```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv('/content/energy_dataset - energy_dataset.csv')

df.head()
```

	time	generation biomass	generation fossil brown coal/lignite	generation fossil coal-derived gas	generation fossil gas	generation fossil hard coal	gener fossi
0	2015-01-01 00:00:00+01:00	447.0	329.0	0.0	4844.0	4821.0	
1	2015-01-01 01:00:00+01:00	449.0	328.0	0.0	5196.0	4755.0	
2	2015-01-01 02:00:00+01:00	448.0	323.0	0.0	4857.0	4581.0	
3	2015-01-01 03:00:00+01:00	438.0	254.0	0.0	4314.0	4131.0	
4	2015-01-01 04:00:00+01:00	428.0	187.0	0.0	4130.0	3840.0	

5 rows × 29 columns

```
df.tail()
```

	time	generation biomass	generation fossil brown coal/lignite	generation fossil coal-derived gas	generation fossil gas	generation fossil hard coal	g f
16564	2016-11-21 04:00:00+01:00	338.0	609.0	0.0	3439.0	1450.0	
16565	2016-11-21 05:00:00+01:00	343.0	613.0	0.0	3451.0	1748.0	

```
df.shape
```

```
(16569, 29)
```

16567	2016-11-21	360.0	886.0	0.0	4268.0	3342.0	

```
df.columns
```

```
Index(['time', 'generation biomass', 'generation fossil brown coal/lignite',
       'generation fossil coal-derived gas', 'generation fossil gas',
       'generation fossil hard coal', 'generation fossil oil',
       'generation fossil oil shale', 'generation fossil peat',
       'generation geothermal', 'generation hydro pumped storage aggregated',
       'generation hydro pumped storage consumption',
       'generation hydro run-of-river and poundage',
       'generation hydro water reservoir', 'generation marine',
       'generation nuclear', 'generation other', 'generation other renewable',
       'generation solar', 'generation waste', 'generation wind offshore',
       'generation wind onshore', 'forecast solar day ahead',
       'forecast wind offshore eday ahead', 'forecast wind onshore day ahead',
       'total load forecast', 'total load actual', 'price day ahead',
       'price actual'],
      dtype='object')
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16569 entries, 0 to 16568
Data columns (total 29 columns):
 #   Column                                       Non-Null Count  Dtype  
---  ------                                       --------------  -----  
 0   time                                         16569 non-null  object 
 1   generation biomass                           16552 non-null  float64
 2   generation fossil brown coal/lignite         16552 non-null  float64
 3   generation fossil coal-derived gas           16552 non-null  float64
 4   generation fossil gas                        16552 non-null  float64
 5   generation fossil hard coal                  16552 non-null  float64
 6   generation fossil oil                        16551 non-null  float64
 7   generation fossil oil shale                  16552 non-null  float64
 8   generation fossil peat                       16552 non-null  float64
 9   generation geothermal                        16552 non-null  float64
 10  generation hydro pumped storage aggregated   0 non-null      float64
 11  generation hydro pumped storage consumption  16551 non-null  float64
 12  generation hydro run-of-river and poundage   16551 non-null  float64
 13  generation hydro water reservoir             16552 non-null  float64
 14  generation marine                            16551 non-null  float64
 15  generation nuclear                           16553 non-null  float64
 16  generation other                             16552 non-null  float64
```

```
 17  generation other renewable                   16552 non-null  float64
 18  generation solar                             16552 non-null  float64
 19  generation waste                             16551 non-null  float64
 20  generation wind offshore                     16552 non-null  float64
 21  generation wind onshore                      16552 non-null  float64
 22  forecast solar day ahead                     16569 non-null  int64
 23  forecast wind offshore eday ahead            0 non-null      float64
 24  forecast wind onshore day ahead              16569 non-null  int64
 25  total load forecast                          16569 non-null  int64
 26  total load actual                            16537 non-null  float64
 27  price day ahead                              16569 non-null  float64
 28  price actual                                 16569 non-null  float64
dtypes: float64(25), int64(3), object(1)
memory usage: 3.7+ MB
```

```
df.describe()
```

	generation biomass	generation fossil brown coal/lignite	generation fossil coal-derived gas	generation fossil gas	generation fossil hard coal	generatic fossil oi
count	16552.000000	16552.000000	16552.0	16552.000000	16552.000000	16551.00000
mean	433.164149	431.282262	0.0	5022.886056	4453.370408	310.12941
std	91.221395	389.813397	0.0	1728.186065	2168.005375	59.50684
min	101.000000	0.000000	0.0	1518.000000	576.000000	87.00000
25%	358.000000	0.000000	0.0	3921.000000	2401.750000	271.00000
50%	448.000000	448.000000	0.0	4597.000000	4839.500000	306.00000
75%	513.000000	865.000000	0.0	5620.000000	6194.000000	354.00000
max	592.000000	997.000000	0.0	16250.000000	8359.000000	449.00000

8 rows × 28 columns

```
df.isnull().sum()
```

```
time                                         0
generation biomass                          17
generation fossil brown coal/lignite        17
generation fossil coal-derived gas          17
generation fossil gas                       17
generation fossil hard coal                 17
generation fossil oil                       18
generation fossil oil shale                 17
generation fossil peat                      17
generation geothermal                       17
generation hydro pumped storage aggregated  16569
generation hydro pumped storage consumption    18
```

```
generation hydro run-of-river and poundage        18
generation hydro water reservoir                  17
generation marine                                 18
generation nuclear                                16
generation other                                  17
generation other renewable                        17
generation solar                                  17
generation waste                                  18
generation wind offshore                          17
generation wind onshore                           17
forecast solar day ahead                           0
forecast wind offshore eday ahead              16569
forecast wind onshore day ahead                    0
total load forecast                                0
total load actual                                 32
price day ahead                                    0
price actual                                       0
dtype: int64
```

```
df = df.drop(['generation hydro pumped storage aggregated','forecast wind offshore e
```

```
df = df.dropna()
```

```
df.isnull().sum()
```

```
time                                         0
generation biomass                           0
generation fossil brown coal/lignite         0
generation fossil coal-derived gas           0
generation fossil gas                        0
generation fossil hard coal                  0
generation fossil oil                        0
generation fossil oil shale                  0
generation fossil peat                       0
generation geothermal                        0
generation hydro pumped storage consumption  0
generation hydro run-of-river and poundage    0
generation hydro water reservoir             0
generation marine                            0
generation nuclear                           0
generation other                             0
generation other renewable                   0
generation solar                             0
generation waste                             0
generation wind offshore                     0
generation wind onshore                      0
forecast solar day ahead                     0
forecast wind onshore day ahead              0
total load forecast                          0
total load actual                            0
price day ahead                              0
price actual                                 0
dtype: int64
```

```
df.nunique()
```

```
time                                         16527
generation biomass                             420
generation fossil brown coal/lignite           918
generation fossil coal-derived gas               1
generation fossil gas                         5478
generation fossil hard coal                   6521
generation fossil oil                          312
generation fossil oil shale                      1
generation fossil peat                           1
generation geothermal                            1
generation hydro pumped storage consumption   2796
generation hydro run-of-river and poundage    1586
generation hydro water reservoir              6267
generation marine                                1
generation nuclear                            1872
generation other                                96
generation other renewable                      60
generation solar                              4443
generation waste                               247
generation wind offshore                         1
generation wind onshore                       8718
forecast solar day ahead                      4480
forecast wind onshore day ahead               8703
total load forecast                           9350
total load actual                            10530
price day ahead                               4177
price actual                                  5724
dtype: int64
```

```
df = df.drop(['time'],axis = 1)
```

```
round((df.isnull().sum()/len(df)*100),2)
```

```
generation biomass                           0.0
generation fossil brown coal/lignite         0.0
generation fossil coal-derived gas           0.0
generation fossil gas                        0.0
generation fossil hard coal                  0.0
generation fossil oil                        0.0
generation fossil oil shale                  0.0
generation fossil peat                       0.0
generation geothermal                        0.0
generation hydro pumped storage consumption  0.0
generation hydro run-of-river and poundage   0.0
generation hydro water reservoir             0.0
generation marine                            0.0
generation nuclear                           0.0
generation other                             0.0
generation other renewable                   0.0
generation solar                             0.0
generation waste                             0.0
generation wind offshore                     0.0
generation wind onshore                      0.0
forecast solar day ahead                     0.0
forecast wind onshore day ahead              0.0
total load forecast                          0.0
total load actual                            0.0
price day ahead                              0.0
```

```
price actual                                    0.0
dtype: float64
```

```
df.corr()
```

neration fossil peat	generation geothermal	generation hydro pumped storage consumption	...	generation solar	generation waste	generation wind offshore	generation wind onshore
NaN	NaN	-0.084706	...	-0.042877	-0.125411	NaN	-0.040659
NaN	NaN	-0.365730	...	0.061419	0.421326	NaN	-0.489034
NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
NaN	NaN	-0.442106	...	0.054442	0.330771	NaN	-0.469760
NaN	NaN	-0.455732	...	0.072058	0.368203	NaN	-0.525176
NaN	NaN	-0.369389	...	0.085478	-0.112008	NaN	-0.077952
NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
NaN	NaN	1.000000	...	-0.249971	-0.250089	NaN	0.381634
NaN	NaN	0.066210	...	0.056022	-0.514166	NaN	0.253256

```
correlations = df.corr(method = 'pearson')

print(correlations['price actual'].sort_values(ascending=False).to_string())
```

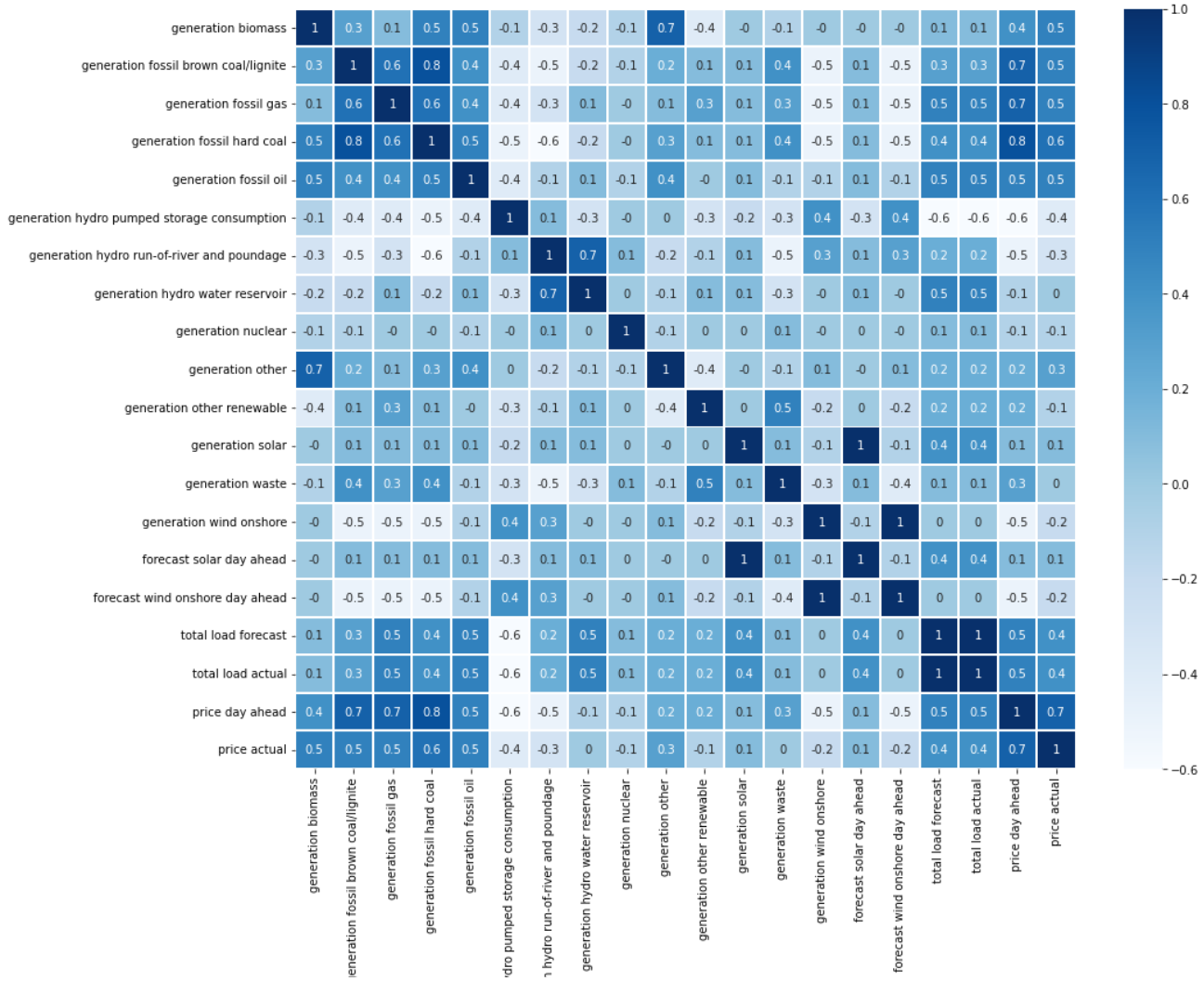
```
price actual                                1.000000
price day ahead                             0.733410
generation fossil hard coal                 0.595878
generation fossil oil                       0.478884
generation biomass                          0.459019
generation fossil gas                       0.455885
generation fossil brown coal/lignite        0.452190
```

```
total load forecast                           0.413760
total load actual                             0.410439
generation other                              0.295537
forecast solar day ahead                      0.127138
generation solar                              0.123990
generation waste                              0.025184
generation hydro water reservoir              0.015521
generation nuclear                           -0.058056
generation other renewable                   -0.100459
generation wind onshore                      -0.245027
forecast wind onshore day ahead              -0.246847
generation hydro run-of-river and poundage    -0.269146
generation hydro pumped storage consumption   -0.428538
generation fossil coal-derived gas                  NaN
generation fossil oil shale                         NaN
generation fossil peat                              NaN
generation geothermal                               NaN
generation marine                                   NaN
generation wind offshore                            NaN
```

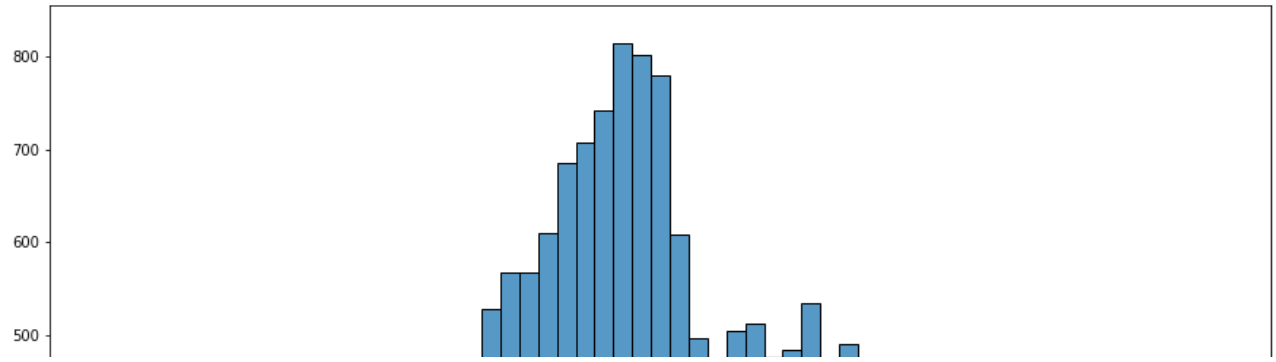
```
null_val_cols = ['generation fossil coal-derived gas',
'generation fossil oil shale',
'generation fossil peat',
'generation geothermal',
'generation marine',
'generation wind offshore' ]
```

```
heat_map_features = df.drop(columns = null_val_cols,axis = 1)
```

```
plt.figure(figsize = (15,12))
sns.heatmap(round(heat_map_features.corr(),1),annot=True,cmap='Blues',linewidth = 0.
plt.show();
```

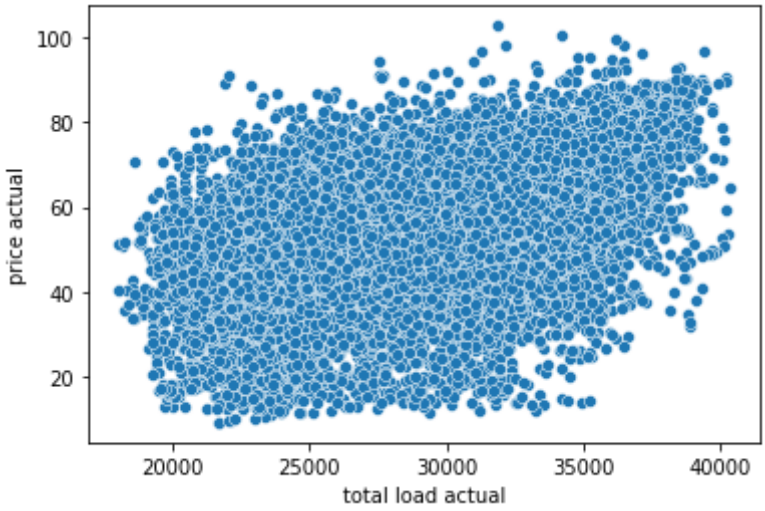



```
plt.figure(figsize=(15,10))
sns.histplot(df,x = 'price actual')
plt.show()
```




```
sns.scatterplot(x='total load actual',y = 'price actual',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f440a9ad390>
```




```
df['total load actual']
```

```
0        25385.0
1        24382.0
2        22734.0
3        21286.0
4        20264.0
          ...
16564    21697.0
16565    22441.0
16566    25567.0
16567    29818.0
16568    32074.0
Name: total load actual, Length: 16527, dtype: float64
```

```
X = df.drop(['price actual'],axis = 1)
```

```
y = df['price actual']
```

```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size = 0.2,random_state =
```

```
from sklearn.linear_model import Ridge,LinearRegression

model = LinearRegression()
model.fit(X_train,y_train)
```

```
LinearRegression()
```

```
y_pred = model.predict(X_test)
```

```
print('Training_accuracy:',model.score(X_train,y_train))
print('Testing_Accuracy:',model.score(X_test,y_test))
```

```
Training_accuracy: 0.6180465791911046
Testing_Accuracy: 0.619945448043943
```

```
from sklearn.ensemble import RandomForestRegressor
```

```
regressor = RandomForestRegressor()
regressor.fit(X_train,y_train)
```

```
RandomForestRegressor()
```

```
y_pred=regressor.predict(X_test)
```

```
print('Training Accuracy:',regressor.score(X_train,y_train))
print('Testing Accuracy:',regressor.score(X_test,y_test))
```

```
Training Accuracy: 0.981664131420906
Testing Accuracy: 0.8835498835001367
```

✓ 4s completed at 4:39 PM