In [11]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import linear_model
```

In [12]:

```
data=pd.read_excel("C:/Users/jcadmin/Downloads/Loyalty.xls")
```

In [13]:

```
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

In [14]:

```
data.head()
```

Out[14]:

	CustomerID	Loyalty	Price	Quality	Community Outreach	Trust	Customer satifaction	Negative publicity
0	920	6.075547	10.0	0.918950	-0.235777	6.39	0.769072	0.328158
1	921	6.585246	10.0	0.926412	0.006779	6.44	0.818781	0.675122
2	923	6.377699	10.0	0.881912	NaN	6.49	0.768604	0.560424
3	924	6.221095	10.0	0.888917	NaN	6.52	0.934050	NaN
4	925	6.480031	10.0	0.861948	NaN	6.55	0.750525	NaN

In [15]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1949 entries, 0 to 1948
Data columns (total 8 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   CustomerID            1949 non-null   int64  
 1   Loyalty               1949 non-null   float64
 2   Price                 1913 non-null   float64
 3   Quality               1936 non-null   float64
 4   Community Outreach    1860 non-null   float64
 5   Trust                 1894 non-null   float64
 6   Customer satifaction  1917 non-null   float64
 7   Negative publicity    1839 non-null   float64
dtypes: float64(7), int64(1)
memory usage: 121.9 KB
```

In [16]:

```
data.shape
```

Out[16]:

```
(1949, 8)
```

In [17]:

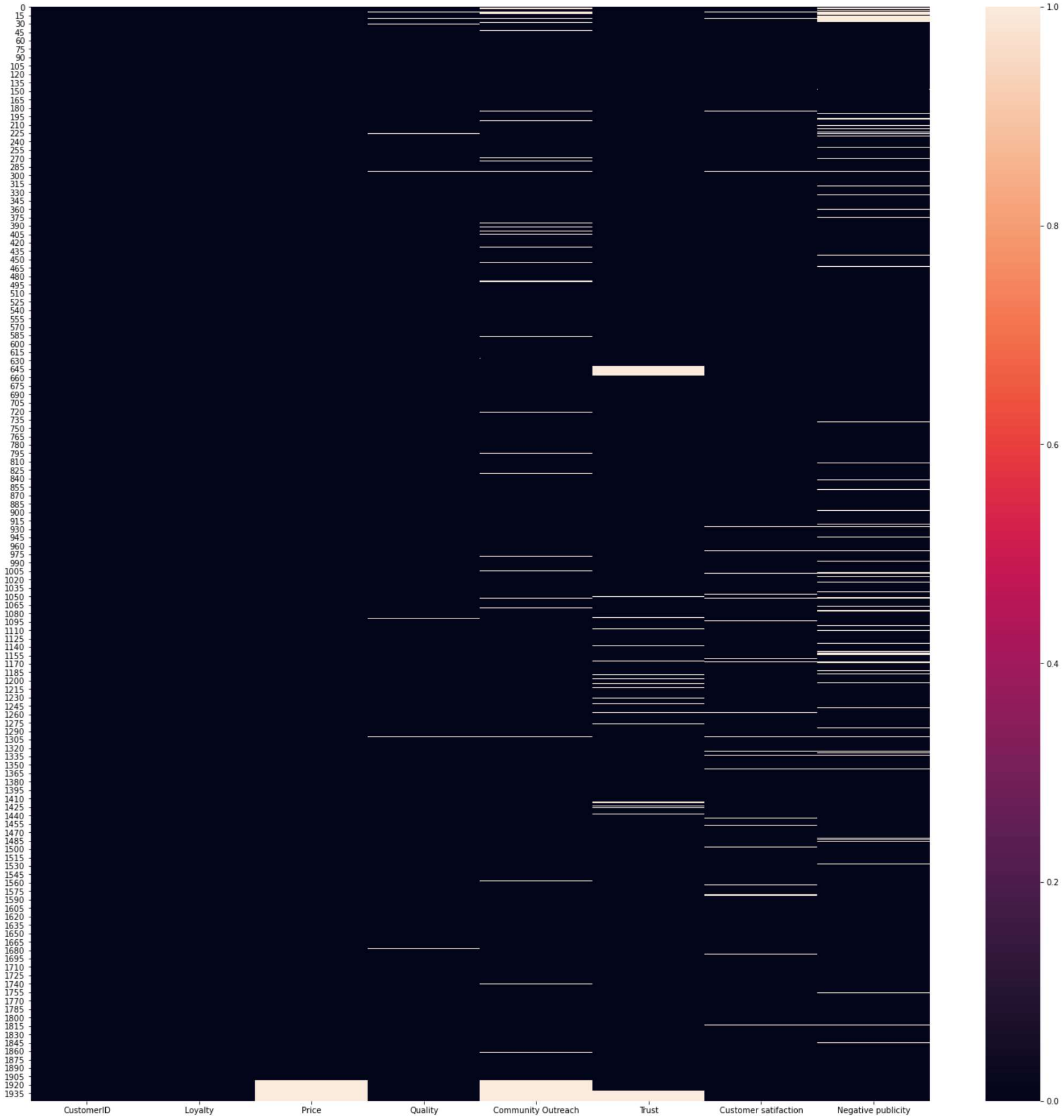
```
data.isnull().sum()
```

Out[17]:

```
CustomerID                0
Loyalty                   0
Price                    36
Quality                  13
Community Outreach       89
Trust                    55
Customer satifaction     32
Negative publicity      110
dtype: int64
```

In [18]:
```
plt.figure(figsize=(25,25))
sns.heatmap(data.isnull())
```

Out[18]:
```
<AxesSubplot:>
```



In [19]:
```
null_var=data.isnull().sum()/data.shape[0]*100
null_var
```

Out[19]:
```
CustomerID              0.000000
Loyalty                 0.000000
Price                   1.847101
Quality                 0.667009
Community Outreach      4.566444
Trust                   2.821960
Customer satifaction    1.641868
Negative publicity      5.643920
dtype: float64
```

In [20]:
```
data2=data.dropna()
```

In [21]:
```
data2.shape
```

Out[21]:
```
(1712, 8)
```

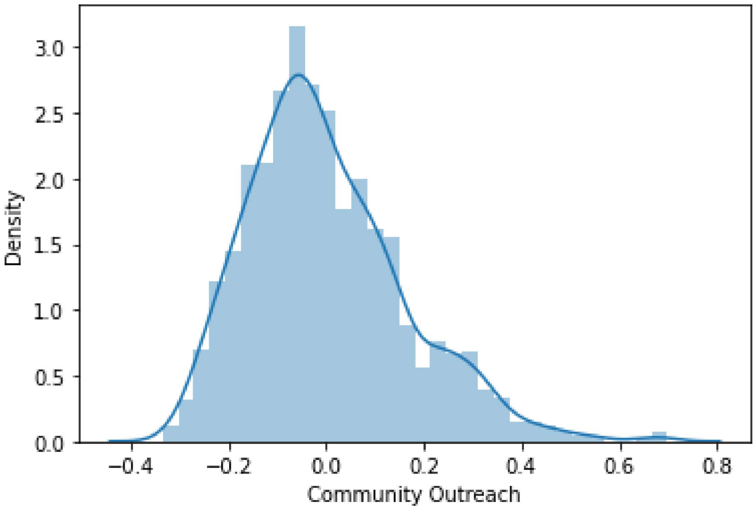
In [22]:
```
null_var=data2.isnull().sum()/data2.shape[0]*100
null_var
```

Out[22]:
```
CustomerID               0.0
Loyalty                  0.0
Price                    0.0
Quality                  0.0
Community Outreach       0.0
Trust                    0.0
Customer satifaction     0.0
Negative publicity       0.0
dtype: float64
```

In [23]:
```
sns.distplot(data['Community Outreach'])
```

```
C:\Users\jcadmin\Anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Please a
dapt your code to use either `displot` (a figure-level function with similar flexibilit
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

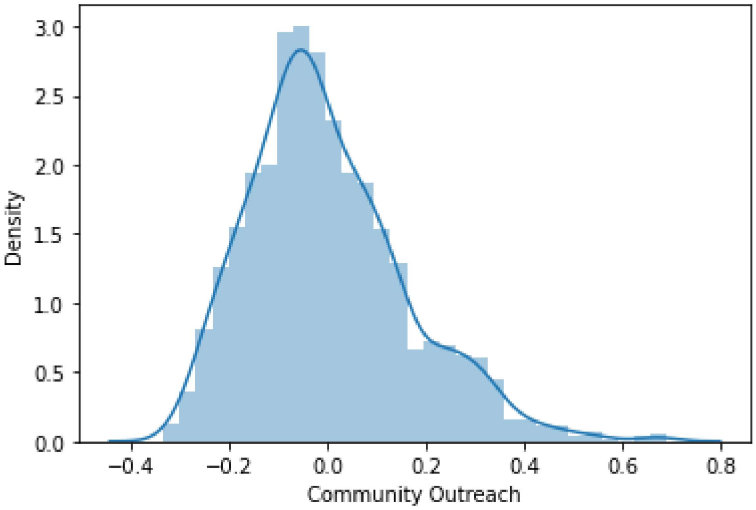
Out[23]:
```
<AxesSubplot:xlabel='Community Outreach', ylabel='Density'>
```



In [24]:
```
sns.distplot(data2['Community Outreach'])
```

```
C:\Users\jcadmin\Anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Please a
dapt your code to use either `displot` (a figure-level function with similar flexibilit
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[24]:
```
<AxesSubplot:xlabel='Community Outreach', ylabel='Density'>
```



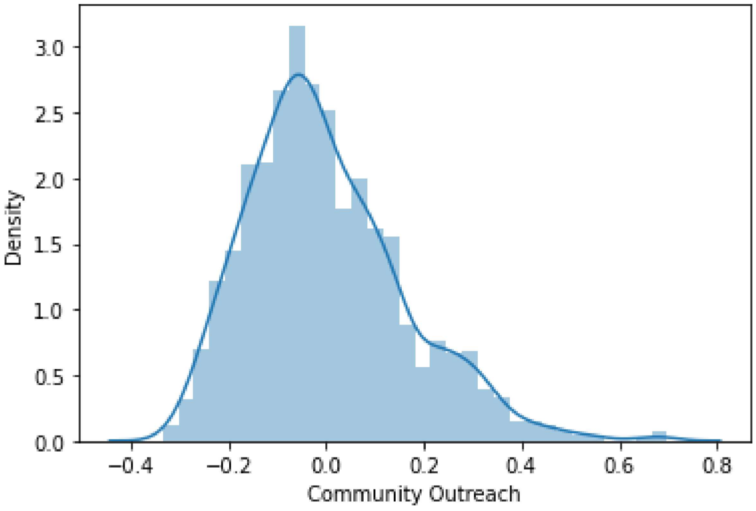
In [25]:

```
sns.distplot(data['Community Outreach'])
sns.distplot(df2['Community Outreach'])
```

```
C:\Users\jcadmin\Anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarnin
g: `distplot` is a deprecated function and will be removed in a future version. Please a
dapt your code to use either `displot` (a figure-level function with similar flexibilit
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-25-f69f358e50cd> in <module>
      1 sns.distplot(data['Community Outreach'])
----> 2 sns.distplot(df2['Community Outreach'])

NameError: name 'df2' is not defined
```



In [ ]:

```
data.columns
```

In [ ]:

```
num_var = ['Loyalty', 'Price', 'Quality', 'Community Outreach',
       'Trust', 'Customer satifaction', 'Negative publicity']

plt.figure(figsize=(25,25))
for i,var in enumerate(num_var):
    plt.subplot(2,4,i+1)
    sns.distplot(data[var],bins=20)
    sns.distplot(data2[var],bins=20)
```

In [ ]:

```
plt.figure(figsize=(12,10))
cor = data2.corr()
sns.heatmap(cor, annot=True, cmap=plt.cm.Reds)
plt.show()
```

In [ ]:

```
data2.plot(kind='scatter',x='Trust',y="Price")
plt.show()
```

In [ ]:

```
data2.plot(kind='box',x='Trust',y='Price')
plt.show()
```

In [ ]:

```
data_with_corr_column = data2.filter(['Trust','Price'], axis=1)
data_with_corr_column.head()
```

In [ ]:

```
data4=pd.DataFrame(data_with_corr_column)
data4.head()
```

In [ ]:

```
from sklearn.linear_model import LinearRegression
```

In [ ]:

```
feature_cols=['Trust']
x= data4[feature_cols]
y = data4.Price
```

In [ ]:

```
model=LinearRegression()
model.fit(x,y)
print(model.intercept_)
print(model.coef_)
```

In [ ]:

```
model.predict([[5]])
```

In [ ]:

```
model.score(x, y)
```