In [9]:

```
import numpy as np
import pandas as pd
```

In [10]:

```
df=pd.read_csv(r"C:\Users\91950\Downloads\Advertising.csv")
df
```

Out[10]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
...	...	...	...	...
195	38.2	3.7	13.8	7.6
196	94.2	4.9	8.1	14.0
197	177.0	9.3	6.4	14.8
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

200 rows × 4 columns

In [11]:

```
df.head()
```

Out[11]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

In [12]:

```
df.tail()
```

Out[12]:

	TV	Radio	Newspaper	Sales
195	38.2	3.7	13.8	7.6
196	94.2	4.9	8.1	14.0
197	177.0	9.3	6.4	14.8
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

In [13]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   TV         200 non-null    float64
 1   Radio      200 non-null    float64
 2   Newspaper  200 non-null    float64
 3   Sales      200 non-null    float64
dtypes: float64(4)
memory usage: 6.4 KB
```

In [14]:

```
df.columns
```

Out[14]:

```
Index(['TV', 'Radio', 'Newspaper', 'Sales'], dtype='object')
```

In [15]:

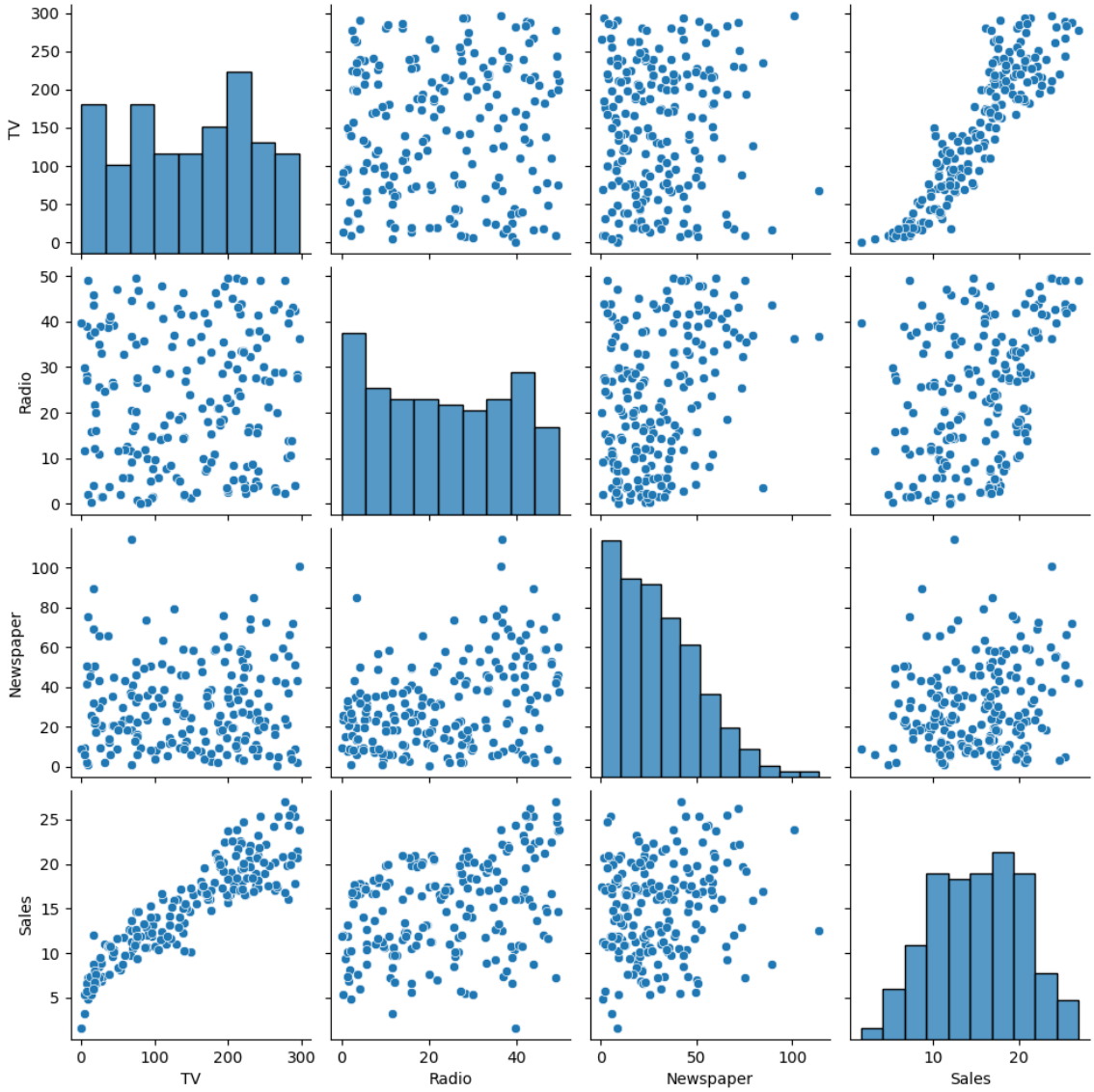
```
import seaborn as sns
import matplotlib.pyplot as plt
```

In [16]:

```
sns.pairplot(df)
```

Out[16]:

```
<seaborn.axisgrid.PairGrid at 0x25ef8d55c90>
```

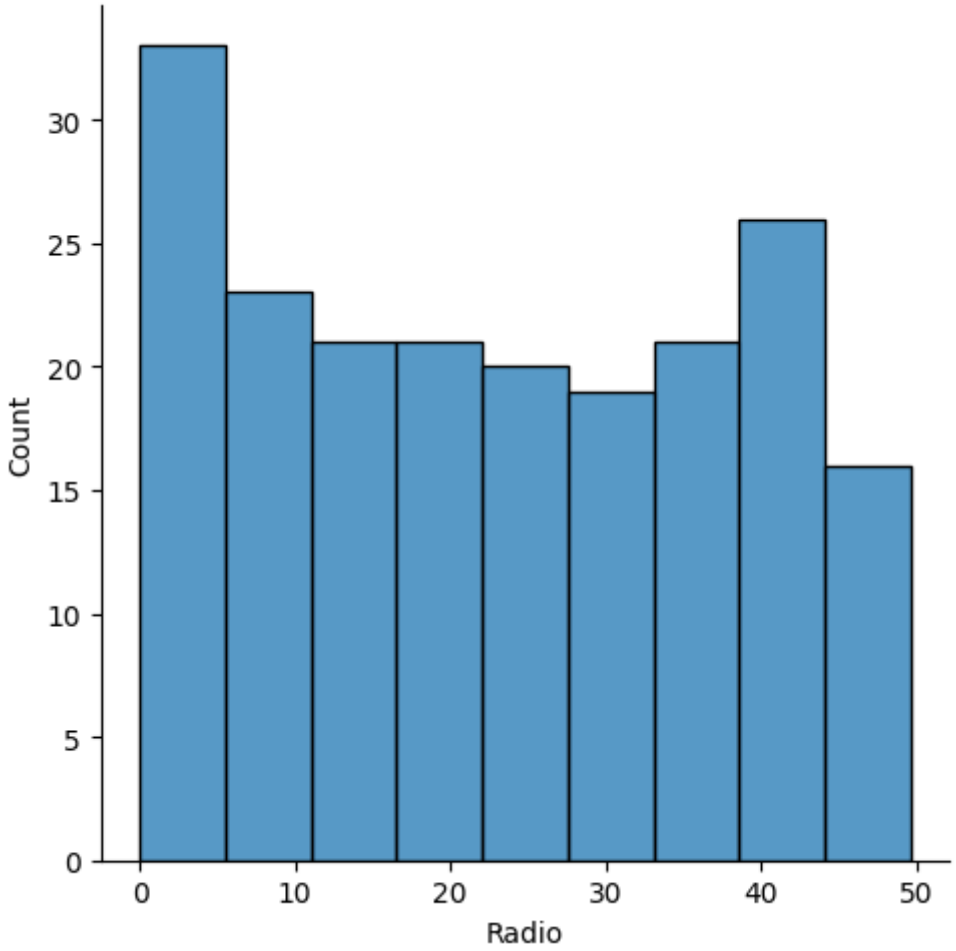


In [17]:

```
sns.displot(df['Radio'])
```

Out[17]:

```
<seaborn.axisgrid.FacetGrid at 0x25e9c79bf10>
```

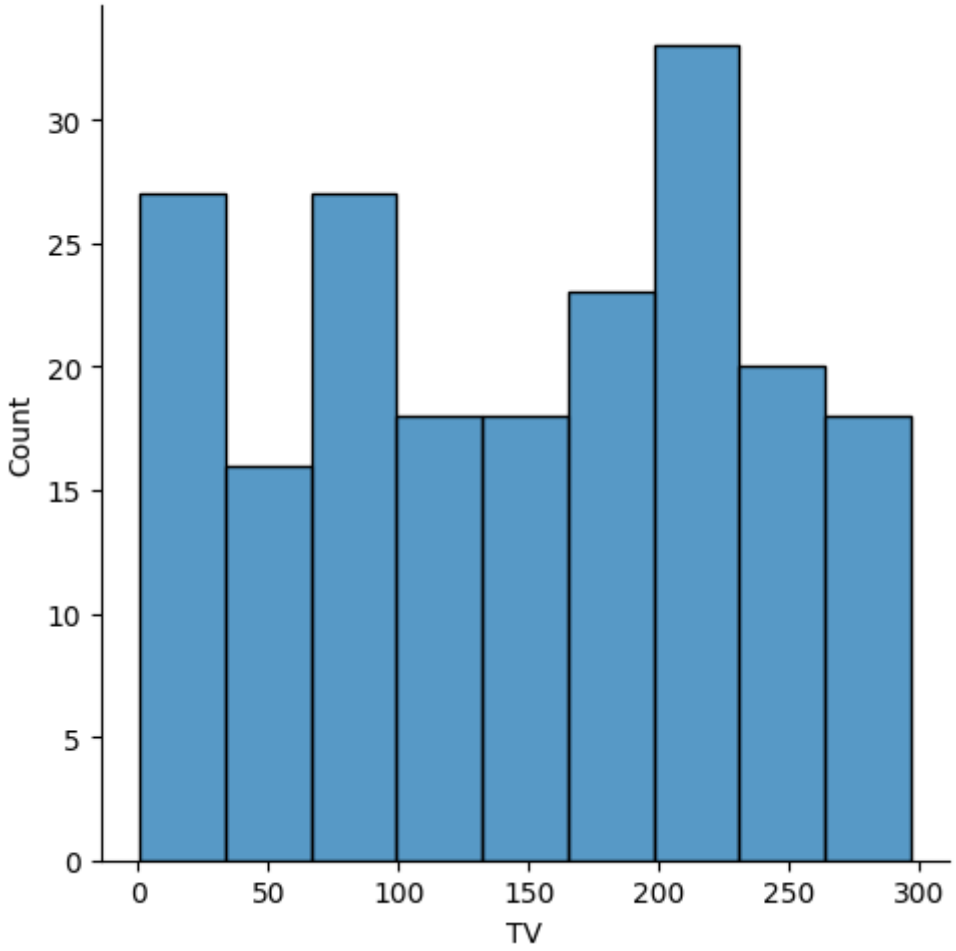


In [33]:

```
sns.displot(df['TV'])
```

Out[33]:

```
<seaborn.axisgrid.FacetGrid at 0x1cf16918f10>
```

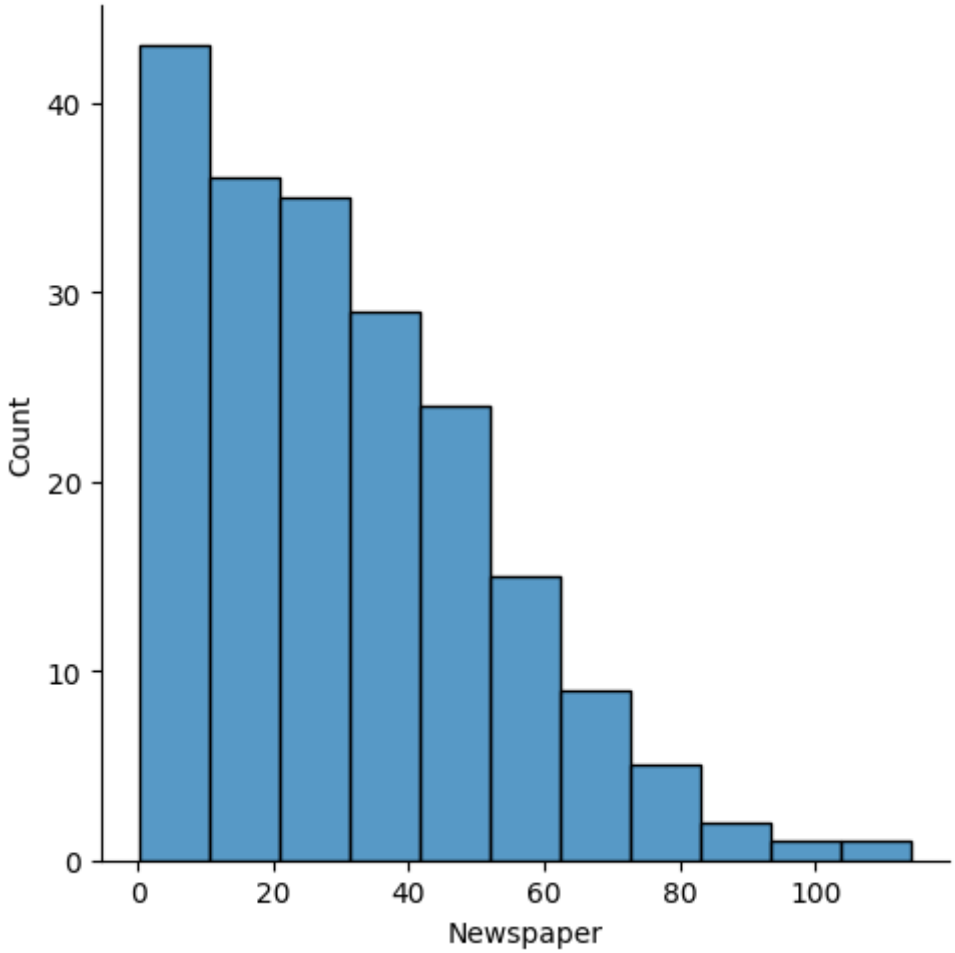


In [18]:

```
sns.displot(df['Newspaper'])
```

Out[18]:

```
<seaborn.axisgrid.FacetGrid at 0x25e9c745b10>
```

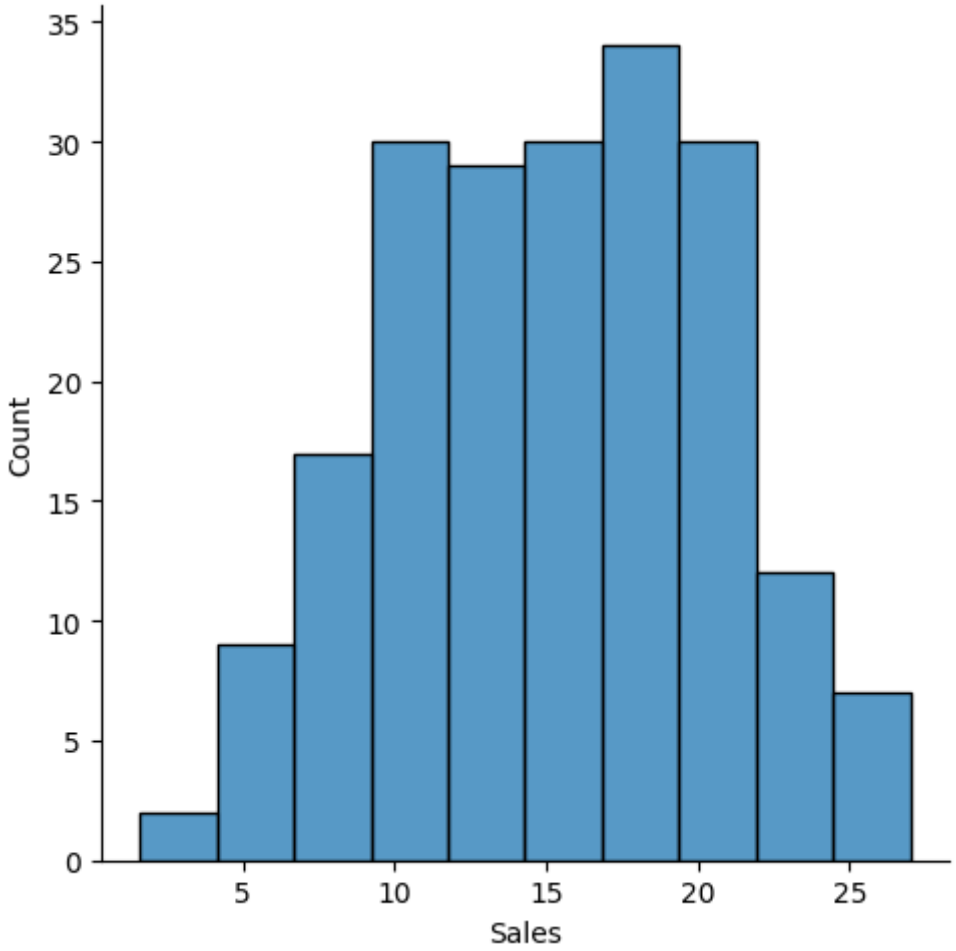


In [19]:

```
sns.displot(df['Sales'])
```

Out[19]:

```
<seaborn.axisgrid.FacetGrid at 0x25e9cc92850>
```



In [20]:

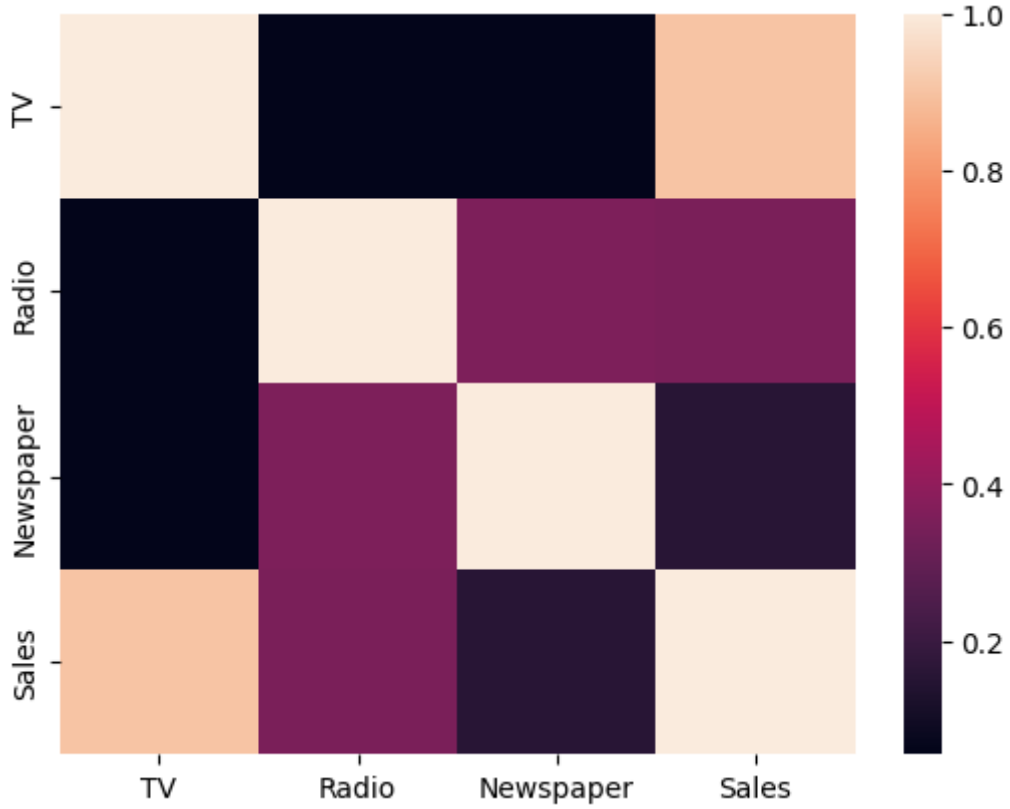
```
Advertisingdf=df[['TV', 'Radio', 'Newspaper','Sales']]
```

In [21]:

```
sns.heatmap(Advertisingdf.corr())
```

Out[21]:

<Axes: >



In [22]:

```
x=Advertisingdf[['TV', 'Radio', 'Newspaper', 'Sales']]
y=df['Radio']
```

In [23]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=101)
```

In [24]:

```
from sklearn.linear_model import LinearRegression
lm=LinearRegression()
lm.fit(x_train,y_train)
```

Out[24]:

```
LinearRegression()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [27]:

```
print(lm.intercept_)
```

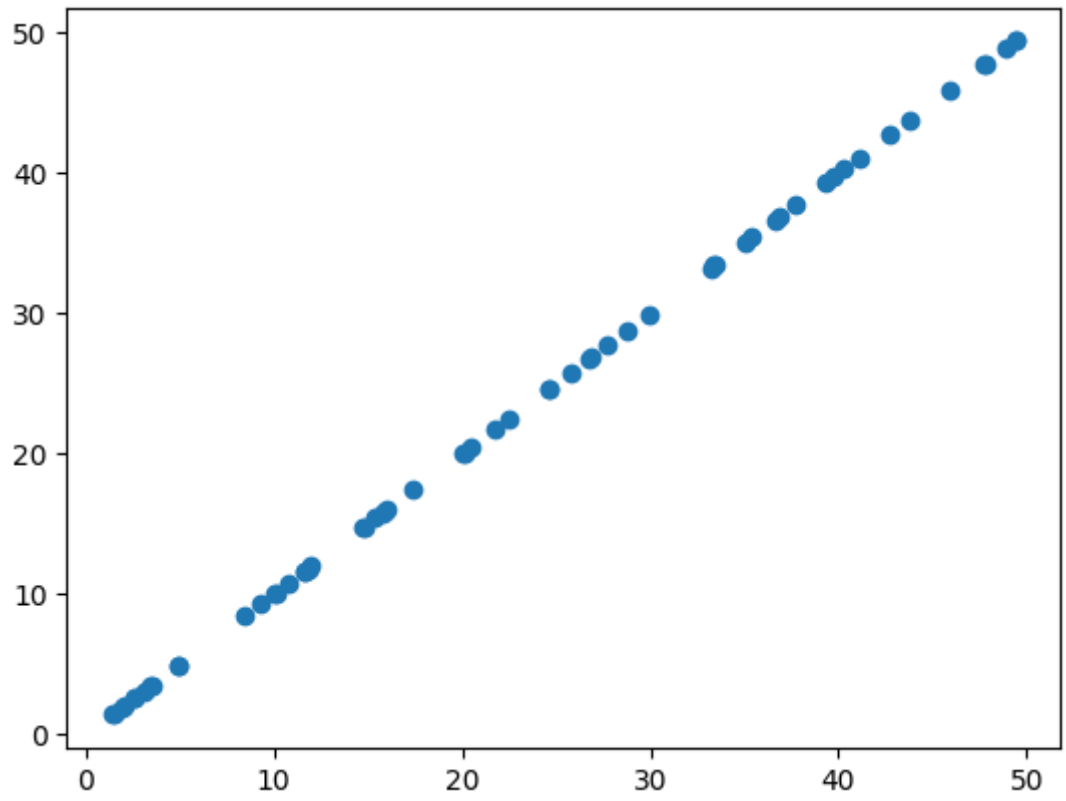
```
0.0
```

In [28]:

```
predictions=lm.predict(x_test)
plt.scatter(y_test,predictions)
```

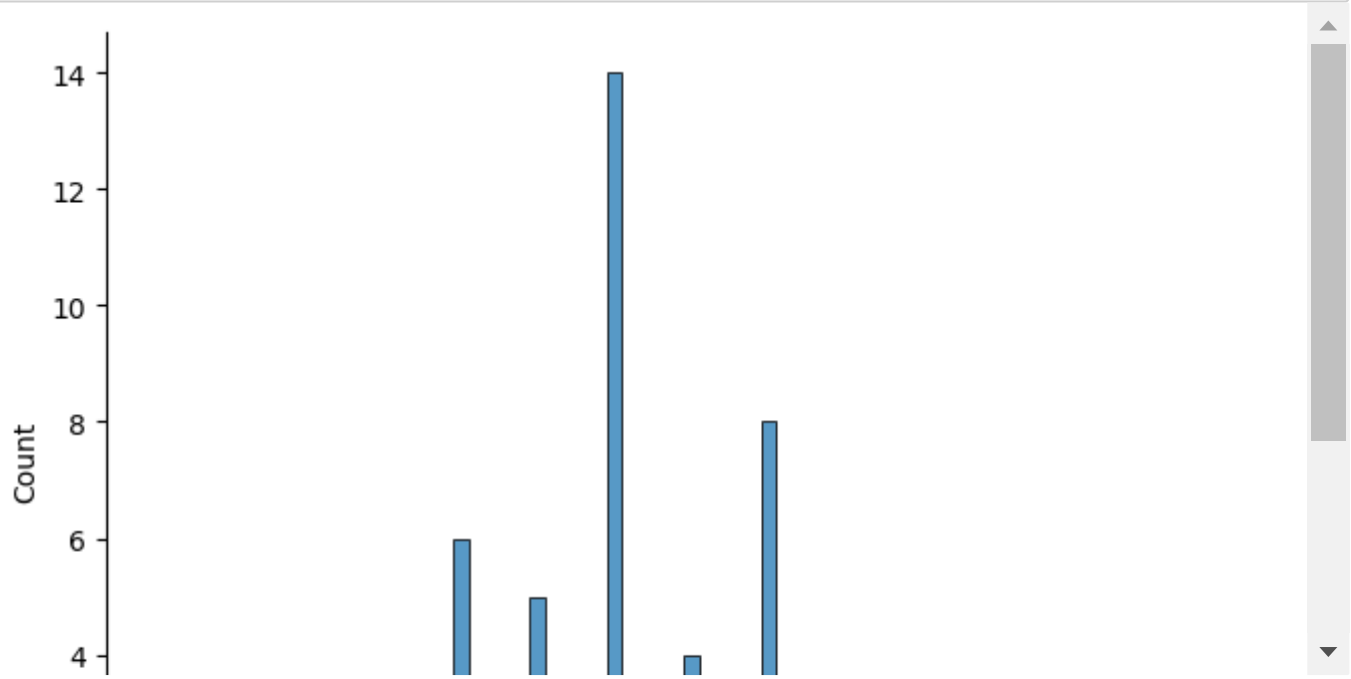
Out[28]:

```
<matplotlib.collections.PathCollection at 0x25e9e092510>
```



In [29]:

```
sns.displot((y_test-predictions),bins=50);
```



In [33]:

```
from sklearn import metrics
print('MAE:',metrics.mean_absolute_error(y_test,predictions))
print('MSE:',metrics.mean_squared_error(y_test,predictions))
print('RMSE',np.sqrt(metrics.mean_squared_error(y_test,predictions)))
```

```
MAE: 5.684341886080802e-15
MSE: 5.784815625598832e-29
RMSE 7.605797542400686e-15
```