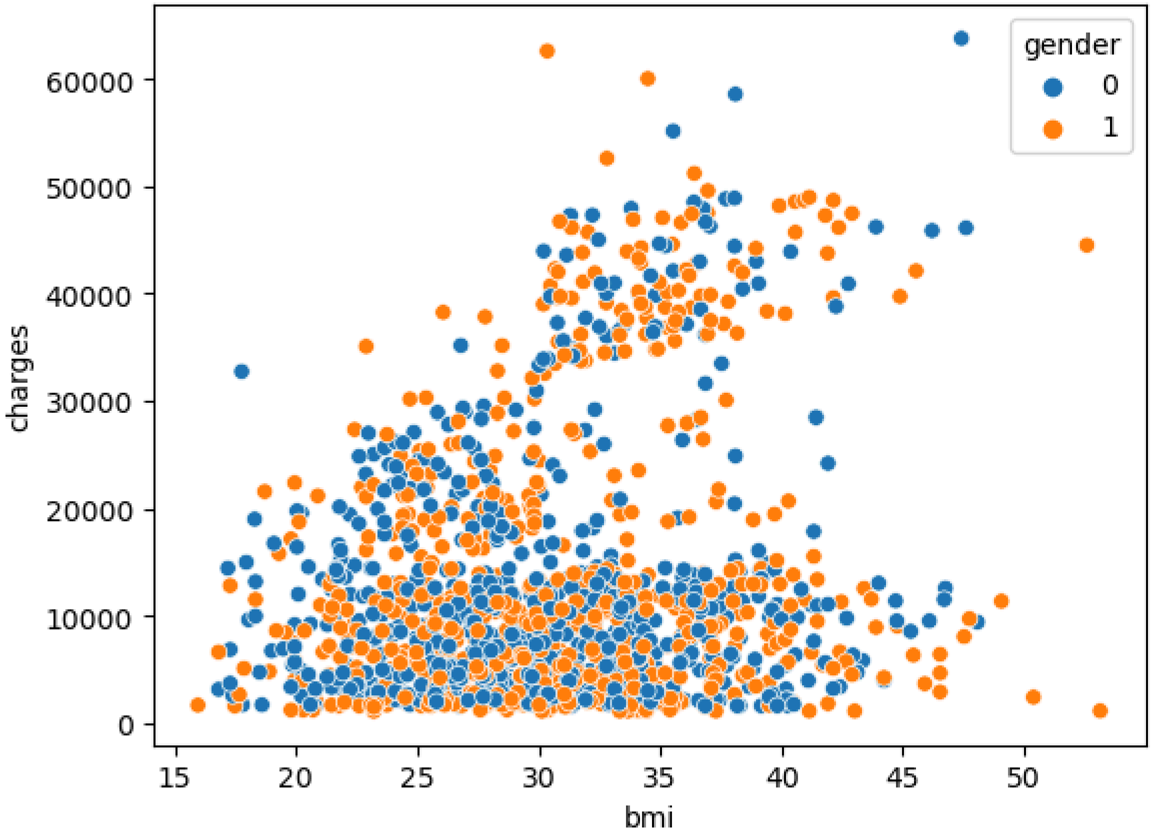
```
In [37]: data.head()
```

Out[37]:

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

```
In [41]: sns.scatterplot(data = data , x = "bmi" , y = "charges" , hue = "gender" )
```

```
Out[41]: <Axes: xlabel='bmi', ylabel='charges'>
```



In [43]:

```
x = data.drop("charges" , axis = 1)
```

In [46]:

```
y = data[["charges"]]
```

In [47]:

```
x
```

Out[47]:

	age	bmi	children	gender	Smoker
0	19	27.900	0	0	1
1	18	33.770	1	1	0
2	28	33.000	3	1	0
3	33	22.705	0	1	0
4	32	28.880	0	1	0
...	...	...	...	...	...
1333	50	30.970	3	1	0
1334	18	31.920	0	0	0
1335	18	36.850	0	0	0
1336	21	25.800	0	0	0
1337	61	29.070	0	0	1

1338 rows × 5 columns

In [48]:

```
y
```

Out[48]:

	charges
0	16884.92400
1	1725.55230
2	4449.46200
3	21984.47061
4	3866.85520
...	...
1333	10600.54830
1334	2205.98080
1335	1629.83350
1336	2007.94500
1337	29141.36030

1338 rows × 1 columns

In [49]:
```
x_train , x_test , y_train , y_test = train_test_split(x , y , test_size = 0.2 , ra
```

In [50]:
```
x_train
```

Out[50]:

	age	bmi	children	gender	Smoker
1256	51	36.385	3	0	0
147	51	37.730	1	0	0
1042	20	30.685	0	1	1
889	57	33.630	1	1	0
650	49	42.680	2	0	0
...	...	...	...	...	...
1223	20	24.420	0	0	1
667	40	32.775	2	0	1
156	48	24.420	0	1	1
384	44	22.135	2	1	0
645	48	30.780	3	1	0

1070 rows × 5 columns

In [51]:
```
x_test
```

Out[51]:

	age	bmi	children	gender	Smoker
38	35	36.67	1	1	1
126	19	28.30	0	0	1
479	23	32.56	0	1	0
10	25	26.22	0	1	0
195	19	30.59	0	1	0
...	...	...	...	...	...
1059	32	33.82	1	1	0
303	28	33.00	2	0	0
335	64	34.50	0	1	0
792	22	23.18	0	0	0
1213	52	33.30	2	0	0

268 rows × 5 columns

In [52]:

```
y_train
```

Out[52]:

	charges
1256	11436.73815
147	9877.60770
1042	33475.81715
889	11945.13270
650	9800.88820
...	...
1223	26125.67477
667	40003.33225
156	21223.67580
384	8302.53565
645	10141.13620

1070 rows × 1 columns

In [53]:

```
y_test
```

Out[53]:

	charges
38	39774.2763
126	17081.0800
479	1824.2854
10	2721.3208
195	1639.5631
...	...
1059	4462.7218
303	4349.4620
335	13822.8030
792	2731.9122
1213	10806.8390

268 rows × 1 columns

In [55]:

```
from sklearn.preprocessing import StandardScaler
scalar = StandardScaler()
```

In [56]:

```
scalar
```

Out[56]:

```
▾ StandardScaler
StandardScaler()
```

In [57]:

```
x_train_scalar = scalar.fit_transform(x_train)
x_test_scalar = scalar.transform(x_test)
```

In [58]:

```
x_train_scalar
```

Out[58]:

```
array([[ 0.8143715 ,  0.92361714,  1.59576356, -1.00938988, -0.51165658],
       [ 0.8143715 ,  1.14285117, -0.06519657, -1.00938988, -0.51165658],
       [-1.38518087, -0.00547876, -0.89567663,  0.99069747,  1.95443593],
       ...,
       [ 0.60151159, -1.02666924, -0.89567663,  0.99069747,  1.95443593],
       [ 0.31769838, -1.3991226 ,  0.7652835 ,  0.99069747, -0.51165658],
       [ 0.60151159,  0.01000618,  1.59576356,  0.99069747, -0.51165658]])
```

In [59]:

```
x_test_scalar
```

Out[59]:

```
array([[-0.32088134,  0.97007193, -0.06519657,  0.99069747,  1.95443593],
       [-1.45613417, -0.39423204, -0.89567663, -1.00938988,  1.95443593],
       [-1.17232096,  0.30014489, -0.89567663,  0.99069747, -0.51165658],
       ...,
       [ 1.73676442,  0.6163635 , -0.89567663,  0.99069747, -0.51165658],
       [-1.24327426, -1.22878835, -0.89567663, -1.00938988, -0.51165658],
       [ 0.8853248 ,  0.42076436,  0.7652835 , -1.00938988, -0.51165658]])
```

In [61]:

```
model = SVR()
```

In [62]:

```
model
```

Out[62]:

```
▾ SVR
SVR()
```

In [63]:

```
model.fit(x_train_scalar , y_train)
```

```
C:\Users\godde\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataC
onversionWarning: A column-vector y was passed when a 1d array was expected. Pleas
e change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Out[63]:

```
▾ SVR
SVR()
```

In [64]:

```
prediction = model.predict(x_test_scalar)
```

In [67]:

```
prediction[:10]
```

Out[67]:

```
array([9781.83376118, 9741.84235083, 9597.43786842, 9595.62819898,
       9594.07229657, 9635.75545188, 9616.73779584, 9769.18794274,
       9635.94255934, 9635.11359076])
```

In [68]:

```
y_test[:10]
```

Out[68]:

	charges
38	39774.27630
126	17081.08000
479	1824.28540
10	2721.32080
195	1639.56310
43	6313.75900
1302	3208.78700
488	48885.13561
1198	6393.60345
8	6406.41070

In [69]:

```
error = mean_squared_error(prediction , y_test)
```

In [70]:

```
error
```

Out[70]:

```
135719201.65505797
```

In [ ]: