In [127…
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

#import warnings
#warnings.filterwarnings("ignore")
```

In [127…
```
data = pd.read_excel('leads_data.xlsx')
```

In [127…
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30773 entries, 0 to 30772
Data columns (total 23 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   LeadId                    30773 non-null  int64
 1   VisitorId                 30773 non-null  object
 2   BornDate                  30773 non-null  datetime64[ns]
 3   BornDateTime              30773 non-null  datetime64[ns]
 4   Project                   30773 non-null  object
 5   Stage                     30773 non-null  object
 6   CountOfClickEvents        30773 non-null  int64
 7   WebTimeSpent (seconds)    30773 non-null  float64
 8   UTM-Source                30773 non-null  object
 9   InstanceNumber            30773 non-null  int64
 10  HOME_TimeSpent            30773 non-null  float64
 11  LOCATION_TimeSpent        30773 non-null  float64
 12  MEDIA_TimeSpent           30773 non-null  float64
 13  PLAN_TimeSpent            30773 non-null  float64
 14  PRICE_TimeSpent           30773 non-null  float64
 15  SPECIFICATIONS_TimeSpent  30773 non-null  float64
 16  AMENITIES_TimeSpent       30773 non-null  float64
 17  MediaTime                 30773 non-null  float64
 18  OperatingSystem           30773 non-null  object
 19  Country                   30773 non-null  object
 20  State                     30773 non-null  object
 21  City                      30773 non-null  object
 22  MicroMarket               30773 non-null  object
dtypes: datetime64[ns](2), float64(9), int64(3), object(9)
memory usage: 5.4+ MB
```

In [127…
```
data.shape
```

Out[1277]:
```
(30773, 23)
```

In [127…
```
data.isna().sum()
# No Null values in data set
```

Out[1278]:
```
LeadId                        0
VisitorId                     0
BornDate                      0
BornDateTime                  0
Project                       0
Stage                         0
CountOfClickEvents            0
WebTimeSpent (seconds)        0
UTM-Source                    0
InstanceNumber                0
HOME_TimeSpent                0
LOCATION_TimeSpent            0
MEDIA_TimeSpent               0
PLAN_TimeSpent                0
PRICE_TimeSpent               0
SPECIFICATIONS_TimeSpent      0
AMENITIES_TimeSpent           0
MediaTime                     0
OperatingSystem               0
Country                       0
State                         0
City                          0
MicroMarket                   0
dtype: int64
```

In [127…
```
data.duplicated().sum()
```

Out[1279]:
```
0
```

In [128…
```
data.head()
```

Out[1280]:

	LeadId	VisitorId	BornDate	BornDateTime	Project	Stage	CountOfClickEvents	WebTime(sec
0	198611	Visitor-1003563	2024-05-30	2024-05-30 12:52:10	Springs	Not Interested	15	50
1	193927	Visitor-1006753	2024-05-10	2024-05-10 16:07:47	Spectra	Not Interested	5	5
2	230525	Visitor-1017271	2024-08-24	2024-08-24 19:42:04	Springs	Not Interested	18	29
3	208705	Visitor-1029567	2024-07-10	2024-07-10 12:36:38	Spectra	Not Interested	6	17
4	253755	Visitor-1044910	2024-09-22	2024-09-22 14:30:57	Spectra	Not Interested	49	97

5 rows × 23 columns

In [128…
```
data[data['Stage'].isin (['Sales Closure','Pre Site Visit','Post Site Visit','Flat
```

Out[1281]:
```
(178, 23)
```

In [128…
```
data[data['Stage']=='Not Interested'].shape
```

Out[1282]:
```
(30595, 23)
```

In [128…

```
for i in data.columns:
    print("Number of unique values in",i,"columns are:")
    print(data[i].nunique())
    print(data[i].unique())
    print('-'*80)
```

```
Number of unique values in LeadId columns are:
30773
[198611 193927 230525 ... 213048 234936 198230]
--------------------------------------------------------------------------------
Number of unique values in VisitorId columns are:
30770
['Visitor-1003563' 'Visitor-1006753' 'Visitor-1017271' ...
 'Visitor-926471' 'Visitor-944939' 'Visitor-977114']
--------------------------------------------------------------------------------
Number of unique values in BornDate columns are:
184
['2024-05-30T00:00:00.000000000' '2024-05-10T00:00:00.000000000'
 '2024-08-24T00:00:00.000000000' '2024-07-10T00:00:00.000000000'
 '2024-09-22T00:00:00.000000000' '2024-09-25T00:00:00.000000000'
 '2024-05-20T00:00:00.000000000' '2024-09-05T00:00:00.000000000'
 '2024-07-24T00:00:00.000000000' '2024-10-01T00:00:00.000000000'
 '2024-07-05T00:00:00.000000000' '2024-07-22T00:00:00.000000000'
 '2024-06-28T00:00:00.000000000' '2024-06-30T00:00:00.000000000'
 '2024-09-29T00:00:00.000000000' '2024-04-25T00:00:00.000000000'
 '2024-09-24T00:00:00.000000000' '2024-07-16T00:00:00.000000000'
 '2024-09-21T00:00:00.000000000' '2024-07-03T00:00:00.000000000'
 '2024-06-27T00:00:00.000000000' '2024-07-04T00:00:00.000000000'
 '2024-06-11T00:00:00.000000000' '2024-05-01T00:00:00.000000000'
 '2024-08-03T00:00:00.000000000' '2024-07-27T00:00:00.000000000'
 '2024-04-26T00:00:00.000000000' '2024-04-27T00:00:00.000000000'
 '2024-05-26T00:00:00.000000000' '2024-06-19T00:00:00.000000000'
 '2024-04-22T00:00:00.000000000' '2024-07-11T00:00:00.000000000'
 '2024-08-17T00:00:00.000000000' '2024-07-28T00:00:00.000000000'
 '2024-04-21T00:00:00.000000000' '2024-08-08T00:00:00.000000000'
 '2024-10-04T00:00:00.000000000' '2024-10-03T00:00:00.000000000'
 '2024-10-02T00:00:00.000000000' '2024-09-28T00:00:00.000000000'
 '2024-06-29T00:00:00.000000000' '2024-09-08T00:00:00.000000000'
 '2024-06-24T00:00:00.000000000' '2024-10-05T00:00:00.000000000'
 '2024-08-11T00:00:00.000000000' '2024-07-23T00:00:00.000000000'
 '2024-09-20T00:00:00.000000000' '2024-07-14T00:00:00.000000000'
 '2024-08-14T00:00:00.000000000' '2024-04-23T00:00:00.000000000'
 '2024-06-01T00:00:00.000000000' '2024-06-06T00:00:00.000000000'
 '2024-05-23T00:00:00.000000000' '2024-04-17T00:00:00.000000000'
 '2024-08-10T00:00:00.000000000' '2024-07-26T00:00:00.000000000'
 '2024-09-01T00:00:00.000000000' '2024-08-25T00:00:00.000000000'
 '2024-09-23T00:00:00.000000000' '2024-10-11T00:00:00.000000000'
 '2024-07-29T00:00:00.000000000' '2024-05-17T00:00:00.000000000'
 '2024-08-09T00:00:00.000000000' '2024-07-06T00:00:00.000000000'
 '2024-08-30T00:00:00.000000000' '2024-04-18T00:00:00.000000000'
 '2024-09-18T00:00:00.000000000' '2024-04-24T00:00:00.000000000'
 '2024-10-06T00:00:00.000000000' '2024-07-09T00:00:00.000000000'
 '2024-04-30T00:00:00.000000000' '2024-06-09T00:00:00.000000000'
 '2024-06-22T00:00:00.000000000' '2024-09-10T00:00:00.000000000'
 '2024-09-16T00:00:00.000000000' '2024-06-07T00:00:00.000000000'
 '2024-05-28T00:00:00.000000000' '2024-08-13T00:00:00.000000000'
 '2024-07-25T00:00:00.000000000' '2024-08-26T00:00:00.000000000'
 '2024-05-18T00:00:00.000000000' '2024-08-18T00:00:00.000000000'
 '2024-06-20T00:00:00.000000000' '2024-06-03T00:00:00.000000000'
 '2024-04-16T00:00:00.000000000' '2024-09-07T00:00:00.000000000'
 '2024-08-02T00:00:00.000000000' '2024-04-20T00:00:00.000000000'
 '2024-04-28T00:00:00.000000000' '2024-07-21T00:00:00.000000000'
 '2024-06-14T00:00:00.000000000' '2024-06-23T00:00:00.000000000'
 '2024-08-22T00:00:00.000000000' '2024-05-15T00:00:00.000000000'
 '2024-05-07T00:00:00.000000000' '2024-09-13T00:00:00.000000000'
 '2024-04-15T00:00:00.000000000' '2024-06-16T00:00:00.000000000'
 '2024-10-12T00:00:00.000000000' '2024-05-25T00:00:00.000000000'
 '2024-05-19T00:00:00.000000000' '2024-06-10T00:00:00.000000000'
 '2024-10-09T00:00:00.000000000' '2024-09-11T00:00:00.000000000'
 '2024-08-31T00:00:00.000000000' '2024-08-27T00:00:00.000000000'
```

```
 '2024-08-28T00:00:00.000000000' '2024-06-21T00:00:00.000000000'
 '2024-06-15T00:00:00.000000000' '2024-05-13T00:00:00.000000000'
 '2024-05-05T00:00:00.000000000' '2024-07-20T00:00:00.000000000'
 '2024-06-02T00:00:00.000000000' '2024-08-20T00:00:00.000000000'
 '2024-09-27T00:00:00.000000000' '2024-05-27T00:00:00.000000000'
 '2024-09-15T00:00:00.000000000' '2024-07-01T00:00:00.000000000'
 '2024-04-19T00:00:00.000000000' '2024-05-22T00:00:00.000000000'
 '2024-08-04T00:00:00.000000000' '2024-08-19T00:00:00.000000000'
 '2024-06-13T00:00:00.000000000' '2024-09-02T00:00:00.000000000'
 '2024-10-08T00:00:00.000000000' '2024-08-23T00:00:00.000000000'
 '2024-07-30T00:00:00.000000000' '2024-05-11T00:00:00.000000000'
 '2024-06-05T00:00:00.000000000' '2024-04-29T00:00:00.000000000'
 '2024-07-18T00:00:00.000000000' '2024-09-03T00:00:00.000000000'
 '2024-09-14T00:00:00.000000000' '2024-07-13T00:00:00.000000000'
 '2024-05-08T00:00:00.000000000' '2024-05-06T00:00:00.000000000'
 '2024-07-02T00:00:00.000000000' '2024-06-26T00:00:00.000000000'
 '2024-08-01T00:00:00.000000000' '2024-06-12T00:00:00.000000000'
 '2024-05-09T00:00:00.000000000' '2024-08-21T00:00:00.000000000'
 '2024-05-14T00:00:00.000000000' '2024-05-03T00:00:00.000000000'
 '2024-06-17T00:00:00.000000000' '2024-10-10T00:00:00.000000000'
 '2024-08-05T00:00:00.000000000' '2024-05-31T00:00:00.000000000'
 '2024-05-12T00:00:00.000000000' '2024-10-13T00:00:00.000000000'
 '2024-08-16T00:00:00.000000000' '2024-07-08T00:00:00.000000000'
 '2024-08-06T00:00:00.000000000' '2024-08-12T00:00:00.000000000'
 '2024-05-16T00:00:00.000000000' '2024-09-04T00:00:00.000000000'
 '2024-06-25T00:00:00.000000000' '2024-07-07T00:00:00.000000000'
 '2024-10-07T00:00:00.000000000' '2024-07-15T00:00:00.000000000'
 '2024-09-17T00:00:00.000000000' '2024-06-08T00:00:00.000000000'
 '2024-08-07T00:00:00.000000000' '2024-05-29T00:00:00.000000000'
 '2024-09-26T00:00:00.000000000' '2024-09-19T00:00:00.000000000'
 '2024-07-17T00:00:00.000000000' '2024-08-15T00:00:00.000000000'
 '2024-09-12T00:00:00.000000000' '2024-07-19T00:00:00.000000000'
 '2024-05-04T00:00:00.000000000' '2024-07-12T00:00:00.000000000'
 '2024-09-09T00:00:00.000000000' '2024-08-29T00:00:00.000000000'
 '2024-09-06T00:00:00.000000000' '2024-09-30T00:00:00.000000000'
 '2024-05-02T00:00:00.000000000' '2024-07-31T00:00:00.000000000'
 '2024-05-21T00:00:00.000000000' '2024-05-24T00:00:00.000000000'
 '2024-06-04T00:00:00.000000000' '2024-06-18T00:00:00.000000000'
 '2024-10-14T00:00:00.000000000' '2024-10-15T00:00:00.000000000']
--------------------------------------------------------------------------------
Number of unique values in BornDateTime columns are:
30725
['2024-05-30T12:52:10.000000000' '2024-05-10T16:07:47.000000000'
 '2024-08-24T19:42:04.000000000' ... '2024-07-16T20:49:27.000000000'
 '2024-09-03T06:44:12.000000000' '2024-05-29T00:29:14.000000000']
--------------------------------------------------------------------------------
Number of unique values in Project columns are:
4
['Springs' 'Spectra' 'Spire' 'Loft']
--------------------------------------------------------------------------------
Number of unique values in Stage columns are:
5
['Not Interested' 'Sales Closure' 'Pre Site Visit' 'Post Site Visit'
 'Flat Blocked']
--------------------------------------------------------------------------------
Number of unique values in CountOfClickEvents columns are:
178
[  15    5   18    6   49   11   19    4    2   36   14   24  370  226
   45   40   27   10    7   12   16    8    9   13 1452   74   57   28
   50   35   17   32  103    1   62   67   63   51   31   55   52   23
   46   29  155  762  282 1430   33   44  102   21   42   61  219   68
   54  166    0  108   60   37   25   22  111   20   53   34   95  120
   39  106   38    3   26  273   30   90  740   41   64   79   78  107
   48   76   56   47  113   81  145  109   93   87   96  105   94  104
```

```
   70  333  355  127   59  186  305   80  134  139   43   66   82   71
   65 1190   58   72  279   97   86   69   89   99   83   85   73  148
  168   91   84  236  152  122  114  129  117  124  162   88  549  182
  123  128  165  118  144  278  101  121  110   75  327   77  140  163
  275  100  372  136  173  115  125   92  231  154  261  150  119  287
  197  143  116  133  137  250  131  181  142  774]
--------------------------------------------------------------------------------
Number of unique values in WebTimeSpent (seconds) columns are:
29913
[501.779  54.175 292.363 ... 219.587 113.743 724.019]
--------------------------------------------------------------------------------
Number of unique values in UTM-Source columns are:
33
['Google' 'GDN' 'GMB' 'Sakshi' 'Yoptima' 'FIM' 'ASBL' 'Eenadu' 'Organic'
 'Inshorts' 'PD' 'WhatsApp' 'Google_Organic' 'Direct' 'Adonmo' 'zoom'
 'Youtube' 'Mygate' 'Facebook' 'LinkedIn' 'Whatsapp' 'google' 'Blog' 'TOI'
 'GoogleAds' 'ig' 'fb' 'whatsapp' 'IG' 'Newsprint' 'tma' 'Others' 'sakshi']
--------------------------------------------------------------------------------
Number of unique values in InstanceNumber columns are:
47
[  6   2  17   4   3   5  13  11  10   7 745   8   9  98 180  15  54 520
  14  19  31  58  12   1  16 211  40  52  24  22  27  30  48  23  18  29
  21  45  20  26  28  37  35  92  51  50  75]
--------------------------------------------------------------------------------
Number of unique values in HOME_TimeSpent columns are:
24189
[ 262.206   54.175  136.144 ... 2718.068   19.277   93.387]
--------------------------------------------------------------------------------
Number of unique values in LOCATION_TimeSpent columns are:
5405
[88.917  0.     5.676 ... 40.991 55.099 13.449]
--------------------------------------------------------------------------------
Number of unique values in MEDIA_TimeSpent columns are:
855
[0.000000e+00 9.797600e+01 8.952700e+01 6.374000e+02 2.879403e+03
 3.704300e+02 2.387700e+01 3.511810e+02 1.276300e+01 5.406500e+01
 1.530760e+02 5.885000e+00 7.037710e+02 3.047354e+03 1.546330e+02
 4.530068e+03 3.780500e+01 1.354530e+02 1.312450e+02 1.802957e+03
 6.537600e+02 3.662870e+02 6.856900e+01 1.234610e+02 1.379390e+02
 1.616980e+02 3.377000e+00 3.374400e+01 9.826100e+01 2.043810e+02
 8.513800e+01 3.976000e+01 2.118490e+03 2.218630e+02 1.579170e+02
 7.240900e+01 7.519000e+00 1.140900e+01 4.787280e+02 2.268710e+02
 1.477120e+02 9.581000e+01 5.913700e+01 8.043700e+01 2.896600e+01
 2.228100e+01 3.155120e+02 3.032000e+01 1.797085e+03 5.083490e+02
 1.092600e+01 2.536070e+02 2.071160e+02 2.428200e+01 8.544000e+00
 2.348580e+02 2.679200e+02 6.922000e+00 2.896000e+01 1.136800e+02
 2.159460e+02 5.604830e+02 4.318000e+00 5.322800e+01 7.710700e+01
 1.897000e+00 1.233330e+02 8.624000e+00 9.367300e+01 3.081230e+02
 1.393710e+02 6.751100e+01 7.079400e+01 8.586000e+00 2.037000e+01
 1.297500e+01 4.273100e+01 1.958830e+02 1.473300e+01 1.691400e+01
 1.457800e+01 2.598300e+01 1.088910e+02 3.878500e+01 7.494200e+01
 2.960700e+01 4.336800e+01 4.035000e+00 1.416870e+02 3.000330e+02
 1.625080e+02 1.480200e+01 4.810920e+02 4.103700e+01 1.239030e+02
 2.281000e+00 9.907800e+01 4.881400e+01 6.768000e+00 2.922000e+01
 4.180000e+01 1.158800e+01 1.118380e+02 2.337300e+01 8.099500e+01
 1.339460e+02 5.133000e+00 9.460600e+01 1.164100e+01 1.920400e+01
 2.594100e+01 7.149300e+01 1.502590e+02 1.655960e+02 2.675760e+02
 1.187200e+01 1.711900e+01 1.030200e+01 1.185300e+01 1.070860e+02
 1.504100e+01 3.252000e+01 1.568410e+02 3.281880e+02 6.725700e+01
 5.309900e+01 4.599300e+01 2.393290e+02 1.952000e+00 1.712360e+02
 9.493100e+01 1.741900e+01 9.899000e+00 7.174800e+01 3.692820e+02
 1.970780e+02 2.389000e+00 7.620300e+01 1.010800e+01 2.701600e+01
 1.106630e+03 5.238000e+00 6.428200e+01 8.533000e+00 5.037900e+01
 1.503950e+02 5.613900e+01 2.202000e+00 1.080370e+02 4.143100e+02
```

3.995900e+01 9.804000e+00 3.544180e+02 5.483300e+01 1.195500e+01
1.908840e+02 8.454200e+01 8.530000e-01 7.601000e+00 8.680900e+01
2.243940e+02 4.064220e+02 3.328100e+01 1.749120e+02 1.235100e+01
7.437700e+01 2.501020e+02 1.820890e+02 5.758000e+00 2.312700e+01
1.102290e+02 1.798500e+01 1.380500e+01 4.897940e+02 3.124800e+01
9.400000e+00 1.343300e+01 1.097300e+01 1.026650e+02 3.568500e+01
4.932540e+02 3.432400e+01 2.875000e+00 6.737800e+01 2.271800e+01
1.277160e+02 1.459800e+02 2.580600e+01 1.350810e+02 2.455960e+02
4.398100e+01 9.501200e+01 4.344700e+01 5.329000e+00 8.826400e+01
4.506900e+01 8.291500e+01 2.953000e+00 1.529100e+01 5.394000e+00
9.145400e+01 2.082620e+02 8.446400e+01 3.453460e+02 3.892700e+01
7.421400e+01 5.193600e+01 1.445710e+02 1.043120e+02 9.266000e+00
1.501880e+02 4.835300e+01 7.185900e+01 4.583700e+01 8.907700e+01
7.091000e+00 5.698100e+01 8.989600e+01 7.360610e+02 5.439000e+00
8.050000e-01 2.518300e+01 3.881000e+00 9.412900e+01 4.531000e+01
2.386900e+01 6.152000e+00 2.349500e+01 4.470000e+00 9.899800e+01
1.795500e+01 1.550000e+01 1.924370e+02 5.131680e+02 2.148600e+01
6.289900e+01 1.512020e+02 3.091200e+01 2.230250e+02 7.458500e+01
1.046720e+02 7.490000e-01 4.440300e+01 6.194800e+01 1.746600e+01
8.294000e+00 6.680900e+01 2.061000e+00 1.629400e+01 3.196200e+01
3.993800e+01 1.286970e+02 9.445300e+01 1.349840e+02 3.711630e+02
1.823570e+02 1.044900e+01 2.170690e+02 5.130000e+00 4.410000e+00
1.556200e+01 4.907200e+01 4.006600e+01 4.801980e+02 2.376700e+01
1.673040e+02 2.417900e+01 9.223600e+01 6.794000e+00 1.078920e+02
8.224000e+00 1.310740e+02 1.345500e+01 1.222350e+02 1.607280e+02
1.855600e+01 7.132600e+01 1.987200e+01 1.245200e+01 1.690000e+00
6.565400e+01 1.808000e+01 1.344210e+02 1.283280e+02 3.302890e+02
6.510000e+01 3.465000e+00 2.842140e+02 3.531200e+01 1.436800e+01
1.101800e+01 2.303820e+02 2.934200e+01 7.689900e+01 2.738380e+02
2.660500e+01 1.990170e+02 6.232600e+01 1.996500e+01 2.153000e+01
1.316330e+02 3.010400e+01 6.602300e+01 3.918810e+02 6.694770e+02
5.667540e+02 1.864000e+00 5.080000e+01 6.734700e+01 8.571000e+00
1.194800e+01 1.088300e+01 4.452000e+01 4.408200e+01 7.565000e+00
1.501370e+02 8.296200e+01 4.414300e+01 8.421000e+00 2.061980e+02
7.127000e+00 4.055810e+02 2.892000e+00 4.523700e+01 1.784400e+01
8.320100e+01 6.930600e+01 3.462270e+02 1.191100e+01 7.855800e+01
2.396100e+01 7.200000e+00 2.084000e+00 7.250600e+01 6.175400e+01
9.663000e+01 1.542120e+02 2.002000e+01 1.597700e+02 4.961100e+01
5.677200e+01 4.294200e+01 1.647050e+02 2.243820e+02 1.442200e+01
1.715700e+02 2.942700e+01 1.389800e+02 3.814400e+01 6.013300e+01
3.119510e+02 2.422400e+01 3.415000e+00 3.201500e+01 1.200690e+02
7.294500e+01 9.265400e+01 3.340720e+02 3.507400e+01 1.056370e+02
7.464000e+00 2.293100e+01 1.271490e+02 3.875250e+02 2.233200e+01
1.094600e+02 7.285600e+01 1.579000e+01 6.517000e+00 1.378000e+02
8.688800e+01 1.346100e+01 2.672730e+02 1.040270e+02 5.003400e+01
1.164400e+01 1.769200e+01 1.571000e+02 5.618130e+02 1.835330e+02
1.538300e+01 8.611000e+00 1.163490e+02 2.401000e+01 1.548880e+02
1.450480e+02 2.252700e+01 2.552880e+02 5.543000e+00 6.777000e+00
5.414000e+01 1.201800e+01 6.009000e+00 2.832300e+01 1.992500e+02
4.179600e+01 7.266820e+02 1.157870e+02 6.286300e+01 4.418300e+01
1.131900e+02 9.219000e+00 4.591150e+02 5.319200e+01 1.780000e+01
1.687000e+00 3.917000e+01 3.507800e+01 3.819400e+01 1.592920e+02
3.692000e+00 2.729000e+01 7.409000e+01 8.638000e+00 1.595400e+01
1.840030e+02 2.777000e+01 2.807970e+02 6.361900e+01 2.806900e+01
4.119100e+01 7.714000e+00 8.845000e+00 4.554200e+01 5.509070e+02
1.445400e+01 1.044660e+02 1.383120e+02 5.805000e+00 2.241800e+01
6.677300e+01 2.286700e+01 1.141800e+01 1.555650e+02 1.425030e+02
2.080520e+02 7.694200e+01 8.298000e+00 7.534100e+01 7.176500e+01
6.416100e+01 1.021300e+01 4.267000e+00 4.506000e+01 5.501000e+00
4.317200e+01 1.445000e+01 9.391500e+01 5.663900e+01 4.683300e+01
2.247000e+00 2.218700e+01 1.785900e+01 6.985000e+00 1.746140e+02
8.324300e+01 8.141000e+00 1.562170e+02 1.186900e+01 3.767000e+00
5.065000e+01 1.838850e+02 2.575700e+01 6.478600e+01 2.029000e+01
1.814500e+02 1.737600e+01 4.316700e+01 1.186600e+01 2.892900e+01

3.466200e+02 1.066520e+02 3.288100e+01 1.419600e+01 2.883100e+01
5.404000e+00 7.425600e+01 2.067400e+01 4.173590e+02 2.999800e+01
7.760900e+01 6.465000e+00 7.440000e+00 7.210000e+00 2.081100e+01
2.173560e+02 1.829110e+02 2.626100e+02 4.178030e+02 5.182700e+01
4.672000e+00 9.558000e+00 7.852300e+01 6.630000e+00 1.023560e+02
6.644900e+01 3.604700e+01 2.951000e+00 1.312200e+01 1.070900e+01
1.207300e+01 2.853400e+01 6.718000e+00 1.527500e+01 6.053000e+00
1.524820e+02 4.861800e+01 2.529100e+01 1.639890e+02 1.625290e+02
7.819100e+01 1.221010e+02 7.155400e+01 1.256430e+02 1.403600e+01
1.382290e+02 4.593200e+01 3.720300e+01 1.724600e+01 3.786000e+00
1.256750e+02 7.241400e+01 2.609500e+01 9.271900e+01 1.396310e+02
1.479900e+01 2.817800e+01 5.183000e+00 1.373250e+02 2.667680e+02
7.630500e+01 4.611200e+01 1.331000e+00 1.649600e+01 1.219700e+01
4.229300e+01 1.860680e+02 1.884800e+01 5.243990e+02 8.105200e+01
3.422280e+02 2.456000e+01 5.903600e+01 1.345200e+01 1.129000e+00
1.196880e+02 5.733700e+01 2.221100e+01 8.527000e+00 1.102100e+01
5.762640e+02 7.619200e+01 3.294400e+01 2.661700e+01 5.578000e+00
5.313000e+00 2.708000e+00 1.127700e+01 2.215890e+02 7.652000e+00
1.729900e+01 2.917800e+01 1.301800e+02 3.917000e+00 1.296100e+01
9.897000e+00 3.834000e+00 6.518000e+00 2.356360e+02 8.938000e+00
9.069100e+01 5.276700e+01 3.111120e+02 2.070320e+02 9.723400e+01
2.723900e+01 5.810400e+01 1.889500e+01 1.239100e+01 4.348600e+01
1.525100e+01 3.420300e+01 1.821600e+01 4.923000e+00 7.475600e+01
1.772500e+01 1.673000e+00 2.353500e+01 1.787000e+00 4.368700e+01
4.398000e+00 1.409860e+02 4.915200e+01 7.038700e+01 3.708000e+00
1.668700e+02 4.812600e+01 1.047970e+02 1.139150e+03 1.874560e+02
5.934600e+01 1.680260e+02 7.752600e+01 2.075800e+01 1.665230e+02
1.445100e+01 3.368830e+02 3.268900e+01 3.789320e+02 7.002500e+01
1.232300e+01 9.930800e+01 7.697800e+01 3.235100e+01 1.555150e+02
7.795890e+02 5.745000e+00 6.582300e+01 2.035300e+01 1.092250e+02
2.531690e+02 1.371260e+02 1.057400e+02 3.559500e+01 7.969100e+01
3.565500e+01 1.071530e+02 5.097600e+01 4.073900e+01 3.426240e+02
7.287000e+00 2.545400e+01 2.664400e+01 6.974640e+02 3.139390e+02
1.175000e+01 6.069000e+00 1.115400e+01 2.249740e+02 2.358900e+01
4.332200e+01 1.143300e+01 3.387000e+00 2.721680e+02 6.157000e+00
9.814800e+01 2.638470e+02 3.662200e+01 2.693400e+01 5.660000e+00
5.068000e+00 2.563470e+02 3.786800e+01 4.366100e+01 6.321300e+01
7.725000e+00 3.801000e+00 3.190270e+02 2.172600e+01 9.671700e+01
2.884840e+02 8.068200e+01 1.959100e+01 1.748000e+00 4.495590e+02
1.003840e+02 4.512800e+01 1.252650e+02 6.980000e-01 3.441600e+01
2.430520e+02 3.707600e+01 4.840560e+02 7.828000e+00 7.552000e+00
1.512100e+02 6.825500e+01 6.705000e+00 9.907000e+00 9.446000e+00
1.625490e+02 9.466000e+00 9.346800e+01 5.455800e+01 3.817490e+02
2.140600e+01 1.823600e+01 2.972240e+02 1.773000e+01 1.770850e+02
1.615100e+01 3.760240e+02 6.499100e+01 1.094950e+02 3.770000e+00
1.716000e+01 5.492600e+01 3.676800e+02 1.177340e+02 3.151000e+00
3.928000e+00 1.512500e+01 2.767900e+01 9.614400e+01 7.363600e+01
2.427000e+00 6.919200e+01 6.573500e+01 1.362000e+01 4.646100e+01
3.383000e+00 9.556000e+00 6.484000e+00 1.283500e+01 9.116000e+01
2.054030e+02 5.966800e+01 6.770300e+01 5.281800e+01 4.375000e+00
4.358300e+01 4.948000e+00 2.377700e+01 1.465050e+02 2.879000e+01
2.421500e+01 3.971700e+01 4.650000e+00 2.011570e+02 3.487000e+01
3.297530e+02 8.371000e+00 1.515630e+02 2.689940e+02 7.516000e+00
8.515300e+01 1.186070e+02 8.009900e+01 9.617300e+01 2.793000e+00
8.093000e+00 1.453600e+01 2.054100e+01 1.918510e+02 3.416070e+02
4.423000e+00 3.181300e+02 9.079200e+01 8.215500e+01 8.251710e+02
3.635250e+02 2.120200e+01 2.316800e+01 7.706000e+01 1.121930e+02
2.938600e+01 2.958970e+02 1.675940e+02 4.767000e+00 3.705500e+01
3.862700e+01 1.951200e+01 1.499500e+01 3.164600e+01 1.338000e+01
4.699300e+01 8.095900e+01 1.553900e+01 3.532300e+01 8.844600e+01
7.238470e+02 8.475700e+01 1.074280e+02 1.431400e+01 1.632750e+02
8.078030e+02 8.680000e+00 1.293030e+02 4.269300e+01 5.479020e+02
3.309100e+01 3.603880e+02 4.188600e+01 8.560000e+00 5.352400e+01
8.607900e+01 6.445730e+02 4.563000e+00 1.358200e+01 2.174900e+01

```
 1.864100e+01 9.075000e+00 8.390700e+01 7.405000e+00 1.139400e+01
 1.624270e+02 1.113320e+02 3.243000e+00 1.908600e+02 1.440500e+02
 4.844000e+00 3.388620e+02 1.694150e+02 7.321900e+01 8.636400e+01
 5.415900e+01 6.301000e+01 1.012360e+02 6.606700e+01 4.499000e+01
 5.837800e+01 1.995220e+02 6.718600e+01 1.805500e+02 4.706900e+01
 5.543600e+01 4.084600e+01 1.539890e+02 4.892700e+01 2.276660e+02
 4.152000e+00 1.090130e+02 5.178000e+00 5.263000e+00 1.787740e+02
 1.337680e+02 1.054370e+02 7.960200e+01 5.478700e+01 6.460900e+01
 8.423500e+01 3.049950e+02 1.141900e+02 7.165400e+01 9.584900e+01
 1.886830e+02 1.053900e+01 2.031000e+00 3.914000e+00 5.659500e+01
 4.293200e+01 8.594000e+00 2.755220e+02 6.721000e+00 7.863100e+01
 1.048330e+02 2.136300e+01 2.017200e+01 1.103400e+01 1.451180e+02
 1.897700e+01 1.438000e+01 3.218000e+01 3.659750e+02 3.526700e+01]
--------------------------------------------------------------------------------
Number of unique values in PLAN_TimeSpent columns are:
6482
[  13.349    0.     275.002 ...  269.145 2561.207  209.408]
--------------------------------------------------------------------------------
Number of unique values in PRICE_TimeSpent columns are:
21294
[137.307   0.     58.243 ... 187.375  81.017 411.814]
--------------------------------------------------------------------------------
Number of unique values in SPECIFICATIONS_TimeSpent columns are:
1536
[  0.      4.847 219.774 ... 183.19  645.55    9.41 ]
--------------------------------------------------------------------------------
Number of unique values in AMENITIES_TimeSpent columns are:
738
[0.000000e+00 1.267640e+02 1.151631e+03 1.213350e+02 8.579080e+02
 1.942323e+03 1.371330e+02 1.114390e+02 1.195916e+03 2.368000e+00
 5.863400e+01 7.842100e+01 2.014840e+02 6.159900e+01 2.120550e+02
 3.088000e+00 2.830945e+03 6.673200e+01 1.542200e+01 5.577400e+01
 5.584860e+02 5.860500e+02 8.126000e+00 4.352600e+01 4.413500e+01
 6.234500e+01 3.772000e+00 5.397100e+01 1.881300e+01 1.199510e+02
 1.762400e+01 8.404100e+01 1.186200e+02 1.171600e+01 4.509000e+00
 7.223700e+01 1.150000e+00 1.121250e+02 7.900000e-01 5.931800e+01
 7.390000e+01 1.923900e+01 2.884100e+01 4.606000e+00 6.935700e+01
 2.784600e+01 1.131520e+02 8.297000e+01 1.824000e+00 5.915300e+01
 1.238900e+01 1.047200e+01 9.560000e-01 3.005600e+01 2.918500e+01
 9.760700e+01 3.771300e+01 2.636060e+02 2.014370e+02 6.054000e+02
 4.067900e+01 5.579800e+01 1.144770e+02 1.559340e+02 6.654100e+01
 7.998600e+01 3.676200e+01 2.532800e+01 6.066000e+01 1.257000e+00
 2.268500e+01 1.542530e+02 5.785200e+01 7.387000e+00 2.205300e+01
 2.282000e+00 3.846400e+01 1.948200e+01 2.418000e+00 2.870500e+01
 2.371900e+01 5.829700e+01 7.731000e+00 5.265000e+00 6.551800e+01
 8.217800e+01 7.090000e-01 5.361000e+00 2.700100e+01 2.441600e+01
 5.841800e+01 3.499000e+00 3.409800e+01 5.191000e+01 7.656600e+01
 8.917300e+01 4.180700e+01 2.376800e+01 4.197200e+01 1.999940e+02
 2.600700e+01 1.003190e+02 7.515400e+01 1.225590e+02 2.479600e+01
 5.251900e+01 6.377000e+00 6.201000e+00 1.494000e+01 2.085000e+00
 2.384500e+01 4.378700e+01 1.557100e+01 2.638900e+01 3.117200e+01
 8.652900e+01 6.065000e+00 7.610600e+01 7.764800e+01 1.263000e+01
 4.419100e+01 2.602800e+01 3.678500e+01 1.009870e+02 1.825200e+01
 4.628200e+01 6.994300e+01 5.502600e+01 3.784300e+01 1.869700e+01
 1.049700e+01 5.375200e+01 5.456300e+01 7.696000e+00 2.525800e+01
 3.496400e+01 1.004390e+02 4.272000e+01 7.462400e+01 2.023330e+02
 1.619700e+02 6.100300e+01 4.947000e+00 1.659800e+01 1.178320e+02
 1.715910e+02 2.781010e+02 2.958700e+01 2.486800e+01 6.839000e+00
 1.658700e+01 1.405700e+01 8.487300e+01 3.399800e+01 5.180000e-01
 2.338780e+02 5.131100e+01 6.573000e+00 1.207530e+02 2.847000e+01
 5.892200e+01 7.599200e+01 5.248100e+01 6.496700e+01 7.104000e+01
 4.611800e+01 5.200500e+01 2.476000e+01 1.821000e+00 1.182600e+01
 2.295200e+01 9.645000e+00 4.155320e+02 1.173180e+02 8.680000e-01
 1.827840e+02 1.917300e+01 1.414600e+01 1.265600e+01 3.989400e+01
```

1.240000e+01 5.053100e+01 8.484200e+01 1.006740e+02 2.901600e+01
4.972400e+01 1.112900e+01 3.084400e+01 1.615400e+01 5.897700e+01
2.084000e+00 1.610400e+01 6.637500e+01 7.318200e+01 1.514850e+02
9.719000e+00 3.580000e+00 2.644900e+01 8.865700e+01 2.601420e+02
2.885900e+01 2.155100e+01 1.097400e+01 1.277850e+02 1.463460e+02
5.700200e+01 4.123700e+01 7.616600e+01 2.501000e+00 1.508500e+01
8.500000e+01 7.345900e+01 3.337500e+01 4.102800e+01 2.618200e+01
2.578657e+03 6.988400e+01 6.614100e+01 2.134000e+00 1.193090e+02
5.060000e+01 1.699600e+01 4.890300e+01 2.352100e+01 2.508600e+01
6.691000e+00 6.529800e+01 1.505600e+02 3.032800e+01 3.940700e+01
1.320400e+01 2.897400e+01 7.556000e+00 3.989500e+01 4.935210e+02
1.950000e+00 3.106600e+01 1.479100e+01 1.210800e+01 2.206700e+01
5.277000e+00 4.837270e+02 3.456700e+01 2.230400e+01 3.016100e+01
1.530400e+01 3.101700e+01 8.480200e+01 9.446000e+00 2.374940e+02
1.297000e+00 1.710190e+02 4.538900e+01 9.503200e+01 9.587000e+00
2.851400e+01 3.719030e+02 8.470700e+01 1.241000e+01 1.691700e+01
5.579700e+01 6.234000e+00 1.953080e+02 6.650000e-01 1.060558e+03
5.170900e+01 2.906930e+02 6.200000e-01 5.667300e+01 1.689900e+01
5.070400e+01 8.997700e+01 1.569120e+02 3.680900e+01 5.797800e+01
6.478300e+01 4.784100e+01 3.536180e+02 5.094600e+01 5.762700e+01
3.930800e+01 3.577100e+01 7.052800e+01 3.773800e+01 4.056300e+01
2.426700e+01 4.314200e+01 4.502900e+01 1.075910e+02 1.926000e+00
4.428300e+01 1.924800e+01 8.946400e+01 1.820900e+01 4.478100e+01
7.178200e+01 1.041210e+02 5.810800e+01 2.277000e+00 1.066500e+02
2.324130e+02 6.545000e+00 8.404700e+01 3.612900e+01 8.357000e+01
4.968800e+01 2.563000e+00 2.080530e+02 1.666400e+01 8.987000e+00
4.621000e+00 1.847900e+02 6.241800e+01 2.200930e+02 2.262500e+01
1.425300e+01 9.816300e+01 2.183900e+01 2.095900e+01 3.246000e+01
1.053040e+02 4.144900e+01 1.370630e+02 5.228900e+01 7.535100e+01
3.822000e+00 1.040490e+02 2.984100e+01 3.681200e+01 7.010000e-01
3.088600e+01 8.031800e+01 4.391500e+01 2.940100e+01 1.063000e+01
1.890000e+00 1.369760e+02 1.942900e+01 8.173600e+01 1.848790e+02
6.175000e+01 7.258100e+01 2.164870e+02 2.998800e+01 3.618830e+02
2.347400e+01 2.102600e+01 3.207200e+01 6.242000e+00 6.008000e+00
3.714100e+01 8.162000e+00 8.786400e+01 2.221200e+01 8.906300e+01
2.337000e+00 1.430040e+02 7.144700e+01 5.475600e+01 3.734100e+01
4.833600e+01 2.340300e+01 1.903500e+01 4.044600e+01 2.261400e+01
2.247600e+01 1.112000e+01 3.119000e+01 5.422100e+01 5.350600e+01
9.559200e+01 6.120000e-01 2.334700e+01 1.064370e+02 2.602000e+00
7.011000e+01 5.797600e+01 2.159900e+01 6.399700e+01 1.778500e+01
1.453040e+02 6.830100e+01 4.973000e+01 5.623900e+01 3.019100e+01
2.535560e+02 6.720000e-01 5.118000e+00 1.468000e+01 2.396000e+01
4.911660e+02 6.927200e+01 8.785000e+00 4.460900e+01 9.432700e+01
1.161090e+02 1.197320e+02 4.710300e+01 1.081970e+02 1.123800e+01
1.214040e+02 1.093070e+02 6.474000e+00 8.220100e+01 1.643920e+02
4.941900e+01 8.253500e+01 3.146000e+00 5.299030e+02 4.162300e+01
2.256900e+01 1.777000e+01 4.019100e+01 1.507800e+01 1.159645e+03
1.214280e+02 9.930000e+00 6.072000e+01 2.400200e+01 1.126160e+02
1.913000e+01 3.590700e+01 7.323800e+01 8.894000e+00 8.059000e+01
6.347000e+00 6.624000e+00 4.754700e+01 3.655300e+01 7.708000e+00
6.863900e+01 2.599930e+02 3.554300e+01 4.938700e+01 5.008200e+01
6.181000e+01 3.068000e+01 3.692000e+00 2.854700e+01 1.830560e+02
7.154000e+00 4.141700e+01 4.759900e+01 8.926000e+00 7.124000e+00
1.416310e+02 1.073720e+02 4.645000e+00 5.104000e+00 2.691800e+01
6.193800e+01 4.078910e+02 5.095800e+01 7.356000e+00 2.208500e+01
5.210200e+01 9.909300e+01 2.140400e+01 3.177800e+01 1.636040e+02
7.579600e+01 1.316700e+01 3.335100e+01 7.355700e+01 5.440100e+01
1.372000e+00 1.190210e+02 2.115500e+01 3.806600e+01 1.112600e+01
1.621000e+01 1.706700e+01 1.388120e+02 7.506600e+01 1.436200e+01
4.671000e+00 2.857300e+01 3.605400e+01 5.154400e+01 2.085500e+01
1.843830e+02 2.052400e+01 3.056200e+01 2.574900e+01 2.870700e+01
2.599300e+01 1.276350e+02 5.667000e+00 3.022300e+01 5.550300e+01
7.959200e+01 3.658790e+02 8.236500e+01 8.681700e+01 3.094100e+01
3.337000e+00 1.343000e+00 1.040800e+01 2.049500e+02 1.254300e+01

```
 1.913700e+01 1.442580e+02 6.887100e+01 8.789000e+00 3.270200e+01
 8.879000e+00 1.193700e+01 5.874000e+01 1.426700e+01 8.781100e+01
 2.521100e+01 2.784700e+01 5.793800e+01 1.746900e+01 6.443000e+00
 5.518600e+01 5.114600e+01 1.267500e+01 6.887800e+01 7.363720e+02
 5.362000e+01 1.002200e+01 2.344100e+01 3.751100e+01 3.978400e+01
 3.194300e+01 6.237100e+01 5.755000e+01 1.047000e+01 5.489900e+01
 7.641700e+01 1.139700e+02 7.697000e+00 2.219000e+01 5.341000e+01
 6.872300e+01 2.264800e+01 1.337100e+01 2.264000e+00 1.223520e+02
 4.104300e+01 4.745000e+00 1.279700e+02 9.230900e+01 5.148900e+01
 1.615000e+00 1.520580e+02 4.268700e+01 3.904500e+01 8.122000e+01
 2.255580e+02 2.601900e+01 1.109900e+01 6.855000e+00 1.668000e+00
 6.362800e+01 9.623200e+01 1.194000e+00 2.599600e+01 4.480400e+01
 8.614300e+01 5.497800e+01 4.110000e+01 4.499700e+01 2.316000e+01
 1.153910e+02 4.252700e+01 7.265900e+01 1.680100e+01 1.182500e+01
 7.824800e+01 5.490800e+01 7.233800e+01 2.069900e+01 3.433500e+01
 4.522100e+01 4.595100e+01 1.098880e+02 5.706600e+01 4.880800e+01
 3.891500e+01 4.071600e+01 3.292000e+00 3.296100e+01 1.085700e+01
 1.454000e+00 5.482000e+00 1.608893e+03 6.342000e+01 6.405700e+01
 9.552000e+00 8.853200e+01 1.072800e+01 6.189200e+01 1.459540e+02
 1.495000e+01 7.624700e+01 8.522000e+01 2.007100e+01 6.938200e+01
 2.998000e+00 3.285900e+01 1.274890e+02 7.073000e+00 2.188200e+01
 2.122500e+01 7.235400e+01 2.375700e+01 5.306100e+01 2.771000e+01
 4.384000e+00 1.986700e+01 8.288800e+01 9.563200e+01 1.560050e+02
 7.236800e+01 4.160300e+01 6.976580e+02 8.630600e+01 1.078250e+02
 1.047550e+02 4.174900e+01 1.413300e+01 4.561000e+00 2.701000e+00
 9.965600e+01 5.807600e+01 7.394000e+00 6.924000e+00 4.265000e+00
 1.615700e+01 1.293000e+01 7.245000e+00 1.424680e+02 1.057600e+01
 1.277000e+01 1.300400e+02 6.361500e+01 5.326800e+01 1.151700e+01
 2.650000e+01 1.383840e+02 5.673100e+01 6.737800e+01 4.466000e+00
 2.669460e+02 2.502100e+01 1.887800e+01 3.847200e+01 1.732400e+01
 1.442870e+02 6.188200e+01 3.294700e+01 9.017700e+01 1.434000e+00
 2.615000e+00 2.361500e+01 6.524200e+01 3.894000e+00 2.654200e+01
 2.056060e+02 1.767100e+01 1.329600e+01 2.362600e+01 1.482390e+02
 3.119000e+00 1.818600e+01 4.642300e+01 9.627000e+00 7.887800e+01
 4.345000e+00 6.475400e+01 2.277400e+01 5.982900e+01 4.918200e+01
 5.070000e+00 1.002300e+01 2.213230e+02 3.381600e+01 6.324900e+01
 1.807760e+02 9.623000e+00 3.996000e+00 9.230100e+01 5.339100e+01
 2.799900e+01 1.188490e+02 2.975000e+00 2.476400e+01 5.235200e+01
 9.103200e+01 4.544500e+01 2.836800e+01 1.434410e+02 4.046000e+00
 7.593500e+01 5.460900e+01 1.188290e+02 2.814800e+01 3.723400e+01
 2.277600e+01 1.092300e+02 2.633900e+01 2.255200e+01 4.050100e+01
 4.429300e+01 1.795900e+01 4.762360e+02 1.938000e+01 2.179300e+01
 3.925800e+01 6.190900e+01 4.915000e+00 3.811200e+01 2.099770e+02
 4.779000e+00 1.136300e+01 1.711900e+01 3.573000e+00 2.001080e+02
 4.623000e+00 8.749500e+01 7.943200e+01 7.557000e+00 9.817000e+00
 1.966500e+01 6.706800e+01 3.048000e+01 7.600500e+01 1.860000e+00
 1.239700e+01 1.395300e+02 2.019660e+02 8.915800e+01 1.729100e+01
 1.886780e+02 8.926000e+00 8.655000e+00]
--------------------------------------------------------------------------------
Number of unique values in MediaTime columns are:
1936
[   0.     10.05   22.36 ... 1868.7    20.     41.85]
--------------------------------------------------------------------------------
Number of unique values in OperatingSystem columns are:
6
['Windows' 'Android' 'Linux' 'Mac' 'iOS' 'Chrome-OS']
--------------------------------------------------------------------------------
Number of unique values in Country columns are:
73
['India' 'United Arab Emirates' 'United States' 'United Kingdom'
 'South Korea' 'Singapore' 'Belgium' 'Germany' 'Saudi Arabia' 'Nigeria'
 'Qatar' 'Ireland' 'None' 'Ghana' 'Australia' 'Indonesia' 'Canada'
 'Sweden' 'Benin' 'Malaysia' 'Norway' 'Switzerland' 'Bangladesh' 'Denmark'
 'Tanzania' 'Bahrain' 'Netherlands' 'Mozambique' 'New Zealand' 'Uganda'
```

```
 'Cambodia' 'Philippines' 'Kuwait' 'Botswana' 'Taiwan' 'France' 'Thailand'
 'Kenya' 'Poland' 'Vietnam' 'Bhutan' 'Japan' 'Oman' 'Jamaica' 'Uzbekistan'
 'Luxembourg' 'Mexico' 'Democratic Republic of the Congo' 'Pakistan'
 'China' 'Hong Kong' 'Zambia' 'Russia' 'Trinidad and Tobago' 'Israel'
 'Aruba' 'Algeria' 'Nepal' 'Finland' 'Malta' 'South Africa' 'Iraq'
 'Austria' 'Argentina' 'Ethiopia' 'Mauritius' 'Egypt' 'Italy' 'Czechia'
 'Guyana' 'Spain' 'Maldives' 'Portugal']
--------------------------------------------------------------------------------
Number of unique values in State columns are:
287
['Maharashtra' 'Telangana' 'دبي' 'Chhattisgarh' 'Texas' 'England'
 'Haryana' 'Seoul' 'Delhi' 'Andhra Pradesh' 'Missouri' 'North Carolina'
 'Karnataka' 'None' 'Brussel' 'Jharkhand' 'Tamil Nadu' 'Kerala' 'Nevada'
 'Odisha' 'California' 'Bihar' 'Uttar Pradesh' 'West Bengal' 'Virginia'
 'Punjab' 'Berlin' 'المنطقة الشرقية' 'Pennsylvania' 'Madhya Pradesh'
 'Eastern Province' 'Lagos' 'Delaware' 'Rangareddy' 'Ohio' 'Maryland'
 'Gujarat' 'Al Rayyan Municipality' 'Goa' 'County Dublin' 'Kentucky'
 'Riyadh Province' 'Manipur' 'Uttarakhand' 'Arkansas' 'Nagaland' 'Dubai'
 'Greater Accra Region' 'New South Wales' 'Daerah Khusus Ibukota Jakarta'
 'Bayern' 'महाराष्ट्र' 'Ontario' 'Tennessee' 'Stockholms län' 'Washington'
 'Abu Dhabi' 'Littoral' 'Québec' 'Victoria' 'Arizona' 'Colorado'
 'New Jersey' 'Selangor' 'Østfold' 'Rogaland' 'Assam' 'Zürich'
 'Dhaka Division' 'Illinois' 'Florida' 'Massachusetts' 'Dar es Salam'
 'Capital Governorate' 'Noord-Holland' 'Queensland' 'Maputo' 'Hamburg'
 'Kansas' 'Auckland' 'Doha Municipality' 'Flevoland' 'Puducherry'
 'New York' 'Central Region' 'Georgia' 'Al Wakrah Municipality'
 'Baden-Württemberg' 'Phnom Penh' 'Cavite' 'Khulna Division' 'حولي'
 'Michigan' 'County Carlow' 'Gaborone City' 'Taipei City' 'Vlaams Gewest'
 'Rajasthan' 'Al Sheehaniya Municipality' 'Wales' 'Chandigarh'
 'Nouvelle-Aquitaine' 'Hessen' 'British Columbia' 'Alberta'
 'Makkah Province' 'North Holland' 'Scotland' 'Indiana' 'Alabama'
 'Krung Thep Maha Nakhon' 'Zuid-Holland' 'Thüringen' 'Kiambu County' 'FL'
 'Wilayah Persekutuan Kuala Lumpur' 'Connecticut'
 'Województwo mazowieckie' 'Hà Nội' 'Western Australia' 'Thimphu' 'Tokyo'
 'Muscat Governorate' 'St. Ann Parish' 'District of Columbia'
 'Noord-Brabant' 'Sharjah' 'Minnesota' 'Województwo małopolskie' 'Sabah'
 'Tripura' 'Toshkent Shahri' 'County Donegal' 'Województwo łódzkie'
 'Esch-sur-Alzette' 'Jalisco' 'Skåne län' 'Kinshasa' 'Hsinchu County'
 'Sindh' 'Guang Dong Sheng' 'Chittagong Division' 'Rheinland-Pfalz'
 'العاصمة' 'Querétaro' 'Aseer Province' 'Himachal Pradesh'
 'New Territories' 'Hawalli Governorate' 'Uppsala län' 'Wellington'
 'Lusaka Province' 'إمارة الشارقة' 'Bruxelles' 'Stockholm County'
 'Moskva' 'San Juan-Laventille Regional Corporation' 'أبو ظبي' 'Kanagawa'
 'Kyoto' 'Île-de-France' 'Center District' 'New Hampshire' 'Calabarzon'
 'Al Farwaniyah Governorate' 'Västra Götalands län' 'Johor' 'Nebraska'
 'Dadra and Nagar Haveli and Daman and Diu' 'Utrecht' 'Oregon' 'Utah'
 'Nairobi County' 'Kakamega County' 'Copperbelt Province'
 'Al Asimah Governate' 'Oran Province' 'Meghalaya' 'Rhode Island'
 'Haut-Katanga' 'Kowloon' 'Northern Ireland' 'Nordrhein-Westfalen'
 'Bagmati Province' 'Ras al Khaimah' 'Northern Governorate' 'Batam Island'
 'Northwest Territories' 'County Cavan' 'Aargau' 'Kent County'
 'Al Batinah North Governorate' 'County Wicklow' 'Uusimaa' 'Wisconsin'
 'North Dakota' 'Western Cape' 'Southern Governorate' 'KwaZulu-Natal'
 'Sikkim' 'West Virginia' 'Arusha Region' 'Erbil Governorate'
 'بلدية الدوحه' 'Steiermark' 'South Carolina' 'Metro Manila' 'Ajman'
 'Ciudad Autónoma de Buenos Aires' 'Al Khor and Al Thakhira Municipality'
 'Tabuk Province' 'Luxembourg' 'Kilifi County' 'Addis Ababa' 'Free State'
 'Port Louis District' 'Grand Est' 'उत्तर प्रदेश' 'Nagano'
 'South Sinai Governorate' 'Östergötlands län' 'Hauts-de-France' 'Lazio'
 'Wien' 'Ash Sharqiyah South Governorate' 'South Australia' 'Nova Scotia'
 'Tirol' 'Saitama' 'Ras Al Khaimah' 'Andaman and Nicobar Islands'
 'Muḥāfaẓat al-ʿĀṣimah' 'Ad Dakhiliyah \u200dGovernorate' 'County Cork'
 'St. Andrew Parish' 'Sachsen-Anhalt' 'Nuevo León' 'بلدية الريان'
 'Hlavní město Praha' 'Al Madinah Province' 'محافظة العاصمة'
```

```
 'East Berbice-Corentyne' 'Al Qassim Province' 'Iowa'
 "Provence-Alpes-Côte d'Azur" 'Vaud' 'Comunidad de Madrid' 'County Clare'
 'South Dakota' 'Central District' 'Arunachal Pradesh'
 'Australian Capital Territory' 'Giza Governorate' 'Kilimanjaro Region'
 'Overijssel' 'Bavaria' 'Vermont' 'Taranaki' 'Malé' 'منطقة الرياض'
 'New Mexico' 'County Kildare' 'Chang Wat Chon Buri' 'County Tipperary'
 'Tochigi' 'Dong Nai' 'Tashkent' 'North Rhine-Westphalia' 'Sicilia'
 'Al Ahmadi Governorate' 'محافظة مسقط' 'Mizoram'
 'Northern Borders Province' 'Jazan Province' 'Melaka'
 'Dhofar Governorate' 'Kronobergs län' 'Chang Wat Chiang Mai'
 'Cairo Governorate' 'Plaines Wilhems District' 'Guanajuato' 'Brandenburg'
 'Lombardia' 'Provincia de Buenos Aires' 'Lisboa' 'New Brunswick' 'Agder']
--------------------------------------------------------------------------------
Number of unique values in City columns are:
2850
['Mumbai' 'Hyderabad' 'دبي' ... 'Talamanchi' 'Saroornagar'
 'Mopidevi Lanka']
--------------------------------------------------------------------------------
Number of unique values in MicroMarket columns are:
5591
['Andheri West' 'Habsiguda' 'زعبيل' ... 'Talamanchi' 'Bagh Swaniya'
 'Mopidevi Lanka']
--------------------------------------------------------------------------------
```

In [128…
```
# Interesting features:
# Project, stage, UTM-Source, OS, country
```

Preprocessing before EDA

Creating new columns 'CONVERTED'

We will use 'Stage' column to create the new column. 'Not interested' will take value 0 and rest of others will take value 1

In [128…
```
data['Converted'] = data['Stage'].apply(lambda x: 0 if x =="Not Interested" else 1)
```

In [128…
```
data[data['Converted']==1].shape
```

Out[1286]:
```
(178, 24)
```

Creating new columns 'Time of Day' from BornDateTime to find out trend

In [128…
```
data['TimeOfDay'] = data['BornDateTime'].dt.hour.apply(lambda x: 'Morning' if 6<=x<
```

Creating new column 'DayOfWeek'

In [128…
```
data['DayOfWeek'] = data['BornDateTime'].dt.dayofweek.apply(lambda x: x)
```

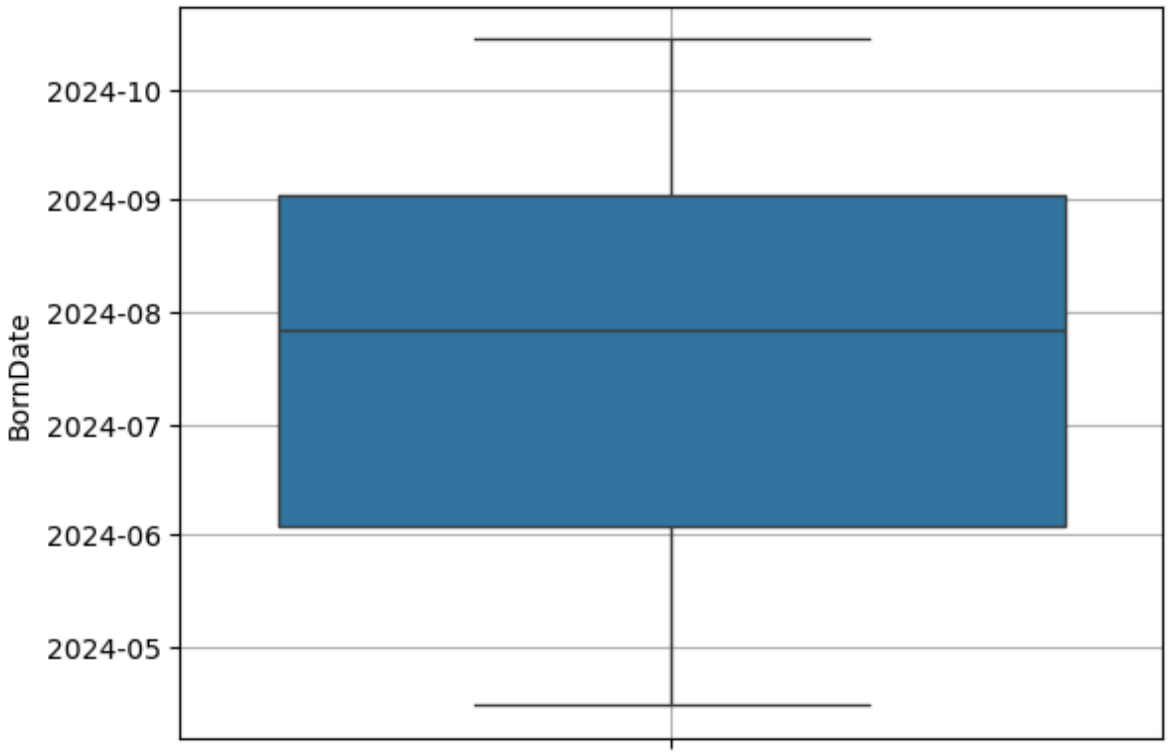
1. EDA

Borndate and TimeOfDay

In [128…
```
data['BornDate'].describe()
```

Out[1289]:
```
count                   30773
unique                    184
top       2024-08-11 00:00:00
freq                      378
first     2024-04-15 00:00:00
last      2024-10-15 00:00:00
Name: BornDate, dtype: object
```

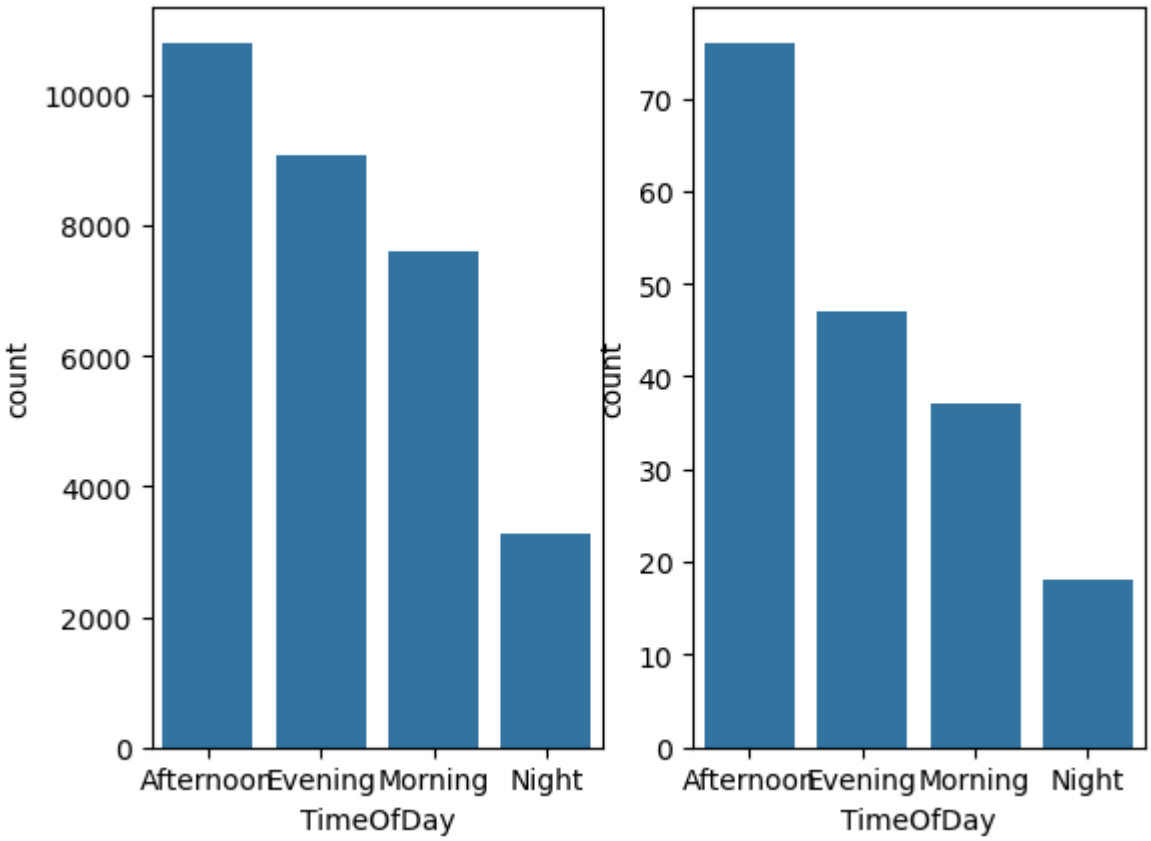
In [129…
```
sns.boxplot(data=data,y='BornDate')
plt.grid()
```



In [120…
```
data['TimeOfDay'].describe()
```

Out[1204]:
```
count         30773
unique            4
top       Afternoon
freq          10803
Name: TimeOfDay, dtype: object
```

In [129…
```
plt.subplot(1,2,1)
sns.countplot(data=data,x='TimeOfDay')
plt.subplot(1,2,2)
sns.countplot(data=data[data['Converted']==1],x='TimeOfDay')
plt.show()
```



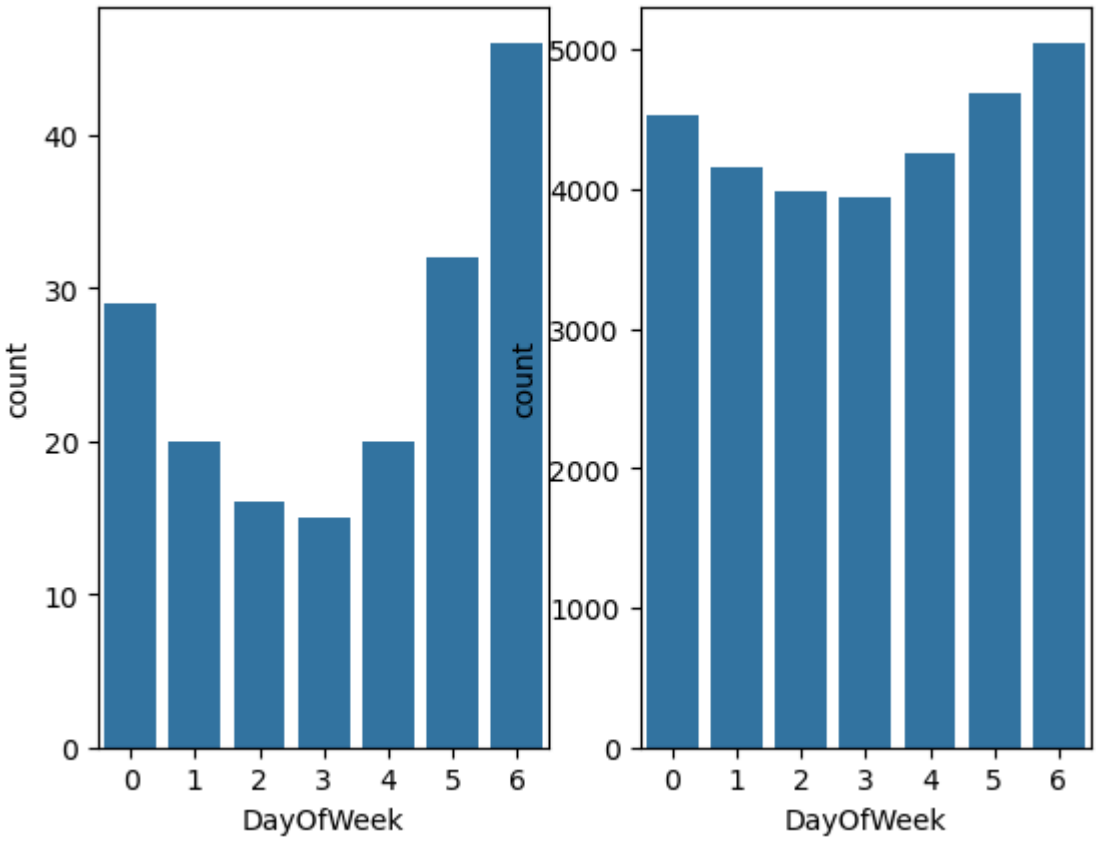
Converted leads TimeOfDay

Evening and Night time conversion rate seems less than Afternoon and Morning time

DayOfWeek: leads originating around weekend has better conversion rate.

In [129…

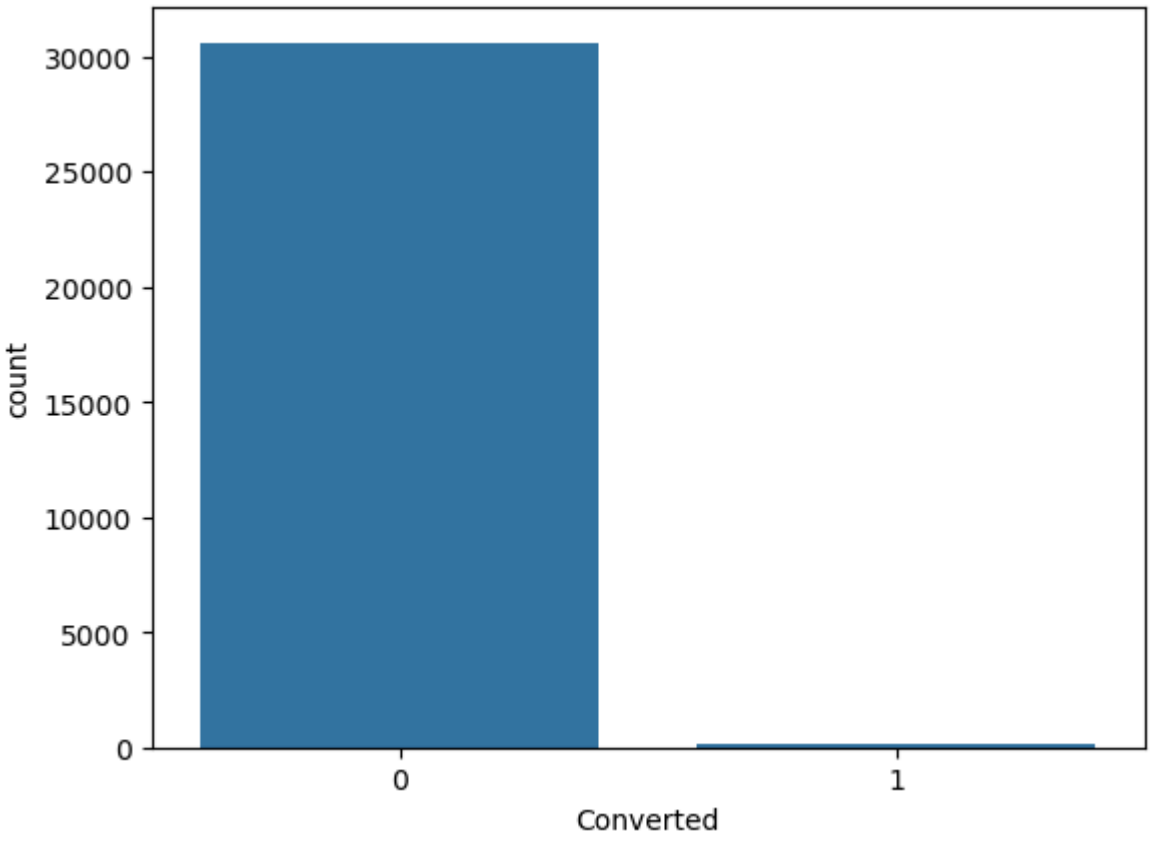
```
plt.subplot(1,2,1)
sns.countplot(data=data[data['Converted']==1],x='DayOfWeek')
plt.subplot(1,2,2)
sns.countplot(data=data[data['Converted']==0],x='DayOfWeek')
plt.show()
```



Converted and Project wise converted

In [120…

```
sns.countplot(data=data,x='Converted')
plt.show()
```



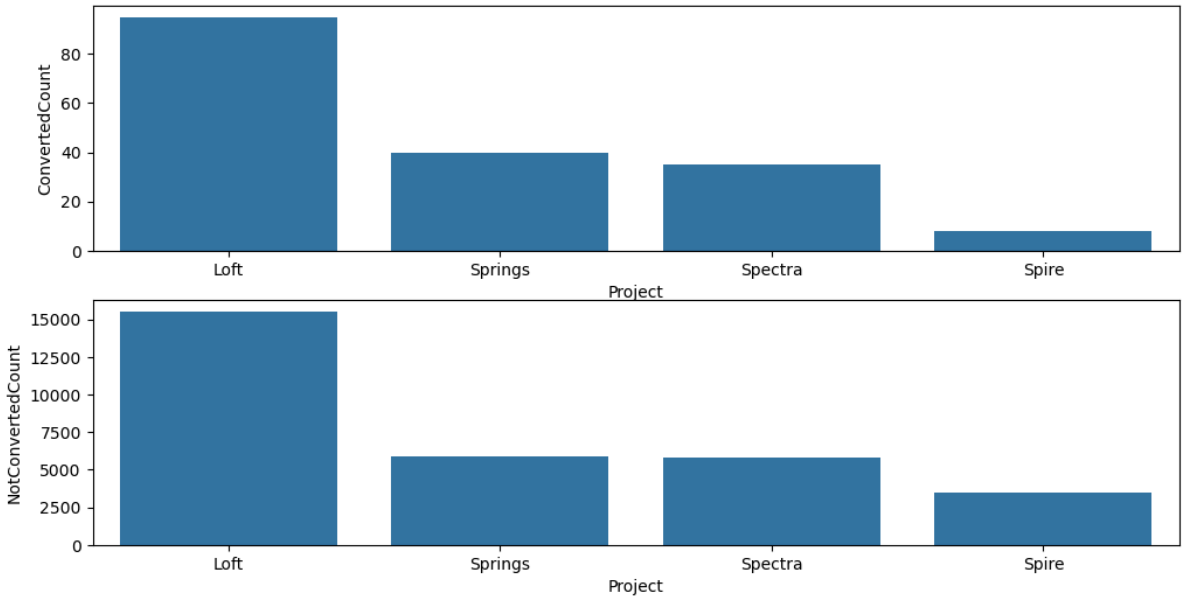
In [120…

```
data[data['Converted']==0].shape[0], data[data['Converted']==1].shape[0]
```

Out[1208]: (30595, 178)

only 178 leads got converted while 30595 did not.

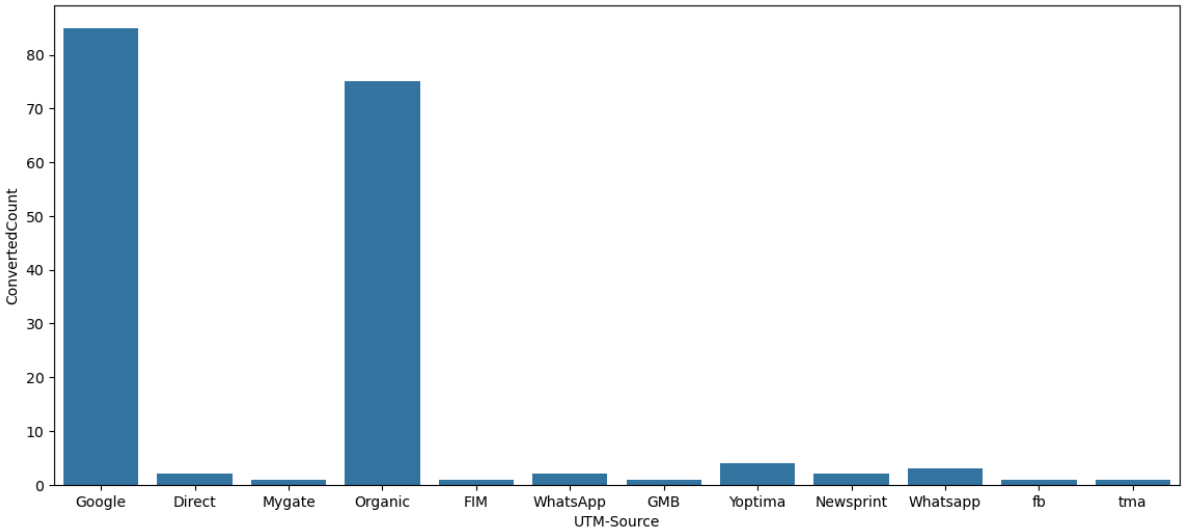
In [120…
```
plt.figure(figsize=(12,6))
plt.subplot(2,1,1)
sns.countplot(data=data[data['Converted']==1],x='Project',order = data['Project'].v
plt.ylabel('ConvertedCount')
plt.subplot(2,1,2)
sns.countplot(data=data[data['Converted']==0],x='Project',order = data['Project'].v
plt.ylabel('NotConvertedCount')
plt.show()
```



Project wise conversion looks propotional except for Spire

UTM-Source

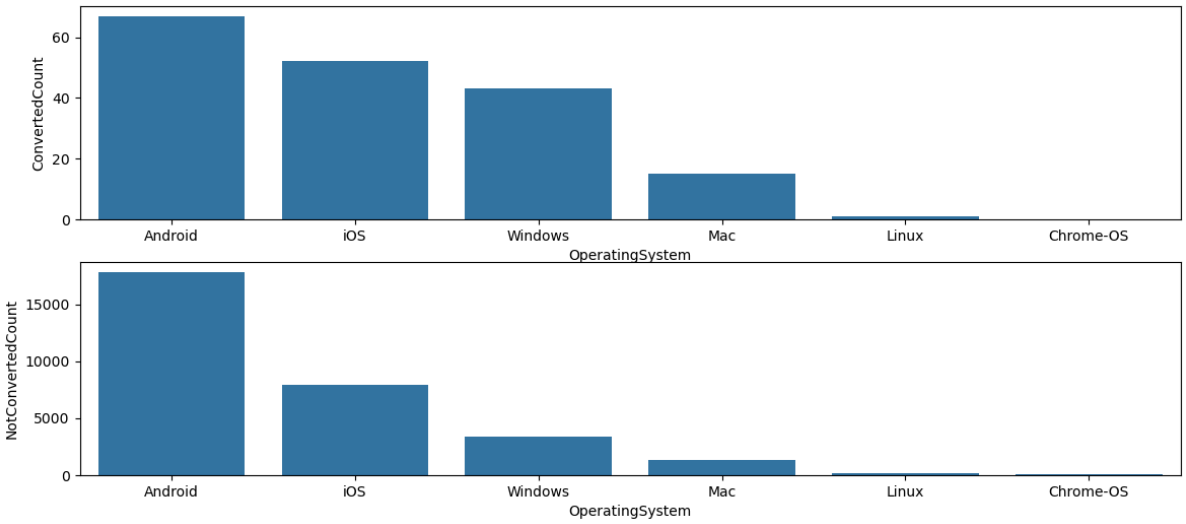
In [121…
```
plt.figure(figsize=(14,6))
sns.countplot(data=data[data['Converted']==1],x='UTM-Source')
plt.ylabel('ConvertedCount')
plt.show()
```



Operating System

In [121…

```
plt.figure(figsize=(14,6))
plt.subplot(2,1,1)
sns.countplot(data=data[data['Converted']==1],x='OperatingSystem', order = data['Op
plt.ylabel('ConvertedCount')

plt.subplot(2,1,2)
sns.countplot(data=data[data['Converted']==0],x='OperatingSystem', order = data['Op
plt.ylabel('NotConvertedCount')
plt.show()
```



Proportion of converted leads vary on basis of operating system.

In [129…

```
data.head()
```

Out[1293]:

	LeadId	VisitorId	BornDate	BornDateTime	Project	Stage	CountOfClickEvents	WebTime(sec
0	198611	Visitor-1003563	2024-05-30	2024-05-30 12:52:10	Springs	Not Interested	15	50
1	193927	Visitor-1006753	2024-05-10	2024-05-10 16:07:47	Spectra	Not Interested	5	5
2	230525	Visitor-1017271	2024-08-24	2024-08-24 19:42:04	Springs	Not Interested	18	29
3	208705	Visitor-1029567	2024-07-10	2024-07-10 12:36:38	Spectra	Not Interested	6	17
4	253755	Visitor-1044910	2024-09-22	2024-09-22 14:30:57	Spectra	Not Interested	49	97

5 rows × 26 columns

CountOfClickEvents

In [121…

```
data[data['Converted']==0]['CountOfClickEvents'].describe()
```

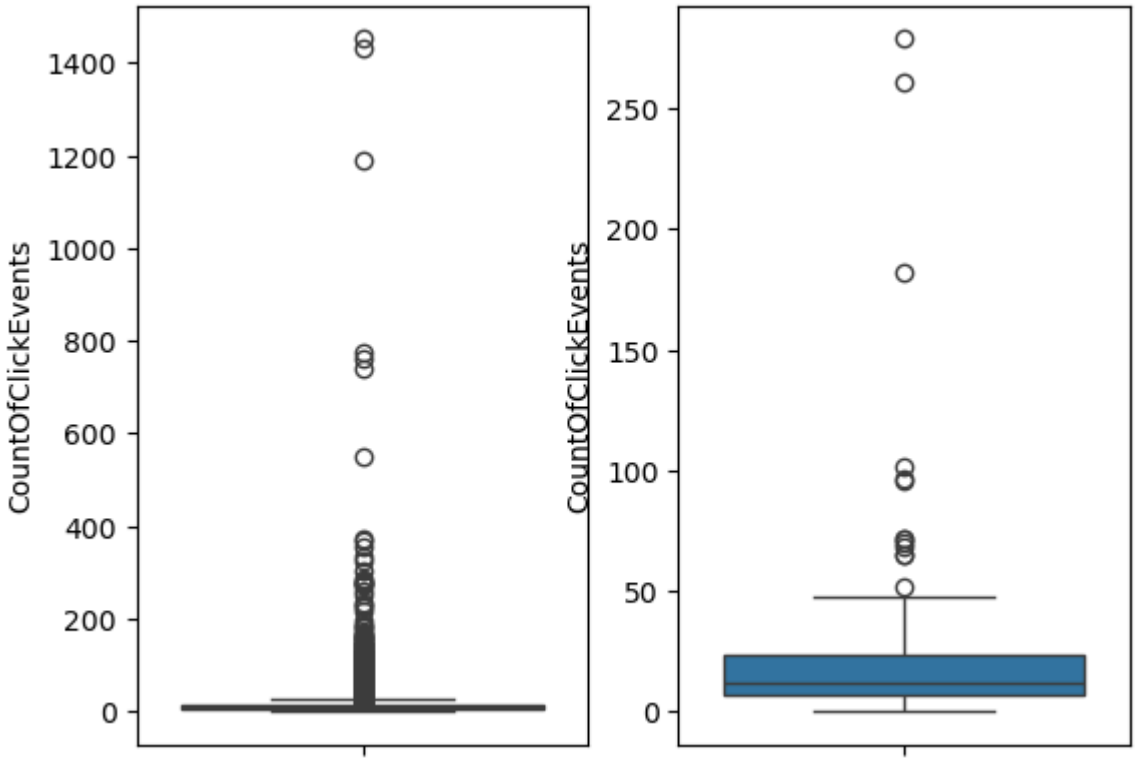
Out[1214]:
```
count    30595.000000
mean        12.937506
std         20.995226
min          0.000000
25%          6.000000
50%          9.000000
75%         14.000000
max       1452.000000
Name: CountOfClickEvents, dtype: float64
```

In [121…
```
data[data['Converted']==1]['CountOfClickEvents'].describe()
```

Out[1215]:
```
count    178.000000
mean      21.994382
std       34.367817
min        0.000000
25%        7.000000
50%       12.000000
75%       23.750000
max      279.000000
Name: CountOfClickEvents, dtype: float64
```

In [129…
```
plt.subplot(1,2,1)
sns.boxplot(data['CountOfClickEvents'])
plt.subplot(1,2,2)
sns.boxplot(data[data['Converted']==1]['CountOfClickEvents'])
```

Out[1294]:
```
<Axes: ylabel='CountOfClickEvents'>
```

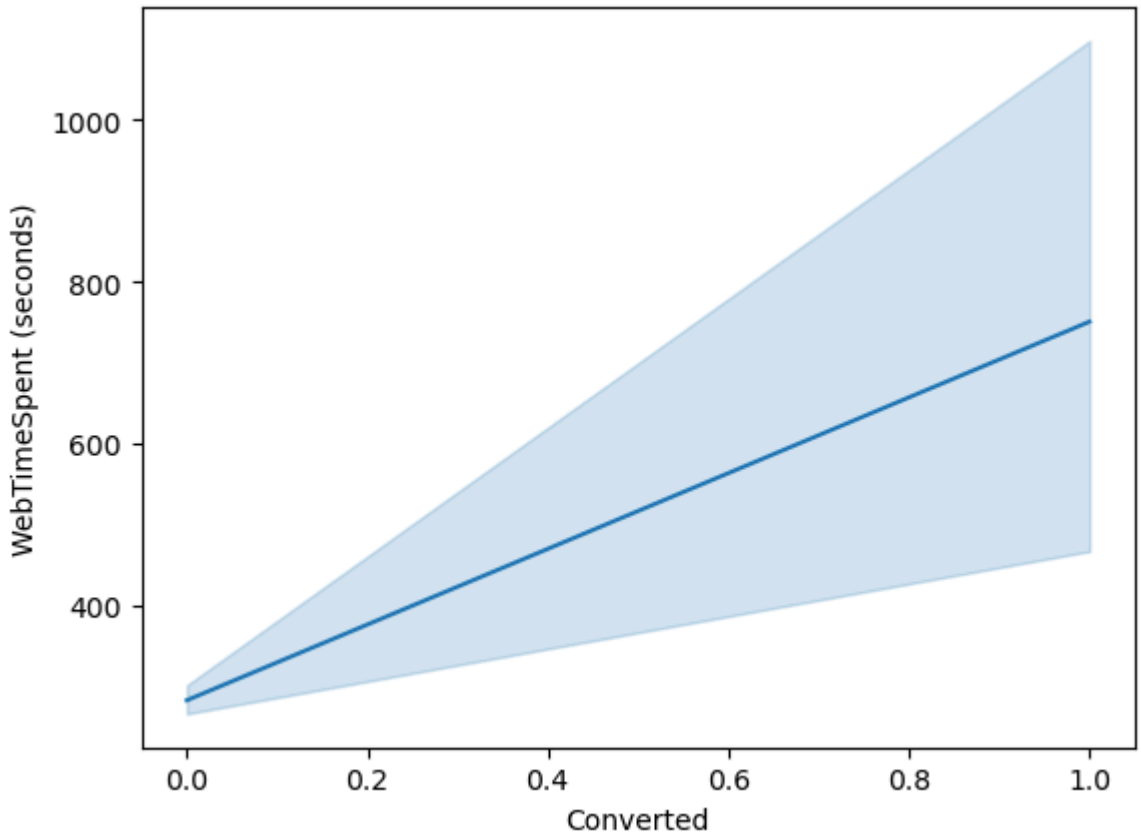


TimeSpent

Web TimeSpent

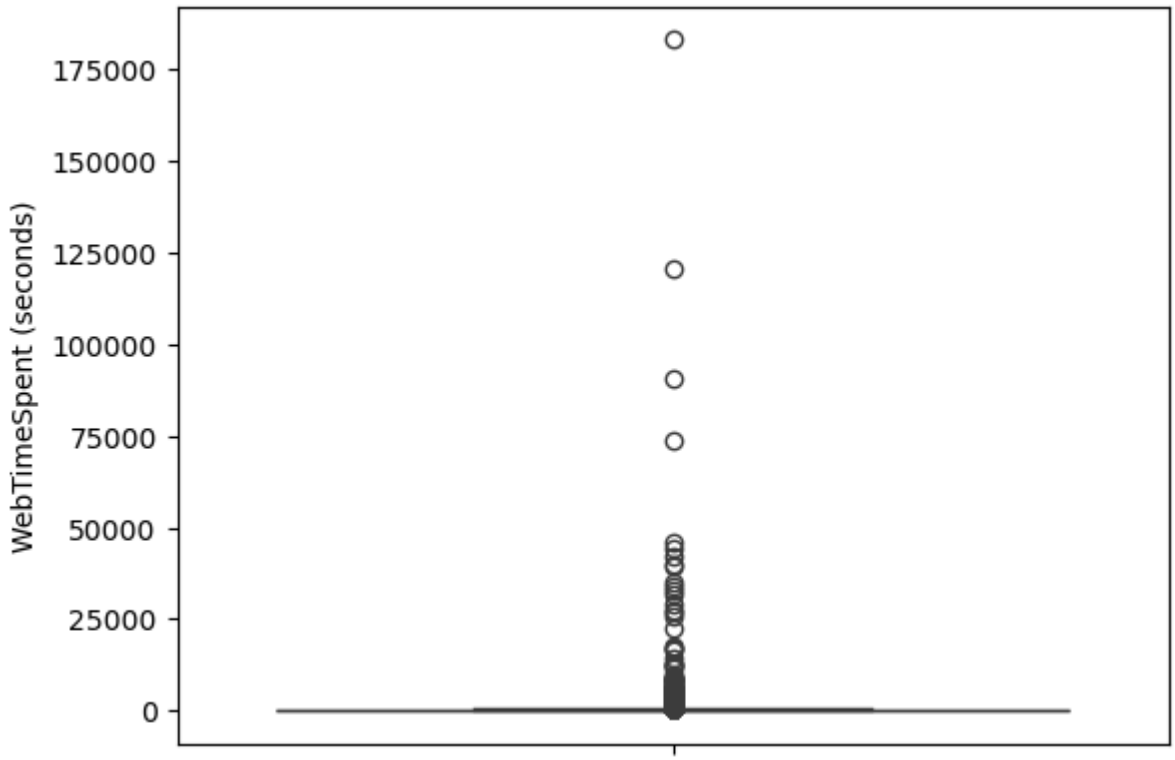
In [129…
```
sns.lineplot(data=data, x='Converted', y='WebTimeSpent (seconds)')
```

Out[1295]:
```
<Axes: xlabel='Converted', ylabel='WebTimeSpent (seconds)'>
```



In [129…
```
sns.boxplot(data['WebTimeSpent (seconds)'])
```

Out[1296]:
```
<Axes: ylabel='WebTimeSpent (seconds)'>
```



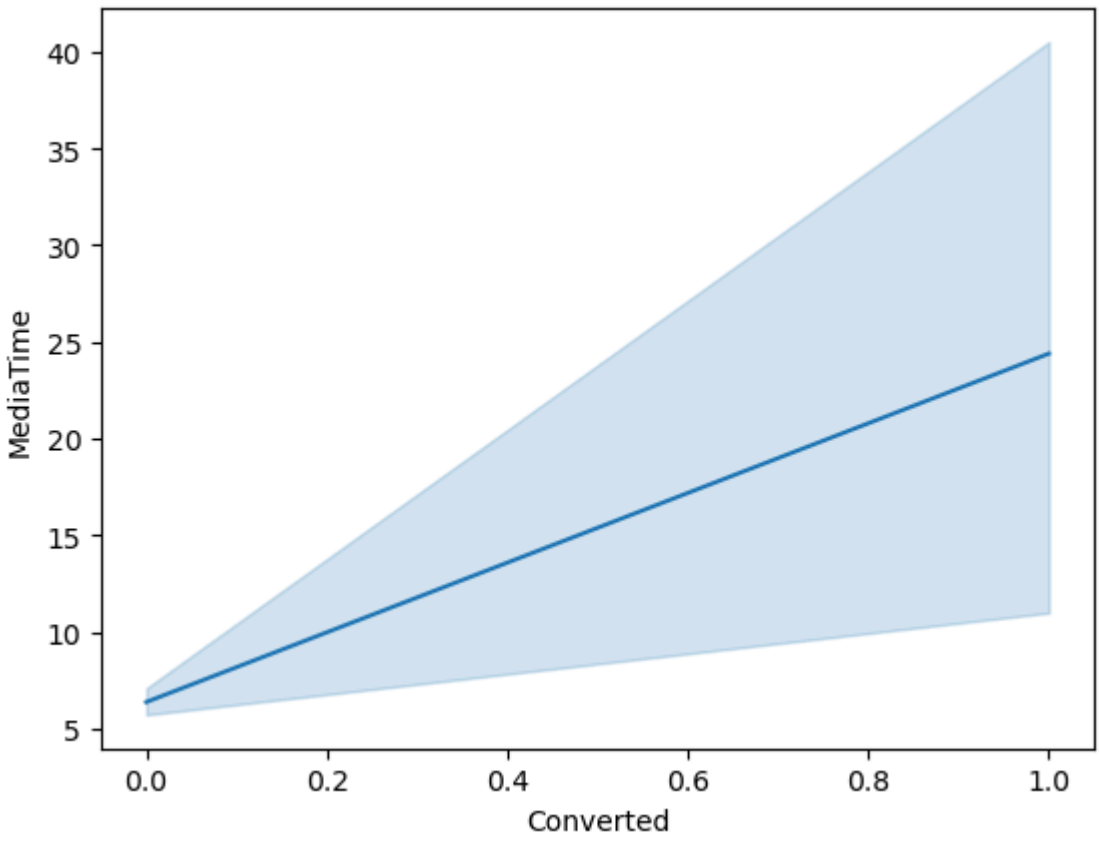
In [129…
```
print(data[data['Converted']==1]['WebTimeSpent (seconds)'].describe())
print(data[data['Converted']==0]['WebTimeSpent (seconds)'].describe())
```

```
count      178.000000
mean       750.456000
std       2087.843881
min          6.399000
25%        151.554750
50%        271.293000
75%        520.304250
max      17130.687000
Name: WebTimeSpent (seconds), dtype: float64
count     30595.000000
mean        282.474635
std        1654.981120
min           0.000000
25%          93.912000
50%         159.491000
75%         281.332000
max      183142.689000
Name: WebTimeSpent (seconds), dtype: float64
```

MediaTime

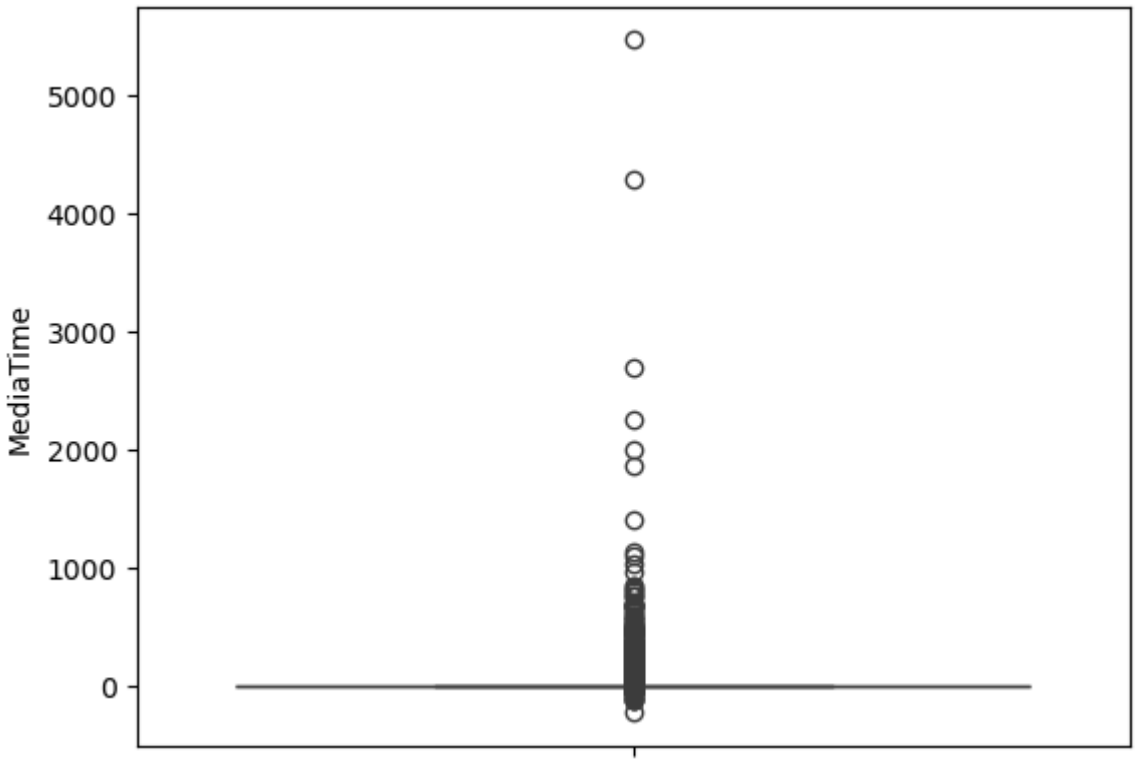
In [129…
```
sns.lineplot(data=data, x='Converted', y='MediaTime')
```

Out[1298]:
```
<Axes: xlabel='Converted', ylabel='MediaTime'>
```



In [129…
```
sns.boxplot(data['MediaTime'])
```

Out[1299]:
```
<Axes: ylabel='MediaTime'>
```



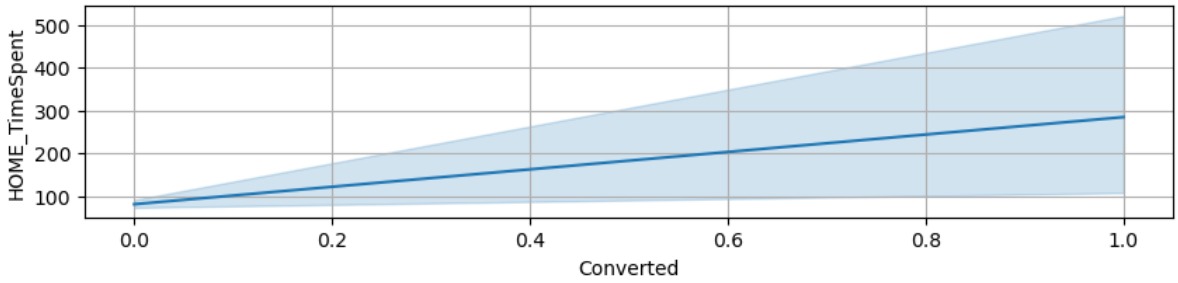
In [130…
```
print(data[data['Converted']==1]['MediaTime'].describe())
print(data[data['Converted']==0]['MediaTime'].describe())
```

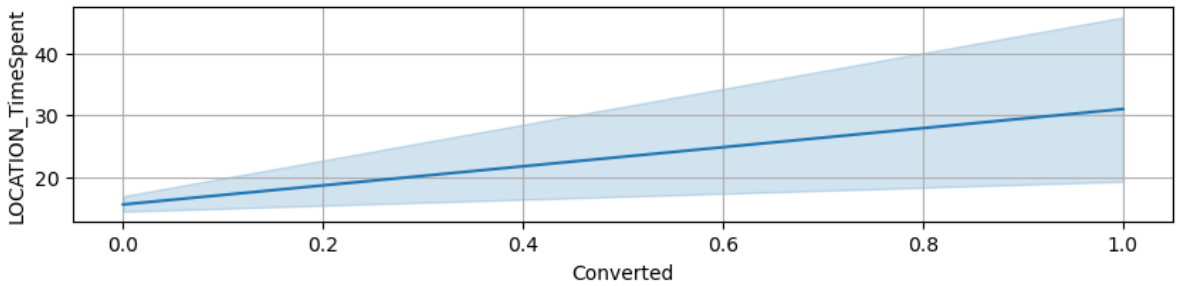
```
count    178.000000
mean      24.398764
std      101.844373
min       -7.220000
25%        0.000000
50%        0.000000
75%        0.000000
max      775.580000
Name: MediaTime, dtype: float64
count    30595.000000
mean         6.380424
std         61.128721
min       -217.500000
25%          0.000000
50%          0.000000
75%          0.000000
max       5474.330000
Name: MediaTime, dtype: float64
```

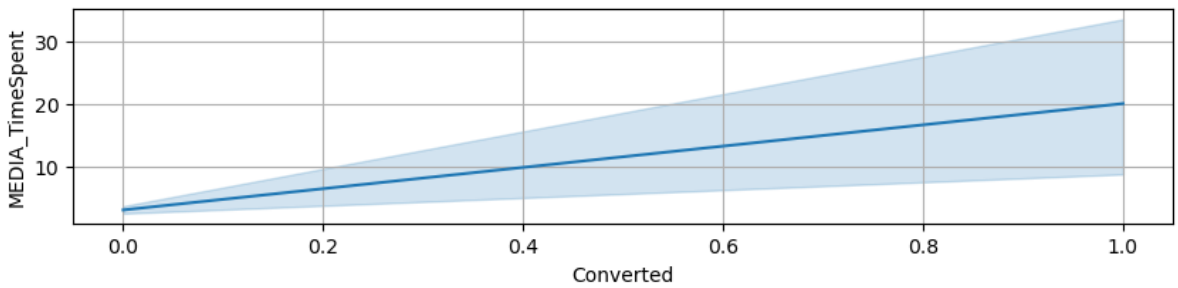
TimeSpent Columns

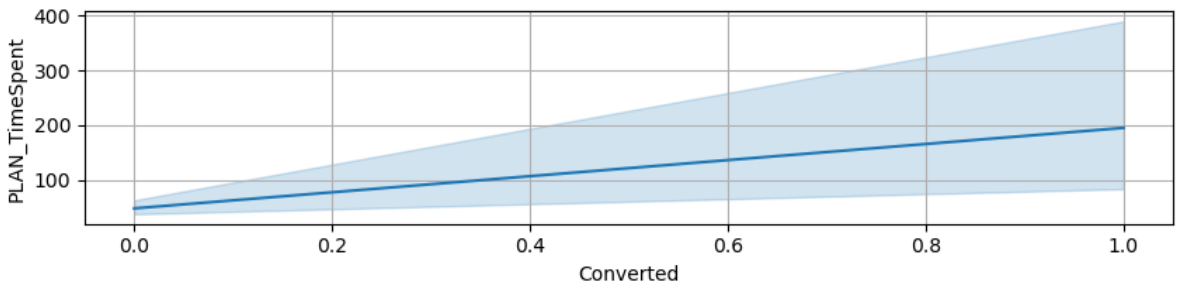
In [130…
```
TimeSpentlist = ['HOME_TimeSpent','LOCATION_TimeSpent','MEDIA_TimeSpent','PLAN_Time
```

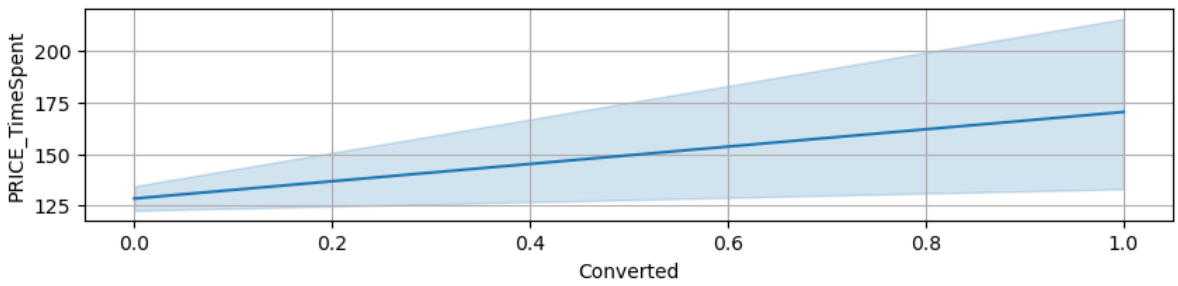
In [130…
```
for i,col in enumerate(TimeSpentlist,1):

    plt.figure(figsize=(10,16))

    plt.subplot(7,1,i)
    sns.lineplot(data=data, x='Converted', y=col)

    plt.grid()
```

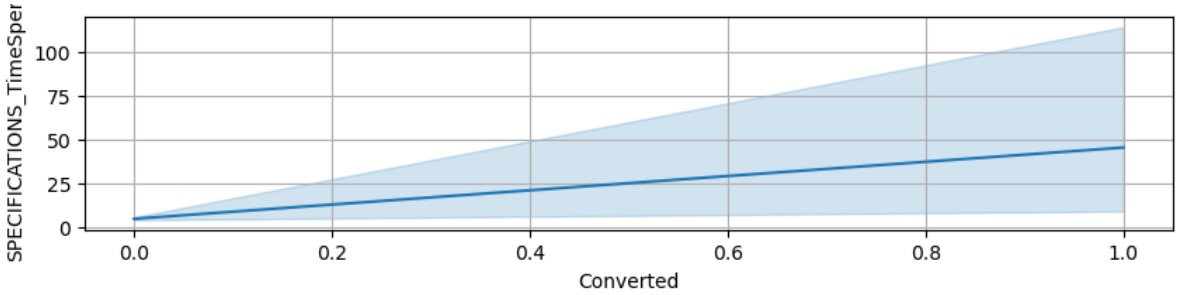

HOME_TimeSpent
500
400
300
200
100
0.0
0.2
0.4
0.6
0.8
1.0
Converted

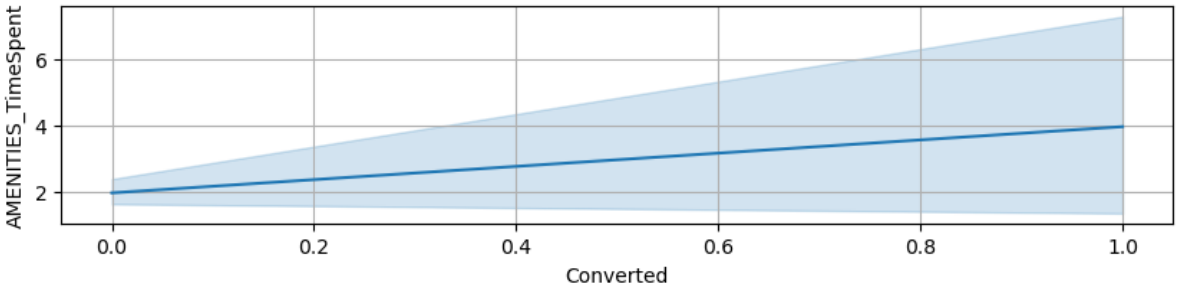

LOCATION_TimeSpent
40
30
20
0.0
0.2
0.4
0.6
0.8
1.0
Converted


MEDIA_TimeSpent
30
20
10
0.0
0.2
0.4
0.6
0.8
1.0
Converted


PLAN_TimeSpent
400
300
200
100
0.0
0.2
0.4
0.6
0.8
1.0
Converted

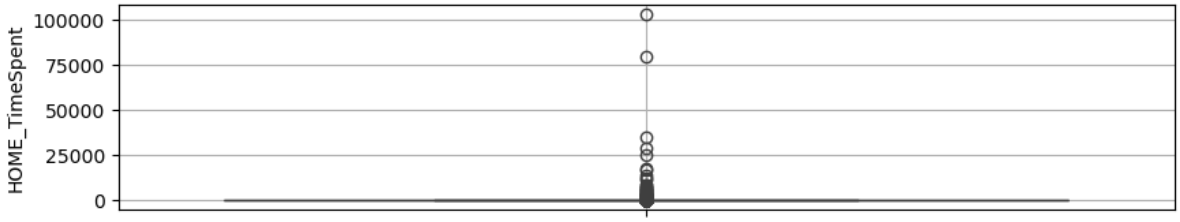

PRICE_TimeSpent
200
175
150
125
0.0
0.2
0.4
0.6
0.8
1.0
Converted

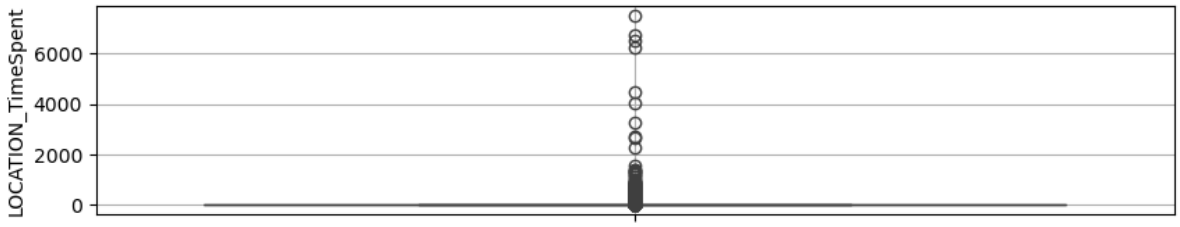

SPECIFICATIONS_TimeSper
100
75
50
25
0
0.0
0.2
0.4
0.6
0.8
1.0
Converted

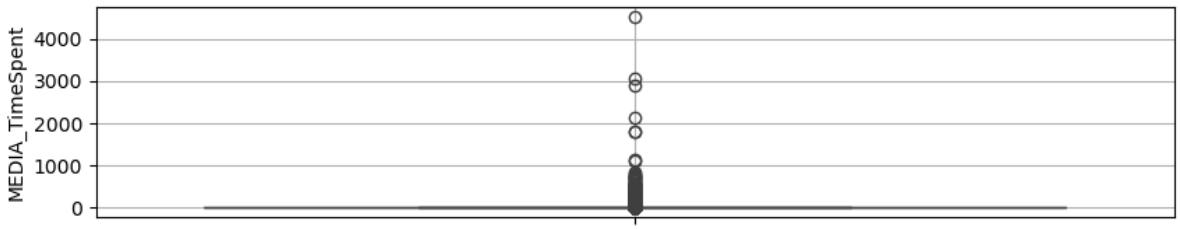


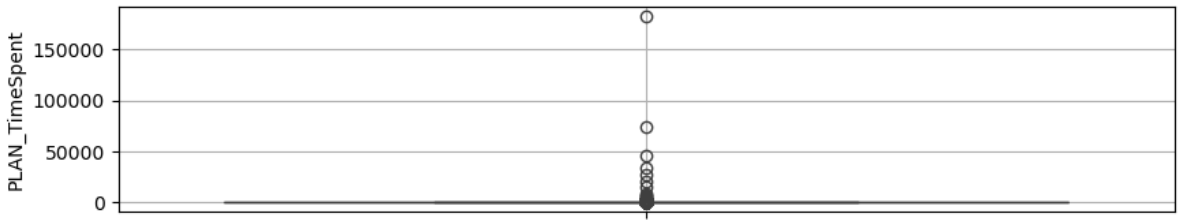
In [130…

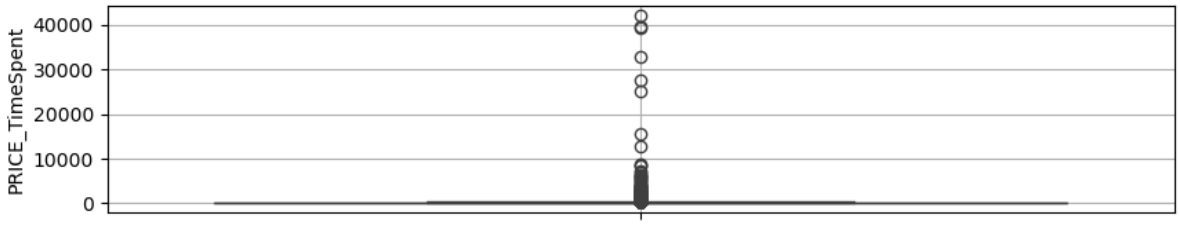
```
for i,col in enumerate(TimeSpentlist,1):

    plt.figure(figsize=(10,16))

    plt.subplot(7,1,i)
    sns.boxplot(data[col])

    plt.grid()
```

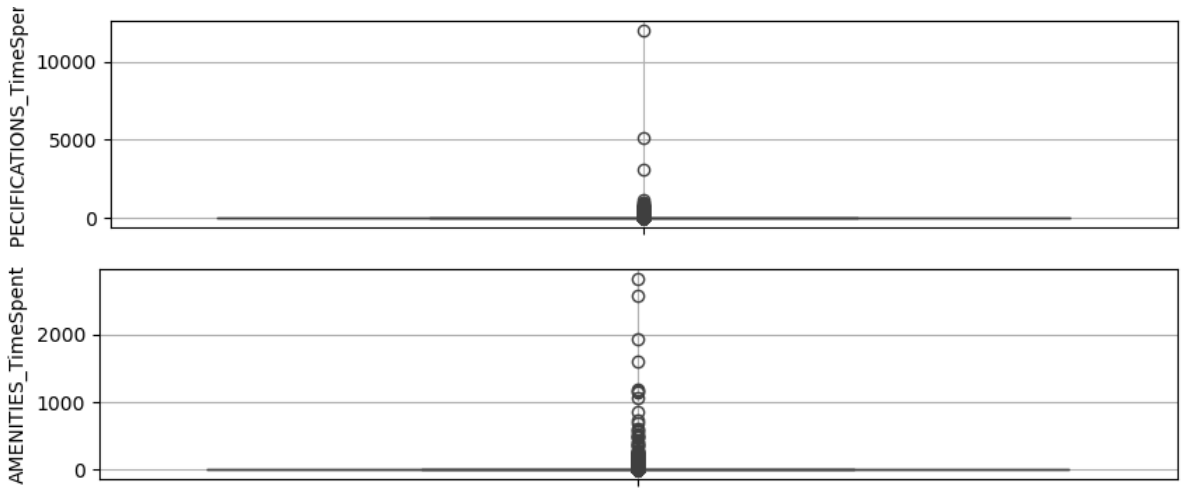












In [130…

```
for i,col in enumerate(TimeSpentlist,1):
    print(data[data['Converted']==1][col].describe())
    print(data[data['Converted']==0][col].describe())
```

```
count      178.000000
mean       284.356247
std       1503.463317
min          0.000000
25%         19.104750
50%         57.870500
75%        144.843000
max      17130.687000
Name: HOME_TimeSpent, dtype: float64
count     30595.000000
mean         80.737560
std         832.477134
min           0.000000
25%          16.184000
50%          42.932000
75%          84.474500
max      103350.413000
Name: HOME_TimeSpent, dtype: float64
count    178.000000
mean      31.019152
std       90.886981
min        0.000000
25%        0.000000
50%        0.000000
75%       19.677250
max      847.621000
Name: LOCATION_TimeSpent, dtype: float64
count    30595.000000
mean        15.513607
std        105.986854
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max       7525.106000
Name: LOCATION_TimeSpent, dtype: float64
count    178.000000
mean      20.141517
std       86.573363
min        0.000000
25%        0.000000
50%        0.000000
75%        0.000000
max      779.589000
Name: MEDIA_TimeSpent, dtype: float64
count    30595.000000
mean         3.204737
std         48.722479
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max       4530.068000
Name: MEDIA_TimeSpent, dtype: float64
count      178.000000
mean       195.013107
std       1175.456045
min          0.000000
25%          0.000000
50%          0.000000
75%         96.801250
max      15353.263000
Name: PLAN_TimeSpent, dtype: float64
count     30595.000000
```

```
mean         47.785357
std        1197.756407
min           0.000000
25%           0.000000
50%           0.000000
75%           0.000000
max      182771.713000
Name: PLAN_TimeSpent, dtype: float64
count     178.000000
mean      170.373017
std       280.893504
min         0.000000
25%         8.827000
50%       105.821000
75%       200.161750
max      2164.411000
Name: PRICE_TimeSpent, dtype: float64
count    30595.000000
mean       128.431537
std        545.851006
min          0.000000
25%          0.000000
50%         88.759000
75%        163.923500
max      42189.682000
Name: PRICE_TimeSpent, dtype: float64
count     178.000000
mean       45.591680
std       389.714795
min         0.000000
25%         0.000000
50%         0.000000
75%         0.000000
max      5122.119000
Name: SPECIFICATIONS_TimeSpent, dtype: float64
count    30595.000000
mean         4.849704
std         78.235459
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max      12003.949000
Name: SPECIFICATIONS_TimeSpent, dtype: float64
count    178.000000
mean       3.961281
std       21.049870
min        0.000000
25%        0.000000
50%        0.000000
75%        0.000000
max      202.333000
Name: AMENITIES_TimeSpent, dtype: float64
count    30595.000000
mean         1.952133
std         33.797148
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max       2830.945000
Name: AMENITIES_TimeSpent, dtype: float64
```

Time spent is higher for Converted leads

Outlier Removal for Numerical Column

We will replace outlier for all Numerical column with Median

In [130…
```
numerical_features = ['CountOfClickEvents','MediaTime','WebTimeSpent (seconds)','HC
                      'MEDIA_TimeSpent','PRICE_TimeSpent','PLAN_TimeSpent','SPEC
```

In [130…
```
data['CountOfClickEvents'].describe()
```

Out[1306]:
```
count    30773.000000
mean        12.989894
std         21.107236
min          0.000000
25%          6.000000
50%          9.000000
75%         14.000000
max       1452.000000
Name: CountOfClickEvents, dtype: float64
```

In [130…
```
for cols in numerical_features:
    Q1 = data[cols].quantile(0.25)
    Q3 = data[cols].quantile(0.75)
    median = data[cols].quantile(0.50)
    IQR = Q3 - Q1
    lower = Q1 - 1.5*IQR
    upper = Q3 + 1.5*IQR
    data[cols] = data[cols].apply(lambda x: x if lower<x<upper else (median+1))
```

In [130…
```
for i,col in enumerate(TimeSpentlist,1):
    print(data[data['Converted']==1][col].describe())
    print(data[data['Converted']==0][col].describe())
```

```
count    178.000000
mean      52.962073
std       44.666719
min        0.000000
25%       19.104750
50%       43.973000
75%       74.117000
max      179.784000
Name: HOME_TimeSpent, dtype: float64
count    30595.000000
mean        48.996561
std         42.520077
min          0.000000
25%         16.184000
50%         42.932000
75%         71.071500
max        187.367000
Name: HOME_TimeSpent, dtype: float64
count    178.0
mean       1.0
std        0.0
min        1.0
25%        1.0
50%        1.0
75%        1.0
max        1.0
Name: LOCATION_TimeSpent, dtype: float64
count    30595.0
mean         1.0
std          0.0
min          1.0
25%          1.0
50%          1.0
75%          1.0
max          1.0
Name: LOCATION_TimeSpent, dtype: float64
count    178.0
mean       1.0
std        0.0
min        1.0
25%        1.0
50%        1.0
75%        1.0
max        1.0
Name: MEDIA_TimeSpent, dtype: float64
count    30595.0
mean         1.0
std          0.0
min          1.0
25%          1.0
50%          1.0
75%          1.0
max          1.0
Name: MEDIA_TimeSpent, dtype: float64
count    178.0
mean       1.0
std        0.0
min        1.0
25%        1.0
50%        1.0
75%        1.0
max        1.0
Name: PLAN_TimeSpent, dtype: float64
count    30595.0
```

```
mean         1.0
std          0.0
min          1.0
25%          1.0
50%          1.0
75%          1.0
max          1.0
Name: PLAN_TimeSpent, dtype: float64
count    178.000000
mean     109.196978
std       98.668797
min        0.000000
25%        8.827000
50%       89.821000
75%      168.598000
max      388.155000
Name: PRICE_TimeSpent, dtype: float64
count    30595.000000
mean        97.994434
std         93.588168
min          0.000000
25%          0.000000
50%         88.759000
75%        150.684000
max        410.102000
Name: PRICE_TimeSpent, dtype: float64
count    178.0
mean       1.0
std        0.0
min        1.0
25%        1.0
50%        1.0
75%        1.0
max        1.0
Name: SPECIFICATIONS_TimeSpent, dtype: float64
count    30595.0
mean         1.0
std          0.0
min          1.0
25%          1.0
50%          1.0
75%          1.0
max          1.0
Name: SPECIFICATIONS_TimeSpent, dtype: float64
count    178.0
mean       1.0
std        0.0
min        1.0
25%        1.0
50%        1.0
75%        1.0
max        1.0
Name: AMENITIES_TimeSpent, dtype: float64
count    30595.0
mean         1.0
std          0.0
min          1.0
25%          1.0
50%          1.0
75%          1.0
max          1.0
Name: AMENITIES_TimeSpent, dtype: float64
```

2. Lead Scoring Logic Development

In [130…
```
relevant_cat_features = ['TimeOfDay','Project','UTM-Source','OperatingSystem']
```

In [131…
```
relevant_num_features = ['CountOfClickEvents','WebTimeSpent (seconds)','HOME_TimeSp
```

In [131…
```
df = data[['TimeOfDay','Project','UTM-Source','OperatingSystem',
           'CountOfClickEvents','WebTimeSpent (seconds)','HOME_TimeSpent','PRICE_Ti
```

In [131…
```
df
```

Out[1312]:

	TimeOfDay	Project	UTM-Source	OperatingSystem	CountOfClickEvents	WebTimeSpent (seconds)	HOM
0	Afternoon	Springs	Google	Windows	15.0	501.779	
1	Afternoon	Spectra	GDN	Android	5.0	54.175	
2	Evening	Springs	GMB	Android	18.0	292.363	
3	Afternoon	Spectra	Google	Android	6.0	178.201	
4	Afternoon	Spectra	Google	Android	10.0	160.877	
...	...	...	...	...	...	...	
30768	Afternoon	Spectra	Others	Android	24.0	160.877	
30769	Morning	Spectra	Organic	Mac	10.0	160.877	
30770	Evening	Spectra	sakshi	Android	5.0	219.587	
30771	Morning	Springs	Google	Android	9.0	113.743	
30772	Night	Spectra	Eenadu	Mac	10.0	160.877	

30773 rows × 9 columns

2.1 Updating typos for UTM-Source

In [131…
```
df.loc[df['UTM-Source']=='fb',['UTM-Source']]= 'Facebook'
df.loc[df['UTM-Source']=='whatsapp',['UTM-Source']]= 'WhatsApp'
df.loc[df['UTM-Source']=='Whatsapp',['UTM-Source']]= 'WhatsApp'
df.loc[df['UTM-Source']=='sakshi',['UTM-Source']]= 'Sakshi'
df.loc[df['UTM-Source']=='ig',['UTM-Source']]= 'IG'
df.loc[df['UTM-Source']=='google',['UTM-Source']]= 'Google'
```

In [131…
```
df['UTM-Source'].value_counts()
```

Out[1314]:
```
Google            19830
Organic            8536
Direct              853
FIM                 408
Yoptima             281
GMB                 204
IG                  171
WhatsApp            148
Newsprint           107
Mygate               74
Facebook             39
Blog                 32
Inshorts             31
LinkedIn             14
Adonmo               10
Eenadu                8
tma                   5
Youtube               5
Sakshi                4
ASBL                  3
GoogleAds             2
zoom                  2
PD                    2
TOI                   1
GDN                   1
Google_Organic        1
Others                1
Name: UTM-Source, dtype: int64
```

Combining UTM-Source value less than 100 into one group 'OTHER'

In [131…
```
df['UTM-Source']=df['UTM-Source'].apply(lambda x: x if x in ['Newsprint','WhatsApp'
```

In [131…
```
UTMSourceData = df.groupby('UTM-Source')['Converted'].mean()
OSData = df.groupby('OperatingSystem')['Converted'].mean()
```

In [131…
```
df.groupby('UTM-Source')['Converted'].mean()
```

Out[1317]:
```
UTM-Source
Direct       0.002345
FIM          0.002451
GMB          0.004902
Google       0.004286
IG           0.000000
Newsprint    0.018692
OTHER        0.012766
Organic      0.008786
WhatsApp     0.033784
Yoptima      0.014235
Name: Converted, dtype: float64
```

2.2 Changing Categorical features to Numerical

In [131…
```
df['UTM-Source'] = df.groupby('UTM-Source')['Converted'].transform('mean')
```

In [131…
```
df['OperatingSystem'] = df.groupby('OperatingSystem')['Converted'].transform('mean'
```

In [132…
```
df['TimeOfDay'] = df.groupby('TimeOfDay')['Converted'].transform('mean')
```

In [132…
```
df['Project'] = df.groupby('Project')['Converted'].transform('mean')
```

In [132…
```
df
```

Out[1322]:

	TimeOfDay	Project	UTM-Source	OperatingSystem	CountOfClickEvents	WebTimeSpent (seconds)	H
0	0.007035	0.006795	0.004286	0.012684	15.0	501.779	
1	0.007035	0.006024	0.012766	0.003738	5.0	54.175	
2	0.005179	0.006795	0.004902	0.003738	18.0	292.363	
3	0.007035	0.006024	0.004286	0.003738	6.0	178.201	
4	0.007035	0.006024	0.004286	0.003738	10.0	160.877	
...	...	...	...	...	...	...	
30768	0.007035	0.006024	0.012766	0.003738	24.0	160.877	
30769	0.004858	0.006024	0.008786	0.011287	10.0	160.877	
30770	0.005179	0.006024	0.012766	0.003738	5.0	219.587	
30771	0.004858	0.006795	0.004286	0.003738	9.0	113.743	
30772	0.005491	0.006024	0.012766	0.011287	10.0	160.877	

30773 rows × 9 columns

In [132…
```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

df = pd.DataFrame(scaler.fit_transform(df), columns=df.columns)
df
```

Out[1323]:

	TimeOfDay	Project	UTM-Source	OperatingSystem	CountOfClickEvents	WebTimeSpent (seconds)	HO
0	1.000000	1.00000	0.126878	1.000000	0.60	0.888861	
1	1.000000	0.82774	0.377872	0.294694	0.20	0.095967	
2	0.147648	1.00000	0.145098	0.294694	0.72	0.517898	
3	1.000000	0.82774	0.126878	0.294694	0.24	0.315669	
4	1.000000	0.82774	0.126878	0.294694	0.40	0.284981	
...	...	...	...	...	...	...	
30768	1.000000	0.82774	0.377872	0.294694	0.96	0.284981	
30769	0.000000	0.82774	0.260075	0.889810	0.40	0.284981	
30770	0.147648	0.82774	0.377872	0.294694	0.20	0.388981	
30771	0.000000	1.00000	0.126878	0.294694	0.36	0.201487	
30772	0.290971	0.82774	0.377872	0.889810	0.40	0.284981	

30773 rows × 9 columns

In [132...
```
df
```

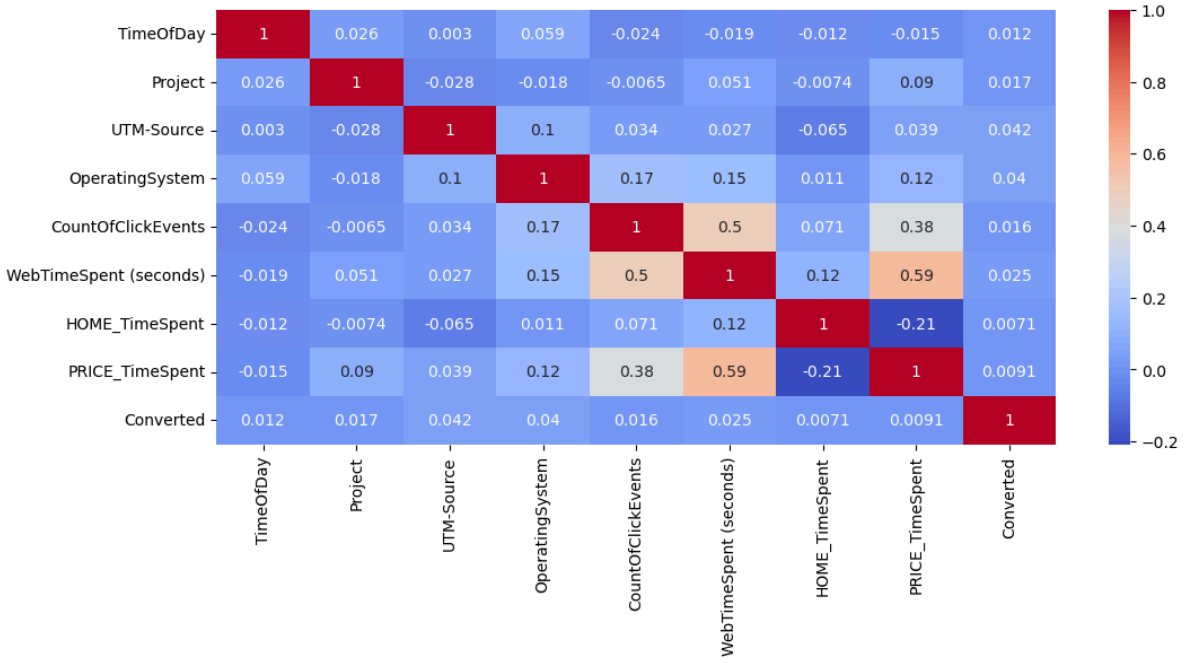
Out[1325]:

	TimeOfDay	Project	UTM-Source	OperatingSystem	CountOfClickEvents	WebTimeSpent (seconds)	HO
0	1.000000	1.00000	0.126878	1.000000	0.60	0.888861	
1	1.000000	0.82774	0.377872	0.294694	0.20	0.095967	
2	0.147648	1.00000	0.145098	0.294694	0.72	0.517898	
3	1.000000	0.82774	0.126878	0.294694	0.24	0.315669	
4	1.000000	0.82774	0.126878	0.294694	0.40	0.284981	
...	...	...	...	...	...	...	
30768	1.000000	0.82774	0.377872	0.294694	0.96	0.284981	
30769	0.000000	0.82774	0.260075	0.889810	0.40	0.284981	
30770	0.147648	0.82774	0.377872	0.294694	0.20	0.388981	
30771	0.000000	1.00000	0.126878	0.294694	0.36	0.201487	
30772	0.290971	0.82774	0.377872	0.889810	0.40	0.284981	

30773 rows × 9 columns

2.3 Correlation

In [132...
```
Corr_mat = df.corr()
plt.figure(figsize=(12,5))
sns.heatmap(Corr_mat,annot=True, cmap='coolwarm')
plt.show()
```



In [132…]
```
Corr_mat
```

Out[1327]:

	TimeOfDay	Project	UTM-Source	OperatingSystem	CountOfClickEvents	Web
TimeOfDay	1.000000	0.025859	0.002958	0.058945	-0.024473	
Project	0.025859	1.000000	-0.028454	-0.018043	-0.006453	
UTM-Source	0.002958	-0.028454	1.000000	0.099837	0.033872	
OperatingSystem	0.058945	-0.018043	0.099837	1.000000	0.170968	
CountOfClickEvents	-0.024473	-0.006453	0.033872	0.170968	1.000000	
WebTimeSpent (seconds)	-0.019434	0.051090	0.027249	0.152955	0.498820	
HOME_TimeSpent	-0.011925	-0.007402	-0.064661	0.010871	0.071145	
PRICE_TimeSpent	-0.014645	0.089780	0.039157	0.123977	0.383174	
Converted	0.012363	0.016642	0.041671	0.039932	0.016485	

In [132…]
```
FeatureWeightage = Corr_mat['Converted'].apply(lambda x: round(x*1000))
```

In [132…]
```
FeatureWeightage
```

Out[1329]:
```
TimeOfDay                 12
Project                   17
UTM-Source                42
OperatingSystem           40
CountOfClickEvents        16
WebTimeSpent (seconds)    25
HOME_TimeSpent             7
PRICE_TimeSpent            9
Converted               1000
Name: Converted, dtype: int64
```

Feature weightage also indicates the Highest value a feature can take. We will other values accordingly

Scoring example

In [133…
```
df['TimeOfDay'] * FeatureWeightage.loc['TimeOfDay'] / df['TimeOfDay'].max()
```

Out[1331]:
```
0        12.000000
1        12.000000
2         1.771778
3        12.000000
4        12.000000
           ...
30768    12.000000
30769     0.000000
30770     1.771778
30771     0.000000
30772     3.491654
Name: TimeOfDay, Length: 30773, dtype: float64
```

2.4 Scoring Formula

In [133…
```
for cols in df.columns:
    maxval = df[cols].max()
    df[cols] = df[cols] * FeatureWeightage.loc[cols] /maxval
```

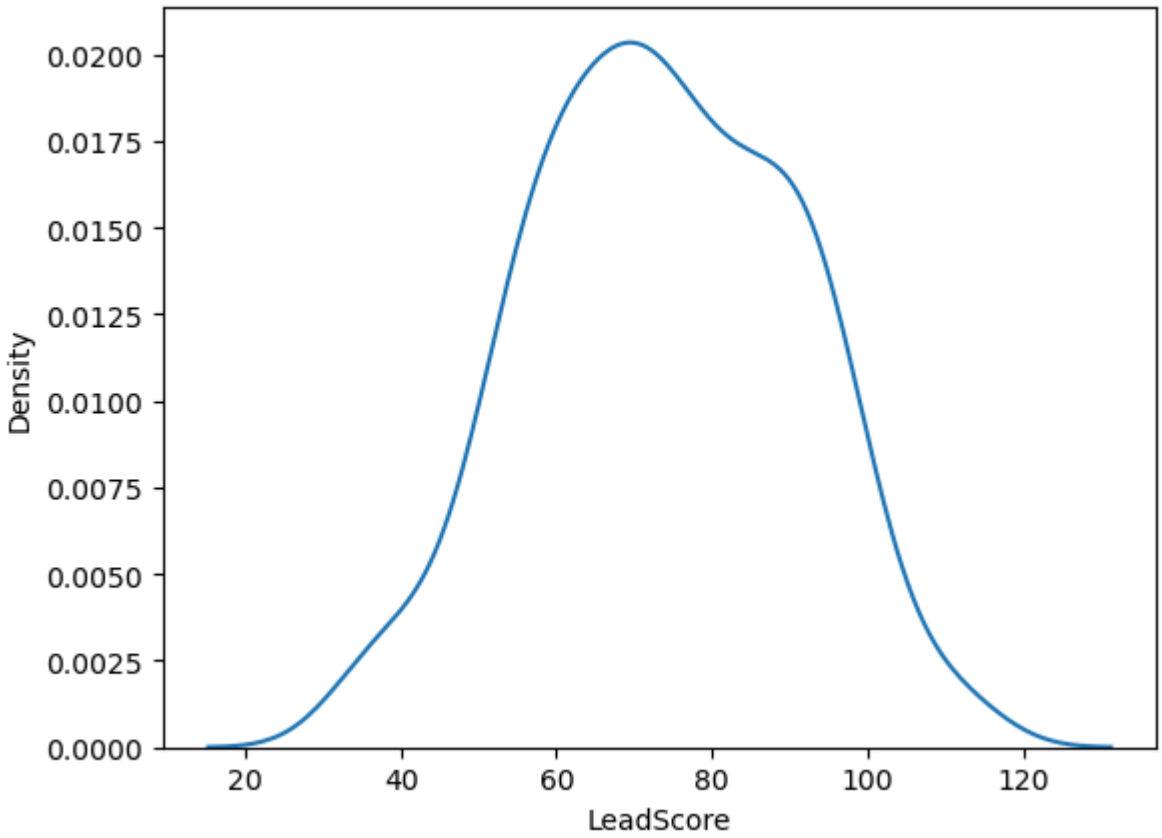
In [133…
```
df['LeadScore']= df.drop('Converted',axis=1).sum(axis=1)
```

In [133…
```
df[df['Converted']!=0]['LeadScore'].describe()
```

Out[1334]:
```
count    178.000000
mean      73.522380
std       16.960847
min       33.333265
25%       61.234337
50%       73.022760
75%       86.492508
max      113.188550
Name: LeadScore, dtype: float64
```

In [133…
```
sns.kdeplot(df[df['Converted']!=0]['LeadScore'])
```

Out[1335]:
```
<Axes: xlabel='LeadScore', ylabel='Density'>
```

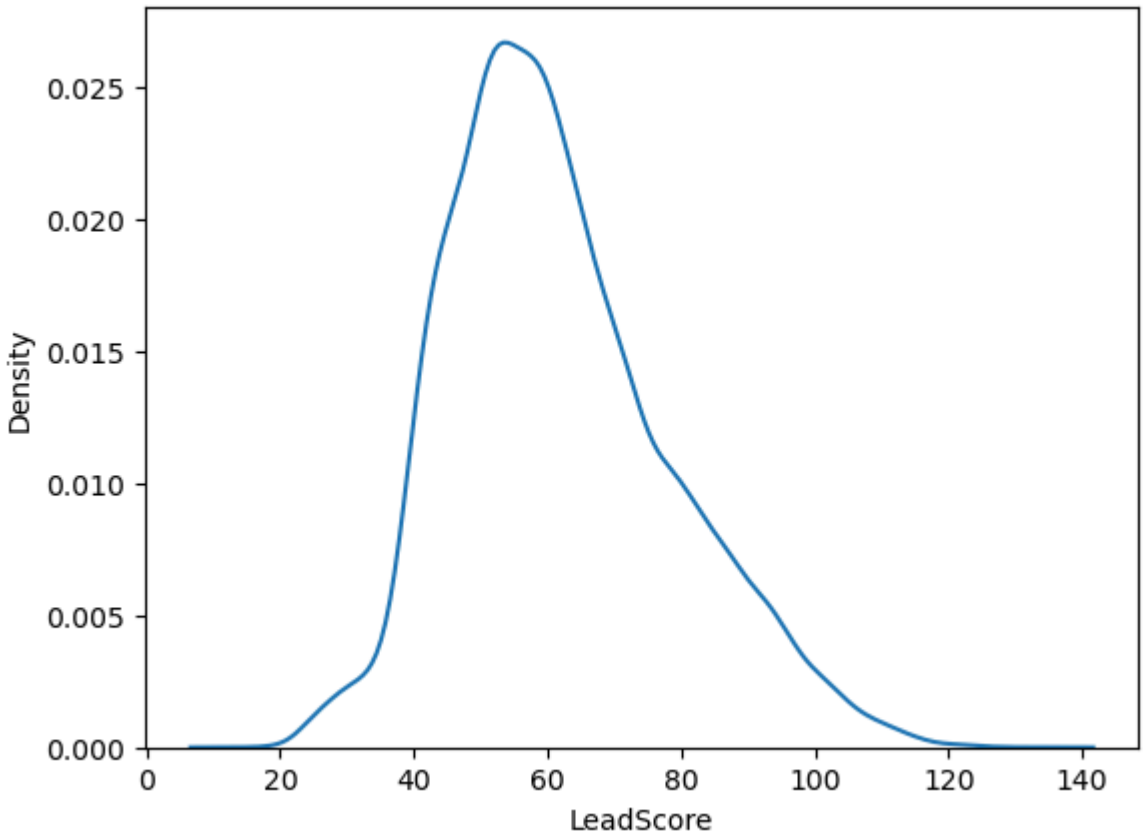


In [133…
```
df[df['Converted']==0]['LeadScore'].describe()
```

Out[1336]:
```
count    30595.000000
mean        61.690940
std         16.632536
min         12.932862
25%         49.815181
50%         59.167177
75%         71.552618
max        135.314744
Name: LeadScore, dtype: float64
```

In [133…
```
sns.kdeplot(df[df['Converted']==0]['LeadScore'])
```

Out[1337]:
```
<Axes: xlabel='LeadScore', ylabel='Density'>
```



Since Lead score for converted leads and Not converted tend to be normal, we will do Z-test to find out accuracy of our lead score model

In [133…
```
LeadScoreForConvertedLeads = df[df['Converted']!=0]['LeadScore']
LeadScoreForNotConvertedLeads = df[df['Converted']==0]['LeadScore']
```

2.6 Effectiveness of Lead Score

Null Hypothesis: LeadScore for Converted Leads and Not Converted Leads is similar.
Alternate Hypothesis: LeadScore for Converted Leads and Not Converted Leads is not similar.

In [133…
```
from statsmodels.stats import weightstats as stests
from scipy import stats

z_score, pval = stests.ztest(x1 = LeadScoreForConvertedLeads, x2= LeadScoreForNotCo
```

In [134…
```
z_score, pval
```

Out[1340]:
```
(9.461928313794239, 3.0231579901423983e-21)
```

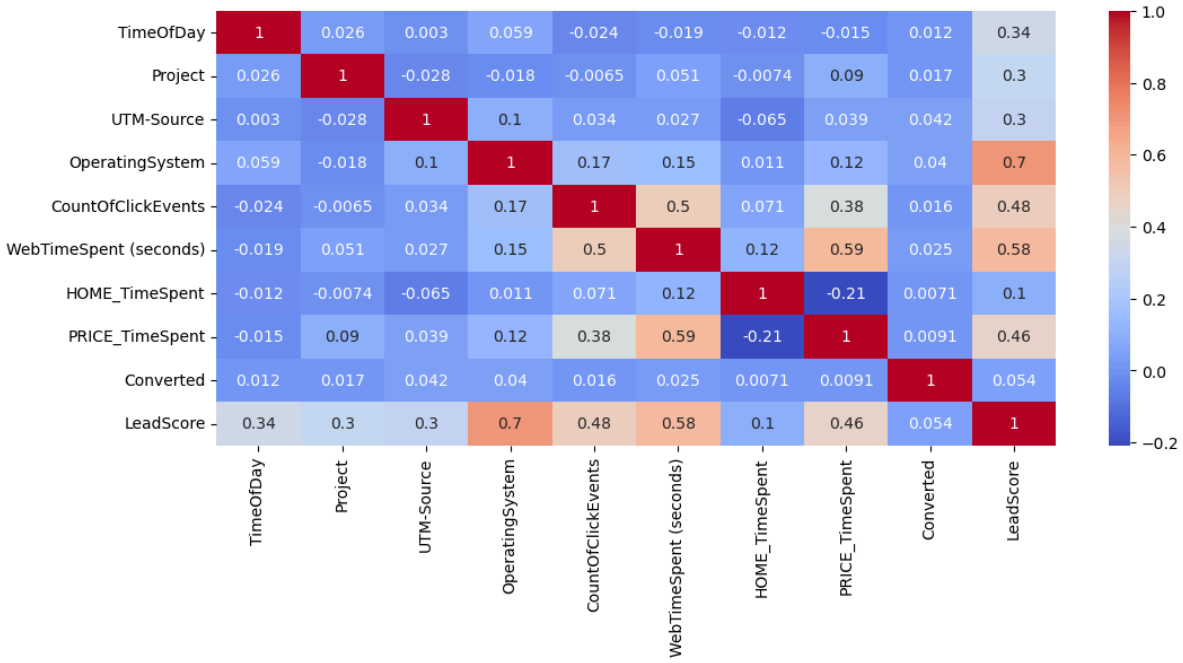
Since p-value is very small, we should with Alternate hypothesis i.e Calculated lead score is signifcantly higher for converted lead

2.7 Questions

2.7.1 What factors significantly contribute to a higher lead score?

In [134…
```
Corr_mat = df.corr()
plt.figure(figsize=(12,5))
```

```
sns.heatmap(Corr_mat,annot=True, cmap='coolwarm')
plt.show()
```



Based on EDA it is detrmined that following are top 4 features which contribute to high lead score:

```
OperatingSystem
WebTimeSpent
PriceTimeSpent
CountOfClickEvents
```

2.7.2 How will you test and validate the scoring model for effectiveness?

I used 2 sample z-test to validate scoring model in section 2.6.

3. Top Contributors/Features

In [134…
```
FeatureWeightage.reset_index().sort_values(by='Converted',ascending=False)
```

Out[1342]:

	index	Converted
8	Converted	1000
2	UTM-Source	42
3	OperatingSystem	40
5	WebTimeSpent (seconds)	25
1	Project	17
4	CountOfClickEvents	16
0	TimeOfDay	12
7	PRICE_TimeSpent	9
6	HOME_TimeSpent	7

3.1 Questions

3.1.1 Which features (e.g., UTM-Source, CountOfClickEvents, WebTimeSpent) have the highest impact on successful conversions?

UTM-Source, OperatingSystem and WebTimeSpent have highest impact on coversion

3.1.2 How can these insights be used to prioritize leads and optimize efforts?

UTM-Source:

In [134…
```
UTMSourceData.reset_index().sort_values(by='Converted',ascending=False)
```

Out[1343]:

	UTM-Source	Converted
8	WhatsApp	0.033784
5	Newsprint	0.018692
9	Yoptima	0.014235
6	OTHER	0.012766
7	Organic	0.008786
2	GMB	0.004902
3	Google	0.004286
1	FIM	0.002451
0	Direct	0.002345
4	IG	0.000000

Above listed UTM source has good conversion rate. Focus should be to reach out to potential customers on these UTM-Source. More presence in these UTM-Source should yield better conversion rate. Whatsapp has best conversion. Organics traffics has a good conversion rate, awareness effort should be made to get more organic traffic.

Operating System

In [134…
```
OSData.reset_index().sort_values(by='Converted',ascending=False)
```

Out[1344]:

	OperatingSystem	Converted
4	Windows	0.012684
3	Mac	0.011287
2	Linux	0.007937
5	iOS	0.006513
0	Android	0.003738
1	Chrome-OS	0.000000

Windows and Mac are PC based OS, which has better conversion rate. Focus should to be increase presence in PC based resources.

4. Differentiation Features for Positive and Negative Sale Results

4.1 What are the common characteristics of leads that converted successfully compared to those that did not?

1. Time of Day: People whose leads originates in Morning and Afternoon has better rate of conversion.(Section EDA).

2. Leads that are interested in Project Spire has least conversion rate.(EDA Section).

3. Leads Originating from PC based OS has signficantly better conversion rate than Mobile based OS.

4 UTM-Source Whatsapp has best conversion rate while Social media sites like IG has less conversion rate.

5. Converted leads spends more time on web researching about project.

6. Leads originating on Monday, Saturday and Sunday has higher conversion rate than the weekdays.

4.2 Are there specific behavioral patterns that correlate with positive or negative outcomes?

1. Leads originating early in day reflects serious interest of the Visitor.

2. A Person who is seriously interested prefers to search on bigger screen e.g. PC/Laptop

3. Source of lead which is personalised has better chance of converting. Whatsapp is personalised Social netwroking tool than IG/FB

4. Higher the time spent reasearching about the project better are the chances of converting.

5. People who are working and earning tend to search for properties on weekends, hence weekend leads has better converion rate.

4.3. Can you propose any recommendations for improving lead conversion based on your findings?

1. Create a presentable and detailed web presence.

2. Reach out to potential customers through personalised channels.

3. Spread awareness about Project through Events/ Sponsorships to generate Organic interests.

4. Advertisement/ Social media marketing in morning and afternoon.

5. Get more advertisment space during the weekends.

6. Internet advertisement should be more focussed on PC friendly web places.

In [ ]: