In [30]:

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

%matplotlib inline
```

In [31]:

```
store_df = pd.read_csv('cleaned_gstore_data.csv',low_memory=False)
```

In [32]:

```
store_df.head()
```

Out[32]:

	channelGrouping	date	fullVisitorId	visitId	visitNumber	visitStartTime	device.browser	device.operatingSystem	de
0	Organic Search	2016-09-02	1131660440785968503	1472830385	1	1472830385	Chrome	Windows	
1	Organic Search	2016-09-02	377306020877927890	1472880147	1	1472880147	Firefox	Macintosh	
2	Organic Search	2016-09-02	3895546263509774583	1472865386	1	1472865386	Chrome	Windows	
3	Organic Search	2016-09-02	4763447161404445595	1472881213	1	1472881213	UC Browser	Linux	
4	Organic Search	2016-09-02	27294437909732085	1472822600	2	1472822600	Chrome	Android	

5 rows × 31 columns

In [33]:

```
store_df.shape
```

Out[33]:

```
(903653, 31)
```

In [34]:

```
rev_df = store_df.groupby('fullVisitorId')['totals.transactionRevenue'].sum().reset_index()
```

In [35]:

```
rev_df
```

Out[35]:

	fullVisitorId	totals.transactionRevenue
0	0000010278554503158	0.0
1	0000020424342248747	0.0
2	0000027376579751715	0.0
3	0000039460501403861	0.0
4	0000040862739425590	0.0
...	...	...
714162	9999963186378918199	0.0

	fullVisitorId	totals.transactionRevenue
714163	9999972259709566660	0.0
714164	9999975500403964600	0.0
714165	9999978264901065827	0.0
714166	9999986437109498564	0.0

714167 rows × 2 columns

In [36]:

```
# CUSTOMERS WHO ARE ACTUALLY GENERATING REVENUE

positive_rev_df = rev_df[rev_df['totals.transactionRevenue']>0.0].reset_index(drop=True)
```

positive_rev_df.head()

In [37]:

```
positive_rev_df.shape
```

Out[37]:

```
(9996, 2)
```

In [38]:

```
# PERCENTAGE WHO GENERATE POSITIVE REVENUE

(9996/714167)*100
```

Out[38]:

```
1.399672625590373
```

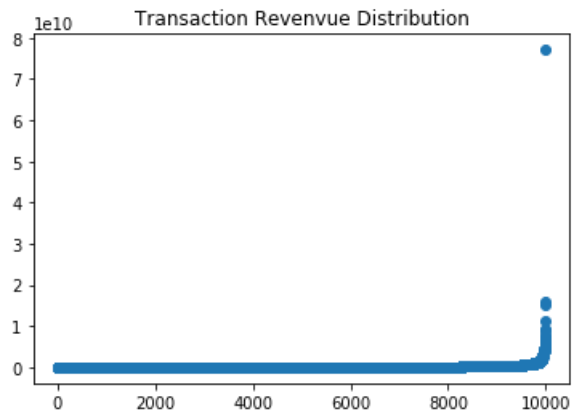
Revenue data is highly imbalanced. Only 1.39 % of total customers generate revenvue.

In [39]:

```
#plotting a scatter plot

x_axis_val = range(positive_rev_df.shape[0])

y_axis_val = np.sort(positive_rev_df['totals.transactionRevenue'].values)

plt.scatter(x_axis_val, y_axis_val)
plt.title('Transaction Revenvue Distribution')
```

Out[39]:

```
Text(0.5, 1.0, 'Transaction Revenvue Distribution')
```



Since the data is skewed, I will be taking logarithmmic value of the data.

Since the data is skewed, I will be taking logarithmic value of the data.

In [40]:

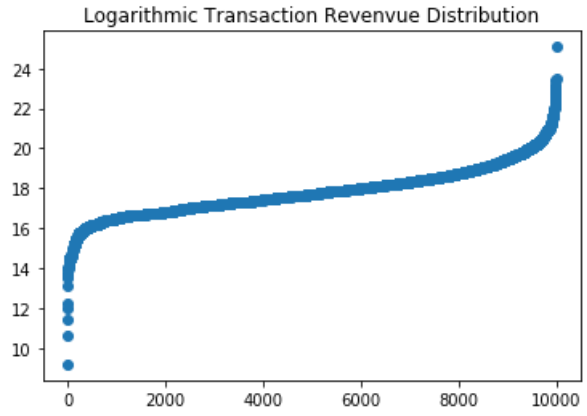
```
positive_rev_df['totals.transactionRevenue']=np.log1p(positive_rev_df['totals.transactionRevenue']
.values)
```

In [41]:

```
#plotting a scatter plot

x_axis_val = range(positive_rev_df.shape[0])

y_axis_val = np.sort(positive_rev_df['totals.transactionRevenue'].values)

plt.scatter(x_axis_val, y_axis_val)
plt.title('Logarithmic Transaction Revenvue Distribution')
```

Out[41]:

```
Text(0.5, 1.0, 'Logarithmic Transaction Revenvue Distribution')
```



In [42]:

```
store_df.corr()['totals.transactionRevenue']
```

Out[42]:

```
visitId                              0.002724
visitNumber                          0.051366
visitStartTime                       0.002724
device.isMobile                     -0.016555
totals.hits                          0.154333
totals.pageviews                     0.155590
totals.bounces                      -0.032206
totals.newVisits                    -0.041164
totals.transactionRevenue            1.000000
trafficSource.isTrueDirect           0.030819
trafficSource.adwordsClickInfo.page  0.000775
Name: totals.transactionRevenue, dtype: float64
```

In [43]:

```
store_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 903653 entries, 0 to 903652
Data columns (total 31 columns):
 #   Column                                    Non-Null Count   Dtype
---  ------                                    --------------   -----
 0   channelGrouping                           903653 non-null  object
 1   date                                      903653 non-null  object
 2   fullVisitorId                             903653 non-null  object
 3   visitId                                   903653 non-null  int64
 4   visitNumber                               903653 non-null  int64
 5   visitStartTime                            903653 non-null  int64
```

```
 6   device.browser                              903653 non-null  object
 7   device.operatingSystem                      903653 non-null  object
 8   device.isMobile                             903653 non-null  bool
 9   device.deviceCategory                       903653 non-null  object
 10  geoNetwork.continent                        903653 non-null  object
 11  geoNetwork.subContinent                     903653 non-null  object
 12  geoNetwork.country                          903653 non-null  object
 13  geoNetwork.region                           903653 non-null  object
 14  geoNetwork.metro                            903653 non-null  object
 15  geoNetwork.city                             903653 non-null  object
 16  geoNetwork.networkDomain                    903653 non-null  object
 17  totals.hits                                 903653 non-null  int64
 18  totals.pageviews                            903653 non-null  int64
 19  totals.bounces                              903653 non-null  float64
 20  totals.newVisits                            903653 non-null  int64
 21  totals.transactionRevenue                   903653 non-null  float64
 22  trafficSource.campaign                      903653 non-null  object
 23  trafficSource.source                        903653 non-null  object
 24  trafficSource.medium                        903653 non-null  object
 25  trafficSource.keyword                       903653 non-null  object
 26  trafficSource.isTrueDirect                  903653 non-null  bool
 27  trafficSource.adwordsClickInfo.page         903653 non-null  int64
 28  trafficSource.adwordsClickInfo.slot         903653 non-null  object
 29  trafficSource.adwordsClickInfo.adNetworkType  903653 non-null  object
 30  trafficSource.adwordsClickInfo.isVideoAd    903653 non-null  object
dtypes: bool(2), float64(2), int64(7), object(20)
memory usage: 201.7+ MB
```

In [44]:

```
cat_cols = ["channelGrouping","fullVisitorId","device.browser","device.operatingSystem",
            "device.deviceCategory","geoNetwork.continent","geoNetwork.subContinent","geoNetwork.cou
ntry",
            "geoNetwork.region","geoNetwork.metro","geoNetwork.city","geoNetwork.networkDomain",
            "trafficSource.campaign","trafficSource.source","trafficSource.medium","trafficSource.ke
yword",
            "trafficSource.adwordsClickInfo.slot","trafficSource.adwordsClickInfo.adNetworkType",
            "trafficSource.adwordsClickInfo.isVideoAd"]
```

In [45]:

```
for col in cat_cols:
    # label encoder
    lbl = preprocessing.LabelEncoder()

    store_df[col] = lbl.fit_transform(store_df[col].astype(str).values)
```

Hence, the categorical columns are label enabelled

In [46]:

```
store_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 903653 entries, 0 to 903652
Data columns (total 31 columns):
 #   Column                                      Non-Null Count   Dtype
---  ------                                      --------------   -----
 0   channelGrouping                             903653 non-null  int32
 1   date                                        903653 non-null  object
 2   fullVisitorId                               903653 non-null  int32
 3   visitId                                     903653 non-null  int64
 4   visitNumber                                 903653 non-null  int64
 5   visitStartTime                              903653 non-null  int64
 6   device.browser                              903653 non-null  int32
 7   device.operatingSystem                      903653 non-null  int32
 8   device.isMobile                             903653 non-null  bool
 9   device.deviceCategory                       903653 non-null  int32
 10  geoNetwork.continent                        903653 non-null  int32
 11  geoNetwork.subContinent                     903653 non-null  int32
 12  geoNetwork.country                          903653 non-null  int32
 13  geoNetwork.region                           903653 non-null  int32
 14  geoNetwork.metro                            903653 non-null  int32
```

```
 15  geoNetwork.city                                903653 non-null  int32
 16  geoNetwork.networkDomain                       903653 non-null  int32
 17  totals.hits                                    903653 non-null  int64
 18  totals.pageviews                               903653 non-null  int64
 19  totals.bounces                                 903653 non-null  float64
 20  totals.newVisits                               903653 non-null  int64
 21  totals.transactionRevenue                      903653 non-null  float64
 22  trafficSource.campaign                         903653 non-null  int32
 23  trafficSource.source                           903653 non-null  int32
 24  trafficSource.medium                           903653 non-null  int32
 25  trafficSource.keyword                          903653 non-null  int32
 26  trafficSource.isTrueDirect                     903653 non-null  bool
 27  trafficSource.adwordsClickInfo.page            903653 non-null  int64
 28  trafficSource.adwordsClickInfo.slot            903653 non-null  int32
 29  trafficSource.adwordsClickInfo.adNetworkType   903653 non-null  int32
 30  trafficSource.adwordsClickInfo.isVideoAd       903653 non-null  int32
dtypes: bool(2), float64(2), int32(19), int64(7), object(1)
memory usage: 136.2+ MB
```

In [47]:

```
store_df.head()
```

Out[47]:

	channelGrouping	date	fullVisitorId	visitId	visitNumber	visitStartTime	device.browser	device.operatingSystem	device.isMob
0	4	2016-09-02	80509	1472830385	1	1472830385	11	16	Fal
1	4	2016-09-02	269007	1472880147	1	1472880147	16	7	Fal
2	4	2016-09-02	277678	1472865386	1	1472865386	11	16	Fal
3	4	2016-09-02	339713	1472881213	1	1472881213	46	6	Fal
4	4	2016-09-02	194517	1472822600	2	1472822600	11	1	Tr

5 rows × 31 columns

In [48]:

```
store_df.corr()['totals.transactionRevenue']
```

Out[48]:

```
channelGrouping                    -0.006644
fullVisitorId                      -0.000599
visitId                             0.002724
visitNumber                         0.051366
visitStartTime                      0.002724
device.browser                     -0.015120
device.operatingSystem             -0.010699
device.isMobile                    -0.016555
device.deviceCategory              -0.015580
geoNetwork.continent               -0.025523
geoNetwork.subContinent            -0.009144
geoNetwork.country                  0.022395
geoNetwork.region                  -0.006807
geoNetwork.metro                    0.004381
geoNetwork.city                    -0.003327
geoNetwork.networkDomain           -0.020174
totals.hits                         0.154333
totals.pageviews                    0.155590
totals.bounces                     -0.032206
totals.newVisits                   -0.041164
totals.transactionRevenue           1.000000
trafficSource.campaign             -0.003823
trafficSource.source               -0.008393
trafficSource.medium               -0.008569
trafficSource.keyword              -0.002485
trafficSource.isTrueDirect          0.030819
```

```
trafficSource.adwordsClickInfo.page              0.000775
trafficSource.adwordsClickInfo.slot              0.000870
trafficSource.adwordsClickInfo.adNetworkType    -0.000837
trafficSource.adwordsClickInfo.isVideoAd        -0.000834
Name: totals.transactionRevenue, dtype: float64
```

In [49]:

```
#storing it to a csv file

store_df.to_csv('preprocessed_gstoredata.csv',header=True,index=False)
```

In [ ]:

```

```