In [17]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [18]:
```
df=pd.read_csv("4_drug200 (1).csv")
df
```

Out[18]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Age          200 non-null    int64  
 1   Sex          200 non-null    object 
 2   BP           200 non-null    object 
 3   Cholesterol  200 non-null    object 
 4   Na_to_K      200 non-null    float64
 5   Drug         200 non-null    object 
dtypes: float64(1), int64(1), object(4)
memory usage: 9.5+ KB
```

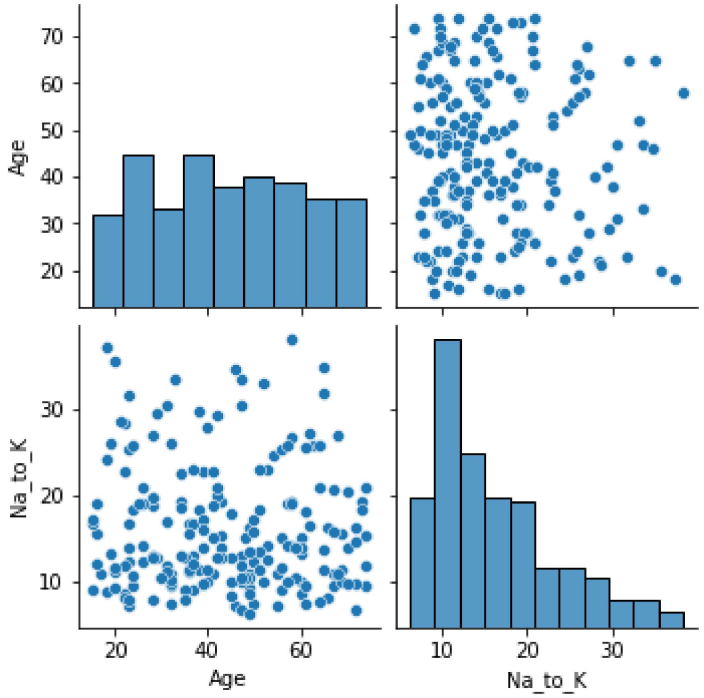
In [20]:
```
df.describe()
```

Out[20]:

	Age	Na_to_K
count	200.000000	200.000000
mean	44.315000	16.084485
std	16.544315	7.223956
min	15.000000	6.269000
25%	31.000000	10.445500
50%	45.000000	13.936500
75%	58.000000	19.380000
max	74.000000	38.247000

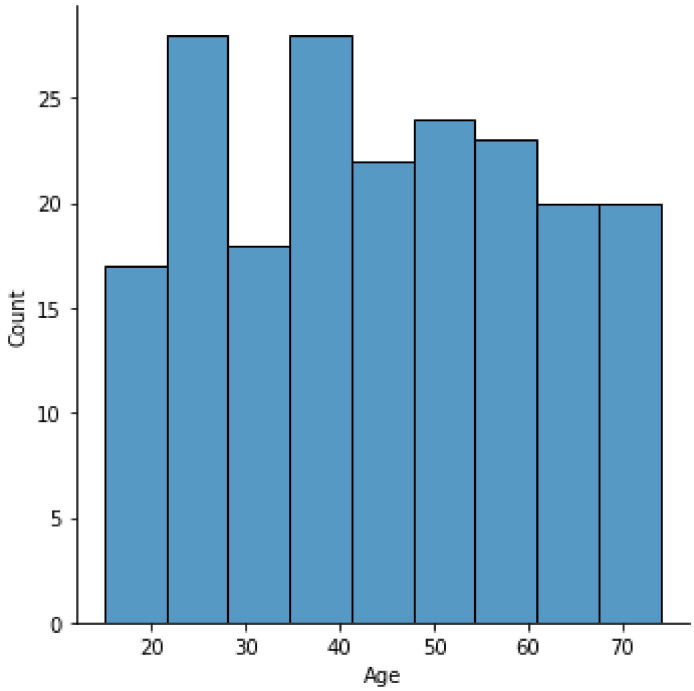
In [21]:
```
sns.pairplot(df)
```

Out[21]:
```
<seaborn.axisgrid.PairGrid at 0x2596e966a90>
```



In [22]:
```
sns.displot(df['Age'])
```

Out[22]:
```
<seaborn.axisgrid.FacetGrid at 0x259c1e1f9a0>
```



In [23]:
```
df1=df.drop(['Sex'],axis=1)
df1
```

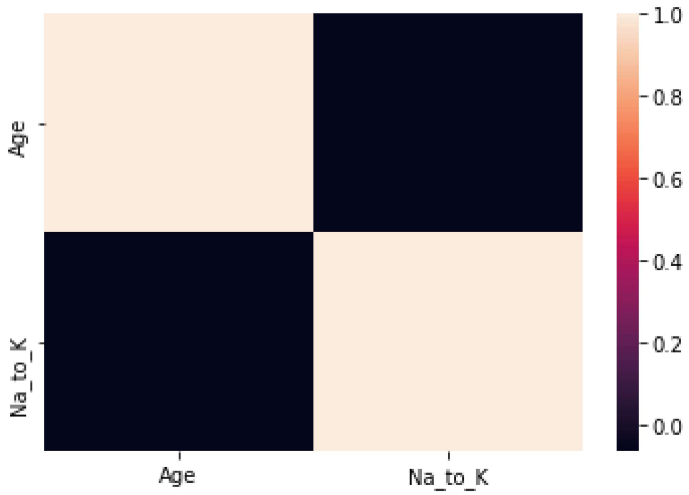
Out[23]:

	Age	BP	Cholesterol	Na_to_K	Drug
0	23	HIGH	HIGH	25.355	drugY
1	47	LOW	HIGH	13.093	drugC
2	47	LOW	HIGH	10.114	drugC
3	28	NORMAL	HIGH	7.798	drugX
4	61	LOW	HIGH	18.043	drugY
...	...	...	...	...	...
195	56	LOW	HIGH	11.567	drugC
196	16	LOW	HIGH	12.006	drugC
197	52	NORMAL	HIGH	9.894	drugX
198	23	NORMAL	NORMAL	14.020	drugX
199	40	LOW	NORMAL	11.349	drugX

200 rows × 5 columns

In [24]:
```
sns.heatmap(df1.corr())
```

Out[24]: <AxesSubplot:>



In [25]:
```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [26]:
```
y=df['Age']
x=df1.drop(['Age','BP','Cholesterol','Drug'],axis=1)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
print(x_train)
```

```
     Na_to_K
27     9.381
112    9.170
130   20.489
69    24.276
125   25.969
..       ...
117   10.103
172   17.225
185   25.893
169   11.262
63     8.370

[140 rows x 1 columns]
```

In [27]:
```
model=LinearRegression()
model.fit(x_train,y_train)
model.intercept_
```

Out[27]: 46.531395227472174

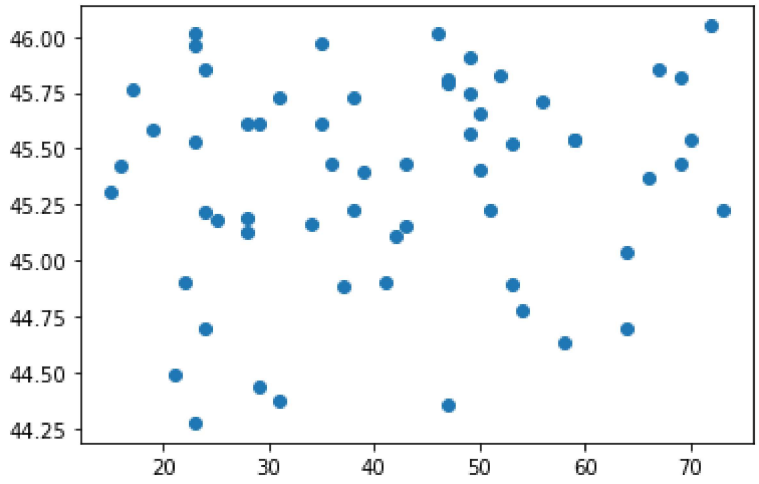
In [28]:
```
coeff=pd.DataFrame(model.coef_,x.columns,columns=["Coefficient"])
coeff
```

Out[28]:

	Coefficient
Na_to_K	-0.071192

In [29]:
```
prediction=model.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[29]:
```
<matplotlib.collections.PathCollection at 0x259c1fb2ca0>
```



In [30]:
```
model.score(x_test,y_test)
```

Out[30]:
```
-0.041694123836048425
```

In [32]:
```
from sklearn.linear_model import Ridge,Lasso
```

In [33]:
```
rr = Ridge(alpha=10)
rr.fit(x_train,y_train)
```

Out[33]:
```
Ridge(alpha=10)
```

In [34]:
```
rr.score(x_test,y_test)
```

Out[34]:
```
-0.04170334355486549
```

In [35]:
```
la = Lasso(alpha=10)
la.fit(x_train,y_train)
```

Out[35]:
```
Lasso(alpha=10)
```

In [36]:
```
la.score(x_test,y_test)
```

Out[36]:
```
-0.04975954502177338
```

In [ ]: