In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:

```python
data = pd.read_csv("C:/Users/iafri/Documents/ds practice/water_potability.csv")
```

In [4]:

```python
data.head()
```

Out[4]:

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0

In [5]:

```python
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3276 entries, 0 to 3275
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   ph               2785 non-null   float64
 1   Hardness         3276 non-null   float64
 2   Solids           3276 non-null   float64
 3   Chloramines      3276 non-null   float64
 4   Sulfate          2495 non-null   float64
 5   Conductivity     3276 non-null   float64
 6   Organic_carbon   3276 non-null   float64
 7   Trihalomethanes  3114 non-null   float64
 8   Turbidity        3276 non-null   float64
 9   Potability       3276 non-null   int64  
dtypes: float64(9), int64(1)
memory usage: 256.1 KB
```

In [6]:

```python
data.isnull().sum()
```

Out[6]:

```
ph                 491
Hardness             0
Solids               0
Chloramines          0
Sulfate            781
Conductivity         0
Organic_carbon       0
Trihalomethanes    162
Turbidity            0
Potability           0
dtype: int64
```

In [7]:

```python
data.describe()
```

Out[7]:

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Po
count	2785.000000	3276.000000	3276.000000	3276.000000	2495.000000	3276.000000	3276.000000	3114.000000	3276.000000	3276
mean	7.080795	196.369496	22014.092526	7.122277	333.775777	426.205111	14.284970	66.396293	3.966786	0
std	1.594320	32.879761	8768.570828	1.583085	41.416840	80.824064	3.308162	16.175008	0.780382	0
min	0.000000	47.432000	320.942611	0.352000	129.000000	181.483754	2.200000	0.738000	1.450000	0
25%	6.093092	176.850538	15666.690297	6.127421	307.699498	365.734414	12.065801	55.844536	3.439711	0
50%	7.036752	196.967627	20927.833607	7.130299	333.073546	421.884968	14.218338	66.622485	3.955028	0
75%	8.062066	216.667456	27332.762127	8.114887	359.950170	481.792304	16.557652	77.337473	4.500320	1
max	14.000000	323.124000	61227.196008	13.127000	481.030642	753.342620	28.300000	124.000000	6.739000	1

In [8]:

```python
for col in ["ph", "Sulfate", "Trihalomethanes"]:
    data[col].fillna(value=data[col].mean(), inplace=True)
```

In [9]:

```python
# Reassessing Data Quality
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3276 entries, 0 to 3275
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   ph               3276 non-null   float64
 1   Hardness         3276 non-null   float64
 2   Solids           3276 non-null   float64
 3   Chloramines      3276 non-null   float64
 4   Sulfate          3276 non-null   float64
 5   Conductivity     3276 non-null   float64
 6   Organic_carbon   3276 non-null   float64
 7   Trihalomethanes  3276 non-null   float64
 8   Turbidity        3276 non-null   float64
 9   Potability       3276 non-null   int64  
dtypes: float64(9), int64(1)
memory usage: 256.1 KB
```

In [10]:

```
data.isnull().sum()
```

Out[10]:

```
ph                 0
Hardness           0
Solids             0
Chloramines        0
Sulfate            0
Conductivity       0
Organic_carbon     0
Trihalomethanes    0
Turbidity          0
Potability         0
dtype: int64
```

In [11]:

```
data["Potability"].value_counts()
```

Out[11]:

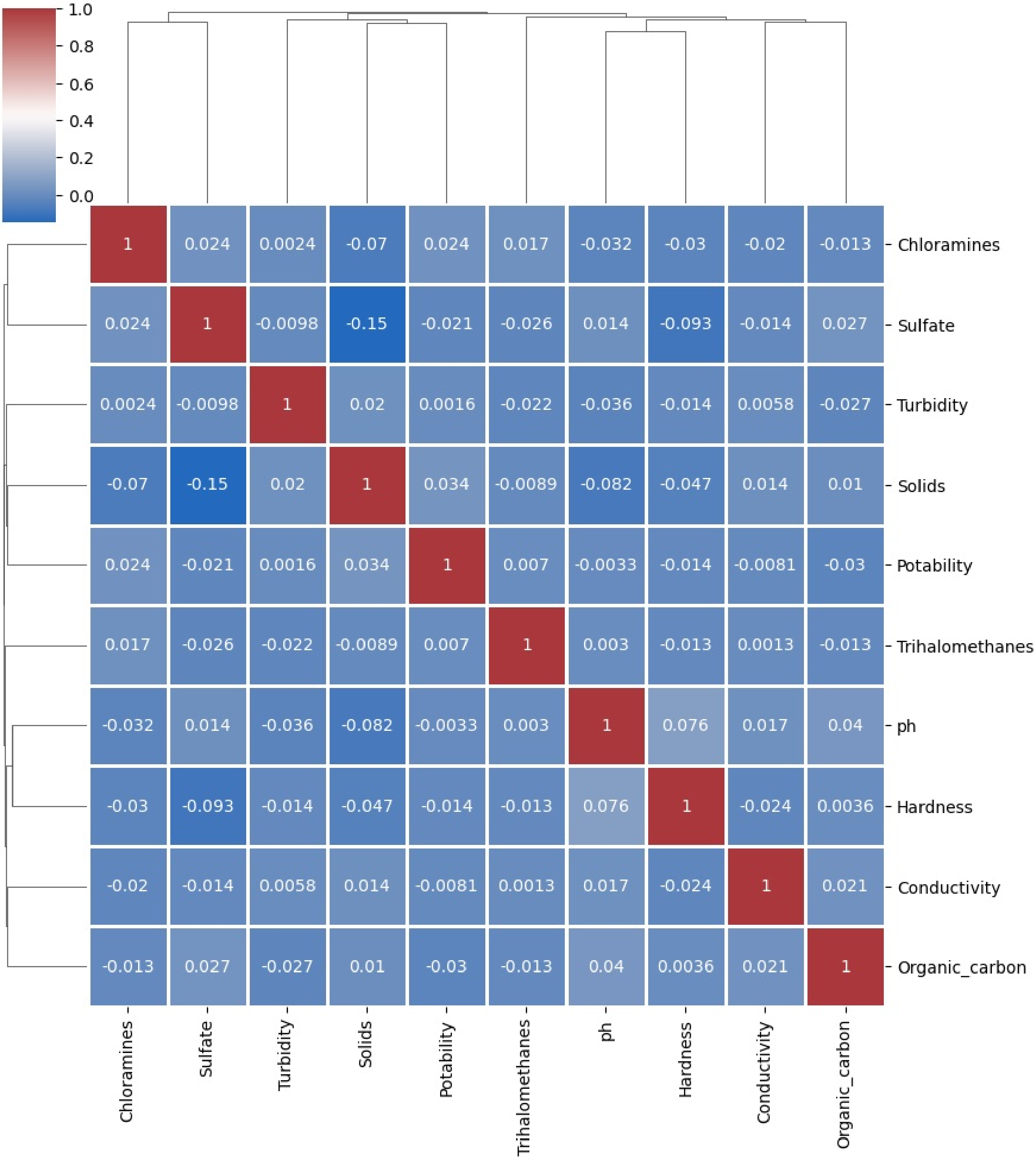
```
Potability
0    1998
1    1278
Name: count, dtype: int64
```

In [12]:

```
#Showing the correlation of the features (with missing values)
sns.clustermap(data.corr(), cmap="vlag", dendrogram_ratio=(0.1, 0.2), annot=True, linewidths=.8, figsize=(9, 10
```

Out[12]:

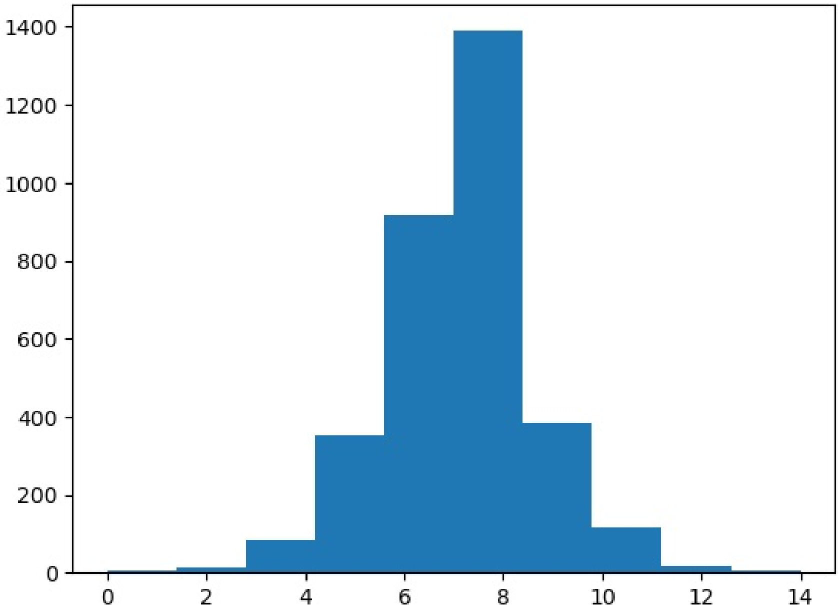
```
<seaborn.matrix.ClusterGrid at 0x214eaf14510>
```



	Chloramines	Sulfate	Turbidity	Solids	Potability	Trihalomethanes	ph	Hardness	Conductivity	Organic_carbon
Chloramines	1	0.024	0.0024	-0.07	0.024	0.017	-0.032	-0.03	-0.02	-0.013
Sulfate	0.024	1	-0.0098	-0.15	-0.021	-0.026	0.014	-0.093	-0.014	0.027
Turbidity	0.0024	-0.0098	1	0.02	0.0016	-0.022	-0.036	-0.014	0.0058	-0.027
Solids	-0.07	-0.15	0.02	1	0.034	-0.0089	-0.082	-0.047	0.014	0.01
Potability	0.024	-0.021	0.0016	0.034	1	0.007	-0.0033	-0.014	-0.0081	-0.03
Trihalomethanes	0.017	-0.026	-0.022	-0.0089	0.007	1	0.003	-0.013	0.0013	-0.013
ph	-0.032	0.014	-0.036	-0.082	-0.0033	0.003	1	0.076	0.017	0.04
Hardness	-0.03	-0.093	-0.014	-0.047	-0.014	-0.013	0.076	1	-0.024	0.0036
Conductivity	-0.02	-0.014	0.0058	0.014	-0.0081	0.0013	0.017	-0.024	1	0.021
Organic_carbon	-0.013	0.027	-0.027	0.01	-0.03	-0.013	0.04	0.0036	0.021	1

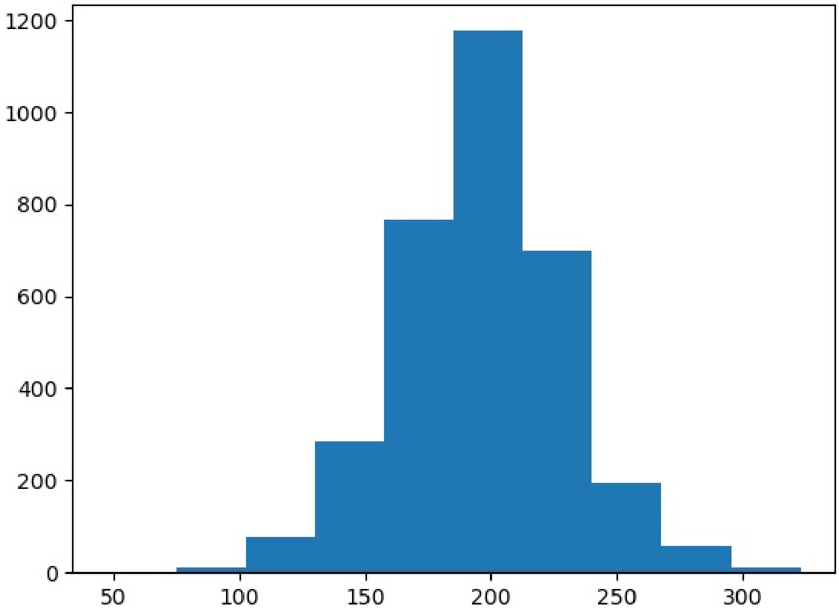
In [13]:

```
plt.figure()
plt.hist(data['ph'])
plt.show()
```



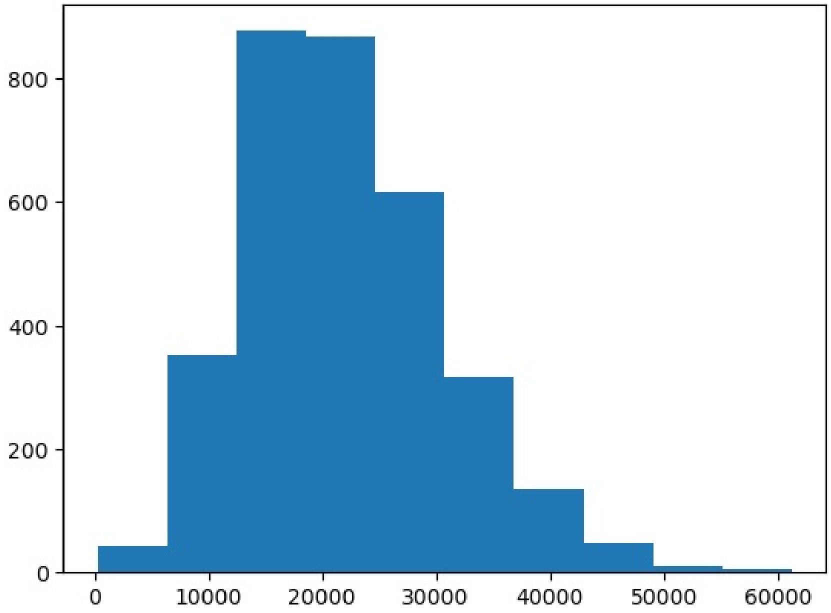
In [14]:

```
plt.figure()
plt.hist(data['Hardness'])
plt.show()
```



In [15]:

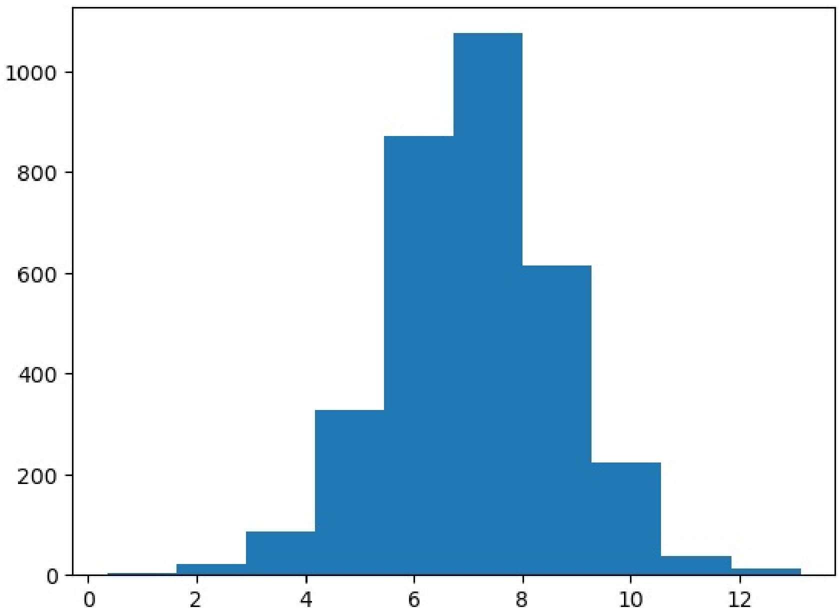
```
plt.figure()
plt.hist(data['Solids'])
plt.show()
```



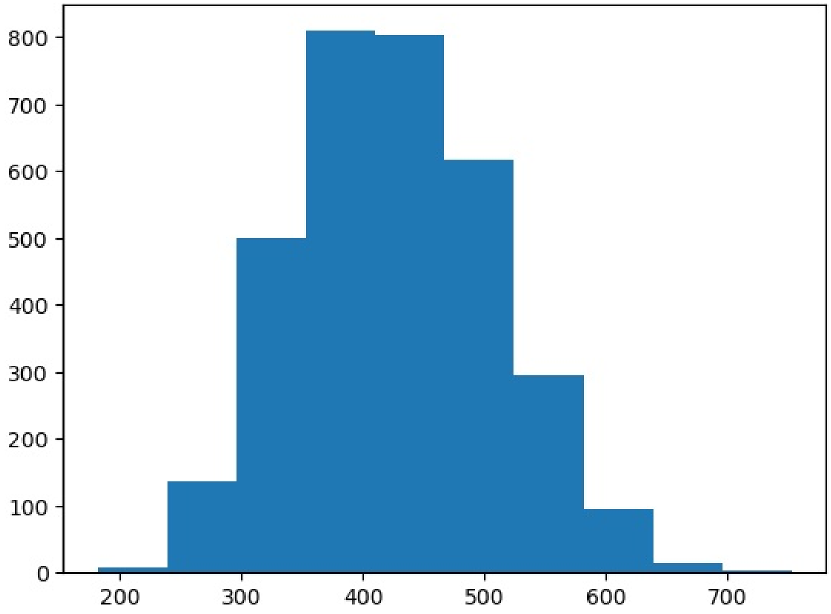
In [16]:

```
plt.figure()
```

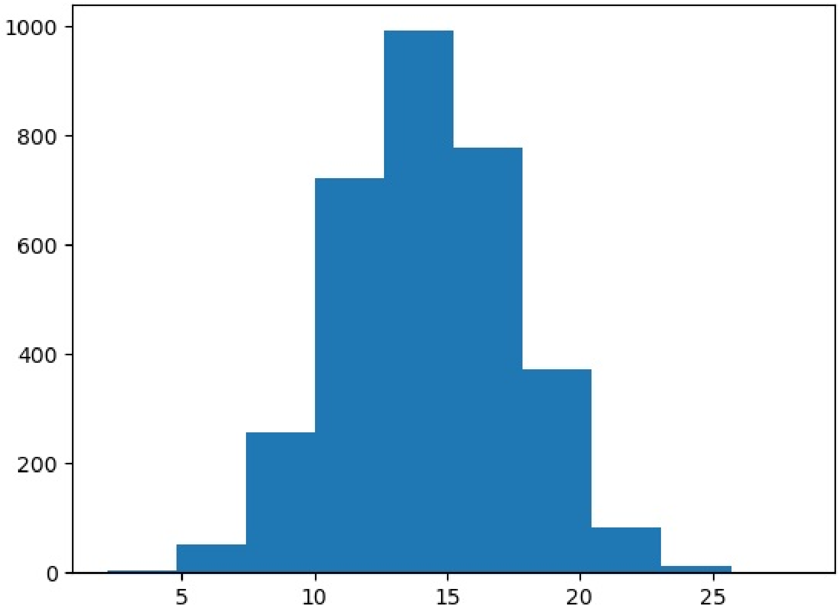
```
In [16]: plt.figure()
         plt.hist(data['Chloramines'])
         plt.show()
```



```
In [17]: plt.figure()
         plt.hist(data['Conductivity'])
         plt.show()
```

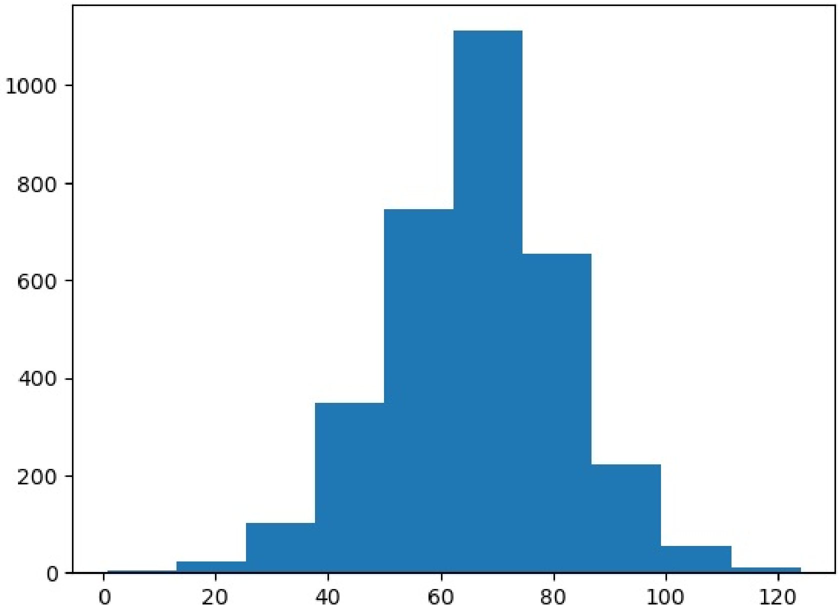


```
In [18]: plt.figure()
         plt.hist(data['Organic_carbon'])
         plt.show()
```



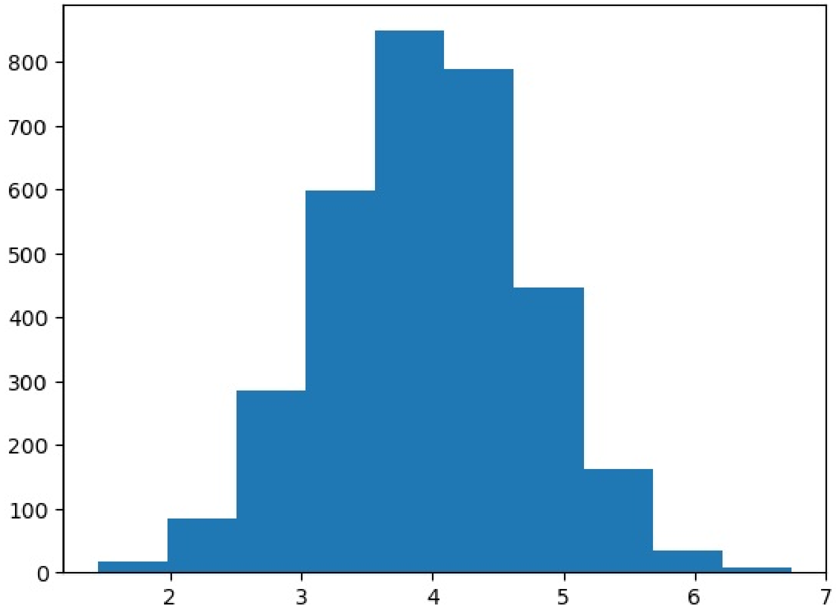
In [19]:

```
plt.figure()
plt.hist(data['Trihalomethanes'])
plt.show()
```



In [20]:

```
plt.figure()
plt.hist(data['Turbidity'])
plt.show()
```



In [ ]:

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js