In [1]:

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

import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor

import sklearn
from sklearn import metrics
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.feature_selection import RFE
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn.metrics import precision_recall_curve

import warnings
warnings.filterwarnings('ignore')

%matplotlib inline

pd.set_option('display.max_rows', 100)
pd.set_option('display.max_columns', 100)
```

In [2]:

```
source_df = pd.read_csv('D:\Practice\Leadscoring\Leads.csv')
source_df.head(6)
```

Out[2]:

	Prospect ID	Lead Number	Lead Origin	Lead Source	Do Not Email	Do Not Call	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Las
0	7927b2df-8bba-4d29-b9a2-b6e0beafe620	660737	API	Olark Chat	No	No	0	0.0	0	0.0	Pa o
1	2a272436-5132-4136-86fa-dcc88c88f482	660728	API	Organic Search	No	No	0	5.0	674	2.5	
2	8cc8c611-a219-4f35-ad23-fdfd2656bd8a	660727	Landing Page Submission	Direct Traffic	No	No	1	2.0	1532	2.0	
3	0cc2df48-7cf4-4e39-9de9-19797f9b38cc	660719	Landing Page Submission	Direct Traffic	No	No	0	1.0	305	1.0	Un

	Prospect ID	Lead Number	Lead Origin	Lead Source	Do Not Email	Do Not Call	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Las
4	3256f628-e534-4826-9d63-4a8b88782852	660681	Landing Page Submission	Google	No	No	1	2.0	1428	1.0	Cor
5	2058ef08-2858-443e-a01f-a9237db2f5ce	660680	API	Olark Chat	No	No	0	0.0	0	0.0	(Cor

In [3]:
```
source_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9240 entries, 0 to 9239
Data columns (total 37 columns):
 #   Column                                         Non-Null Count  Dtype  
---  ------                                         --------------  -----  
 0   Prospect ID                                    9240 non-null   object 
 1   Lead Number                                    9240 non-null   int64  
 2   Lead Origin                                    9240 non-null   object 
 3   Lead Source                                    9204 non-null   object 
 4   Do Not Email                                   9240 non-null   object 
 5   Do Not Call                                    9240 non-null   object 
 6   Converted                                      9240 non-null   int64  
 7   TotalVisits                                    9103 non-null   float64
 8   Total Time Spent on Website                    9240 non-null   int64  
 9   Page Views Per Visit                           9103 non-null   float64
 10  Last Activity                                  9137 non-null   object 
 11  Country                                        6779 non-null   object 
 12  Specialization                                 7802 non-null   object 
 13  How did you hear about X Education             7033 non-null   object 
 14  What is your current occupation                6550 non-null   object 
 15  What matters most to you in choosing a course  6531 non-null   object 
 16  Search                                         9240 non-null   object 
 17  Magazine                                       9240 non-null   object 
 18  Newspaper Article                              9240 non-null   object 
 19  X Education Forums                             9240 non-null   object 
 20  Newspaper                                      9240 non-null   object 
 21  Digital Advertisement                          9240 non-null   object 
 22  Through Recommendations                        9240 non-null   object 
 23  Receive More Updates About Our Courses         9240 non-null   object 
 24  Tags                                           5887 non-null   object 
 25  Lead Quality                                   4473 non-null   object 
 26  Update me on Supply Chain Content              9240 non-null   object 
 27  Get updates on DM Content                      9240 non-null   object 
 28  Lead Profile                                   6531 non-null   object 
 29  City                                           7820 non-null   object 
 30  Asymmetrique Activity Index                    5022 non-null   object 
 31  Asymmetrique Profile Index                     5022 non-null   object 
 32  Asymmetrique Activity Score                    5022 non-null   float64
```

```
 33  Asymmetrique Profile Score                     5022 non-null   float64
 34  I agree to pay the amount through cheque       9240 non-null   object
 35  A free copy of Mastering The Interview         9240 non-null   object
 36  Last Notable Activity                          9240 non-null   object
dtypes: float64(4), int64(3), object(30)
memory usage: 2.6+ MB
```

Data Preparation

Since Prospect ID is a unique ID with which the customer is identified and Lead Number implies a number assigned to each lead procured, we will check whether these are unique and not-null values.

In [ ]:
```
#print('Prospect ID: ', sum(source_df.duplicated(subset='Prospect ID')))
#print('Lead Number: ', sum(source_df.duplicated(subset='Prospect ID')))
```

In [4]:
```
source_df['Prospect ID'].value_counts(ascending=False)
```

Out[4]:
```
7927b2df-8bba-4d29-b9a2-b6e0beafe620    1
22e9d4ef-d294-4ebf-81c7-7c7a1105aeea    1
46befc49-253a-419b-abea-2fd978d2e2b1    1
9d35a2c2-09d8-439f-9875-0e8bbf267f5a    1
f0de9371-4dc2-48c2-9785-a08d6fc4fcb5    1
                                       ..
ff1f7582-cb7b-4b94-9cdc-3d0d0afdd9a3    1
644099a2-3da4-4d23-9546-7676340a372b    1
2a093175-415b-4321-9e69-ed8d9df65a3c    1
c66249a3-8500-4c66-a511-312d914573de    1
571b5c8e-a5b2-4d57-8574-f2ffb06fdeff    1
Name: Prospect ID, Length: 9240, dtype: int64
```

In [5]:
```
source_df['Lead Number'].value_counts(ascending=False)
```

Out[5]:
```
660737    1
603303    1
602561    1
602557    1
602540    1
         ..
630422    1
630405    1
630403    1
630390    1
579533    1
Name: Lead Number, Length: 9240, dtype: int64
```

In [6]:
```
source_df[['Prospect ID', 'Lead Number']].isnull().sum()
```

Out[6]:
```
Prospect ID    0
Lead Number    0
dtype: int64
```

Prospect ID and Lead number are unique and not null which means each row implies a different customer and so these can be dropped.

In [7]:
```
source_df.drop(['Prospect ID', 'Lead Number'], 1, inplace = True)
```

Check missing values

In [8]:
```
#Most of the categorical variables have a level called 'Select'
# This Value is to be treated as this is similar to Null value
source_df = source_df.replace('Select', np.nan)
```

In [9]:
```
round((100*source_df.isnull().sum()/source_df.shape[0]),2).sort_values(ascending=False)
```

Out[9]:
```
How did you hear about X Education               78.46
Lead Profile                                     74.19
Lead Quality                                     51.59
Asymmetrique Activity Score                      45.65
Asymmetrique Profile Score                       45.65
Asymmetrique Profile Index                       45.65
Asymmetrique Activity Index                      45.65
City                                             39.71
Specialization                                   36.58
Tags                                             36.29
What matters most to you in choosing a course    29.32
What is your current occupation                  29.11
Country                                          26.63
TotalVisits                                       1.48
Page Views Per Visit                              1.48
Last Activity                                     1.11
Lead Source                                       0.39
Get updates on DM Content                         0.00
Update me on Supply Chain Content                 0.00
I agree to pay the amount through cheque          0.00
A free copy of Mastering The Interview            0.00
Lead Origin                                       0.00
X Education Forums                                0.00
Receive More Updates About Our Courses            0.00
Through Recommendations                           0.00
Digital Advertisement                             0.00
Newspaper                                         0.00
Newspaper Article                                 0.00
Magazine                                          0.00
Search                                            0.00
Total Time Spent on Website                       0.00
Converted                                         0.00
Do Not Call                                       0.00
Do Not Email                                      0.00
Last Notable Activity                             0.00
dtype: float64
```

In [10]:
```
#Removing variables with more than 30% missing values
columns_to_drop=source_df.loc[:,list(round((100*source_df.isnull().sum()/source_df.shap
columns_to_drop
```

Out[10]:
```
Index(['Specialization', 'How did you hear about X Education', 'Tags',
       'Lead Quality', 'Lead Profile', 'City', 'Asymmetrique Activity Index',
       'Asymmetrique Profile Index', 'Asymmetrique Activity Score',
       'Asymmetrique Profile Score'],
      dtype='object')
```

In [11]:
```
source_df=source_df.drop(columns_to_drop,axis=1)
```

In [12]:
```
round((100*source_df.isnull().sum()/source_df.shape[0]),2).sort_values(ascending=False)
```

Out[12]:
```
What matters most to you in choosing a course    29.32
What is your current occupation                  29.11
Country                                          26.63
TotalVisits                                       1.48
Page Views Per Visit                              1.48
Last Activity                                     1.11
Lead Source                                       0.39
Lead Origin                                       0.00
Newspaper                                         0.00
A free copy of Mastering The Interview            0.00
I agree to pay the amount through cheque          0.00
Get updates on DM Content                         0.00
Update me on Supply Chain Content                 0.00
Receive More Updates About Our Courses            0.00
Through Recommendations                           0.00
Digital Advertisement                             0.00
Search                                            0.00
X Education Forums                                0.00
Newspaper Article                                 0.00
Magazine                                          0.00
Total Time Spent on Website                       0.00
Converted                                         0.00
Do Not Call                                       0.00
Do Not Email                                      0.00
Last Notable Activity                             0.00
dtype: float64
```

Imputing missing values

In [13]:
```
source_df['What matters most to you in choosing a course'].value_counts(normalize=True,
```

Out[13]:
```
Better Career Prospects      70.649351
NaN                          29.318182
Flexibility & Convenience     0.021645
Other                         0.010823
Name: What matters most to you in choosing a course, dtype: float64
```

'What matters most to you in choosing a course' has dominating category in 'Better Career Prospects'. Imputing null values with mode would cause the variable to be almost constant within the dataset. So dropping the field.

In [14]:
```
source_df.drop('What matters most to you in choosing a course', axis = 1, inplace = Tru
```

In [15]:
```
source_df['What is your current occupation'].value_counts(normalize=True,dropna=False)*
```

Out[15]:
```
Unemployed              60.606061
NaN                     29.112554
Working Professional     7.640693
Student                  2.272727
Other                    0.173160
```

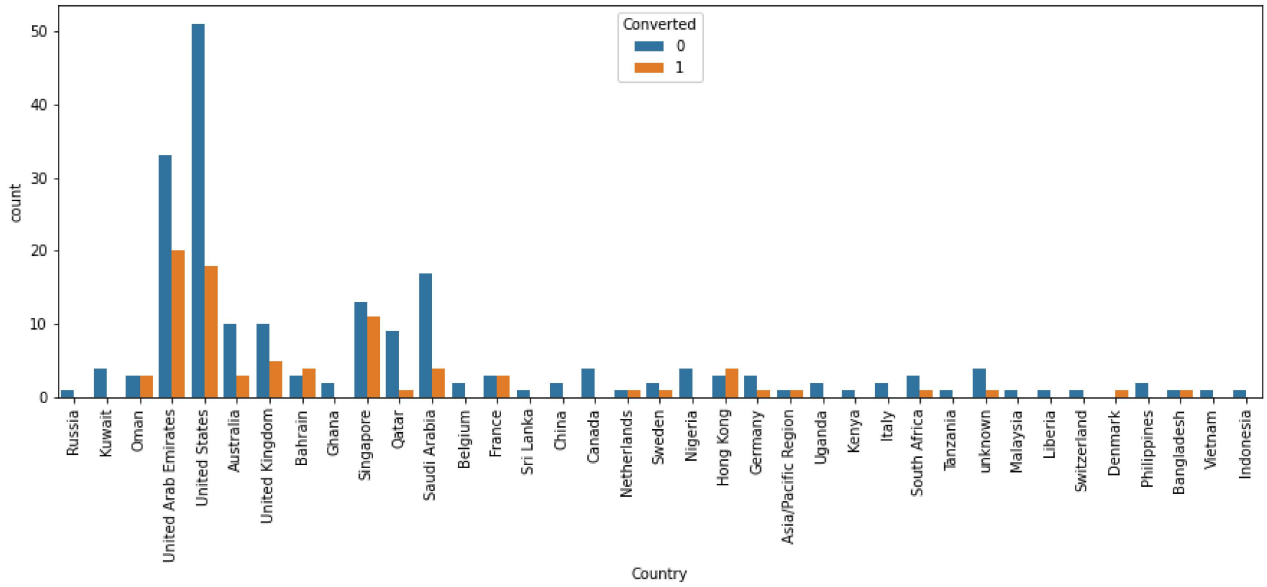
```
Housewife                 0.108225
Businessman               0.086580
Name: What is your current occupation, dtype: float64
```

In [17]:
```
#Imputing with mode
source_df['What is your current occupation'] = source_df['What is your current occupati
```

In [18]:
```
source_df['Country'].value_counts(normalize=True,dropna=False)*100
```

Out[18]:
```
India                   70.259740
NaN                     26.634199
United States            0.746753
United Arab Emirates     0.573593
Singapore                0.259740
Saudi Arabia             0.227273
United Kingdom           0.162338
Australia                0.140693
Qatar                    0.108225
Bahrain                  0.075758
Hong Kong                0.075758
Oman                     0.064935
France                   0.064935
unknown                  0.054113
Kuwait                   0.043290
South Africa             0.043290
Canada                   0.043290
Nigeria                  0.043290
Germany                  0.043290
Sweden                   0.032468
Philippines              0.021645
Uganda                   0.021645
Italy                    0.021645
Bangladesh               0.021645
Netherlands              0.021645
Asia/Pacific Region      0.021645
China                    0.021645
Belgium                  0.021645
Ghana                    0.021645
Kenya                    0.010823
Sri Lanka                0.010823
Tanzania                 0.010823
Malaysia                 0.010823
Liberia                  0.010823
Switzerland              0.010823
Denmark                  0.010823
Russia                   0.010823
Vietnam                  0.010823
Indonesia                0.010823
Name: Country, dtype: float64
```

In [19]:
```
plt.figure(figsize=(15,5))
sns.countplot(source_df[source_df['Country']!='India']['Country'], hue=source_df['Conve
plt.xticks(rotation = 90)
plt.show()
```



There seems to be correlation with country where countries like Oman, Hong Kong, France etc seems top have high conversion rate.

In [20]:
```
#Imputing with mode
source_df['Country'] = source_df['Country'].replace(np.nan, source_df['Country'].mode()
```

In [21]:
```
source_df['TotalVisits'].value_counts(normalize=True,dropna=False)
```

Out[21]:
```
0.0      0.236905
2.0      0.181818
3.0      0.141342
4.0      0.121212
5.0      0.084740
6.0      0.050433
1.0      0.042749
7.0      0.033442
8.0      0.024242
9.0      0.017749
NaN      0.014827
10.0     0.012338
11.0     0.009307
13.0     0.005195
12.0     0.004870
14.0     0.003896
16.0     0.002273
15.0     0.001948
17.0     0.001732
18.0     0.001623
20.0     0.001299
19.0     0.000974
21.0     0.000649
23.0     0.000649
24.0     0.000541
25.0     0.000541
27.0     0.000541
22.0     0.000325
29.0     0.000216
28.0     0.000216
```

```
26.0     0.000216
141.0    0.000108
55.0     0.000108
30.0     0.000108
43.0     0.000108
74.0     0.000108
41.0     0.000108
54.0     0.000108
115.0    0.000108
251.0    0.000108
32.0     0.000108
42.0     0.000108
Name: TotalVisits, dtype: float64
```

In [22]:
```
#Imputing with median
source_df['TotalVisits'].fillna(source_df['TotalVisits'].median(), inplace=True)
```

In [23]:
```
source_df['Page Views Per Visit'].value_counts(normalize=True,dropna=False)
```

Out[23]:
```
0.00    0.236905
2.00    0.194264
3.00    0.129437
4.00    0.096970
1.00    0.070455
          ...
2.56    0.000108
6.33    0.000108
1.64    0.000108
8.21    0.000108
2.08    0.000108
Name: Page Views Per Visit, Length: 115, dtype: float64
```

In [24]:
```
#Imputing with median
source_df['Page Views Per Visit'].fillna(source_df['Page Views Per Visit'].median(), in
```

In [25]:
```
source_df['Last Activity'].value_counts(normalize=True,dropna=False)
```

Out[25]:
```
Email Opened                    0.371970
SMS Sent                        0.297078
Olark Chat Conversation         0.105303
Page Visited on Website         0.069264
Converted to Lead               0.046320
Email Bounced                   0.035281
Email Link Clicked              0.028896
Form Submitted on Website       0.012554
NaN                             0.011147
Unreachable                     0.010065
Unsubscribed                    0.006602
Had a Phone Conversation        0.003247
Approached upfront              0.000974
View in browser link Clicked    0.000649
Email Received                  0.000216
Email Marked Spam               0.000216
Visited Booth in Tradeshow      0.000108
Resubscribed to emails          0.000108
Name: Last Activity, dtype: float64
```

In [26]:
```
#Imputing with mode
source_df['Last Activity'] = source_df['Last Activity'].replace(np.nan, source_df['Last
```

In [27]:
```
source_df['Lead Source'].value_counts(normalize=True,dropna=False)
```

Out[27]:
```
Google               0.310390
Direct Traffic       0.275216
Olark Chat           0.189935
Organic Search       0.124892
Reference            0.057792
Welingak Website     0.015368
Referral Sites       0.013528
Facebook             0.005952
NaN                  0.003896
bing                 0.000649
google               0.000541
Click2call           0.000433
Press_Release        0.000216
Social Media         0.000216
Live Chat            0.000216
youtubechannel       0.000108
testone              0.000108
Pay per Click Ads    0.000108
welearnblog_Home     0.000108
WeLearn              0.000108
blog                 0.000108
NC_EDM               0.000108
Name: Lead Source, dtype: float64
```

In [28]:
```
#Imputing with mode
source_df['Lead Source'] = source_df['Lead Source'].replace(np.nan, source_df['Lead Sou
```

In [29]:
```
round((100*source_df.isnull().sum()/source_df.shape[0]),2).sort_values(ascending=False)
```

Out[29]:
```
Lead Origin                                0.0
Lead Source                                0.0
A free copy of Mastering The Interview     0.0
I agree to pay the amount through cheque   0.0
Get updates on DM Content                  0.0
Update me on Supply Chain Content          0.0
Receive More Updates About Our Courses     0.0
Through Recommendations                    0.0
Digital Advertisement                      0.0
Newspaper                                  0.0
X Education Forums                         0.0
Newspaper Article                          0.0
Magazine                                   0.0
Search                                     0.0
What is your current occupation            0.0
Country                                    0.0
Last Activity                              0.0
Page Views Per Visit                       0.0
Total Time Spent on Website                0.0
TotalVisits                                0.0
Converted                                  0.0
```

```
Do Not Call                                    0.0
Do Not Email                                   0.0
Last Notable Activity                          0.0
dtype: float64
```

In [30]:
```
#Removing columns with constant values
source_df = source_df[source_df.loc[:, (source_df != source_df.iloc[0]).any()].columns]
```

In [31]:
```
source_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9240 entries, 0 to 9239
Data columns (total 19 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   Lead Origin                             9240 non-null   object
 1   Lead Source                             9240 non-null   object
 2   Do Not Email                            9240 non-null   object
 3   Do Not Call                             9240 non-null   object
 4   Converted                               9240 non-null   int64
 5   TotalVisits                             9240 non-null   float64
 6   Total Time Spent on Website             9240 non-null   int64
 7   Page Views Per Visit                    9240 non-null   float64
 8   Last Activity                           9240 non-null   object
 9   Country                                 9240 non-null   object
 10  What is your current occupation         9240 non-null   object
 11  Search                                  9240 non-null   object
 12  Newspaper Article                       9240 non-null   object
 13  X Education Forums                      9240 non-null   object
 14  Newspaper                               9240 non-null   object
 15  Digital Advertisement                   9240 non-null   object
 16  Through Recommendations                 9240 non-null   object
 17  A free copy of Mastering The Interview  9240 non-null   object
 18  Last Notable Activity                   9240 non-null   object
dtypes: float64(2), int64(2), object(15)
memory usage: 1.3+ MB
```

In [32]:
```
categorical_columns = source_df.loc[:, source_df.dtypes == 'object'].columns.tolist()
numerical_columns = [d for d in source_df.columns if d not in categorical_columns and d
```

In [33]:
```
for col in categorical_columns:
    print(source_df[col].astype('category').value_counts(normalize=True,dropna=False))
    print('---------------------------------------------------------')
```

```
Landing Page Submission    0.528788
API                        0.387446
Lead Add Form              0.077706
Lead Import                0.005952
Quick Add Form             0.000108
Name: Lead Origin, dtype: float64
---------------------------------------------------------
Google               0.314286
Direct Traffic       0.275216
Olark Chat           0.189935
Organic Search       0.124892
Reference            0.057792
```

```
Welingak Website     0.015368
Referral Sites       0.013528
Facebook             0.005952
bing                 0.000649
google               0.000541
Click2call           0.000433
Press_Release        0.000216
Social Media         0.000216
Live Chat            0.000216
WeLearn              0.000108
Pay per Click Ads    0.000108
NC_EDM               0.000108
blog                 0.000108
testone              0.000108
welearnblog_Home     0.000108
youtubechannel       0.000108
Name: Lead Source, dtype: float64
----------------------------------------------------
No     0.920563
Yes    0.079437
Name: Do Not Email, dtype: float64
----------------------------------------------------
No     0.999784
Yes    0.000216
Name: Do Not Call, dtype: float64
----------------------------------------------------
Email Opened                    0.383117
SMS Sent                        0.297078
Olark Chat Conversation         0.105303
Page Visited on Website         0.069264
Converted to Lead               0.046320
Email Bounced                   0.035281
Email Link Clicked              0.028896
Form Submitted on Website       0.012554
Unreachable                     0.010065
Unsubscribed                    0.006602
Had a Phone Conversation        0.003247
Approached upfront              0.000974
View in browser link Clicked    0.000649
Email Received                  0.000216
Email Marked Spam               0.000216
Resubscribed to emails          0.000108
Visited Booth in Tradeshow      0.000108
Name: Last Activity, dtype: float64
----------------------------------------------------
India                  0.968939
United States          0.007468
United Arab Emirates   0.005736
Singapore              0.002597
Saudi Arabia           0.002273
United Kingdom         0.001623
Australia              0.001407
Qatar                  0.001082
Bahrain                0.000758
Hong Kong              0.000758
France                 0.000649
Oman                   0.000649
unknown                0.000541
Kuwait                 0.000433
Nigeria                0.000433
```

```
South Africa            0.000433
Germany                 0.000433
Canada                  0.000433
Sweden                  0.000325
Uganda                  0.000216
Philippines             0.000216
Asia/Pacific Region     0.000216
Italy                   0.000216
Ghana                   0.000216
China                   0.000216
Belgium                 0.000216
Bangladesh              0.000216
Netherlands             0.000216
Malaysia                0.000108
Liberia                 0.000108
Russia                  0.000108
Kenya                   0.000108
Indonesia               0.000108
Sri Lanka               0.000108
Switzerland             0.000108
Tanzania                0.000108
Denmark                 0.000108
Vietnam                 0.000108
Name: Country, dtype: float64
----------------------------------------------------
Unemployed              0.897186
Working Professional    0.076407
Student                 0.022727
Other                   0.001732
Housewife               0.001082
Businessman             0.000866
Name: What is your current occupation, dtype: float64
----------------------------------------------------
No     0.998485
Yes    0.001515
Name: Search, dtype: float64
----------------------------------------------------
No     0.999784
Yes    0.000216
Name: Newspaper Article, dtype: float64
----------------------------------------------------
No     0.999892
Yes    0.000108
Name: X Education Forums, dtype: float64
----------------------------------------------------
No     0.999892
Yes    0.000108
Name: Newspaper, dtype: float64
----------------------------------------------------
No     0.999567
Yes    0.000433
Name: Digital Advertisement, dtype: float64
----------------------------------------------------
No     0.999242
Yes    0.000758
Name: Through Recommendations, dtype: float64
----------------------------------------------------
No     0.687446
Yes    0.312554
Name: A free copy of Mastering The Interview, dtype: float64
```

```
-----------------------------------------------------
Modified                        0.368723
Email Opened                    0.305952
SMS Sent                        0.235065
Page Visited on Website         0.034416
Olark Chat Conversation         0.019805
Email Link Clicked              0.018723
Email Bounced                   0.006494
Unsubscribed                    0.005087
Unreachable                     0.003463
Had a Phone Conversation        0.001515
Email Marked Spam               0.000216
Approached upfront              0.000108
Email Received                  0.000108
Form Submitted on Website       0.000108
Resubscribed to emails          0.000108
View in browser link Clicked    0.000108
Name: Last Notable Activity, dtype: float64
-----------------------------------------------------
```

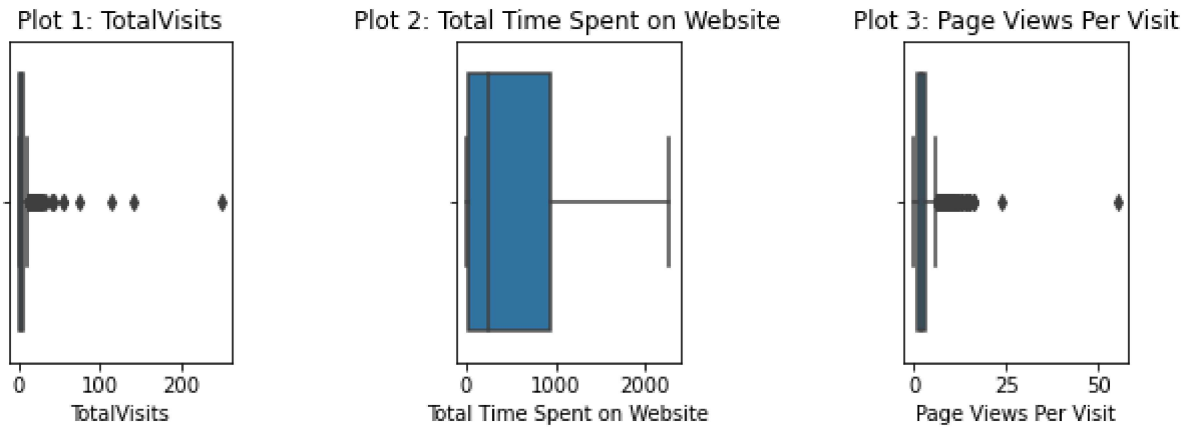
From the above analysis, columns 'Do Not Call', 'Newspaper Article', 'X Education Forums', 'Search', 'Newspaper', 'Digital Advertisement'and 'Through Recommendations' has more than 99% of data points having same category. So removing them.

In [34]:
```
source_df = source_df.drop(['Do Not Call', 'Newspaper Article',
                            'X Education Forums', 'Search', 'Newspaper', 'Digita
```

Outlier treatment for Numerical columns

In [35]:
```
preprocessed_dataframe = source_df.copy(deep=True)
```

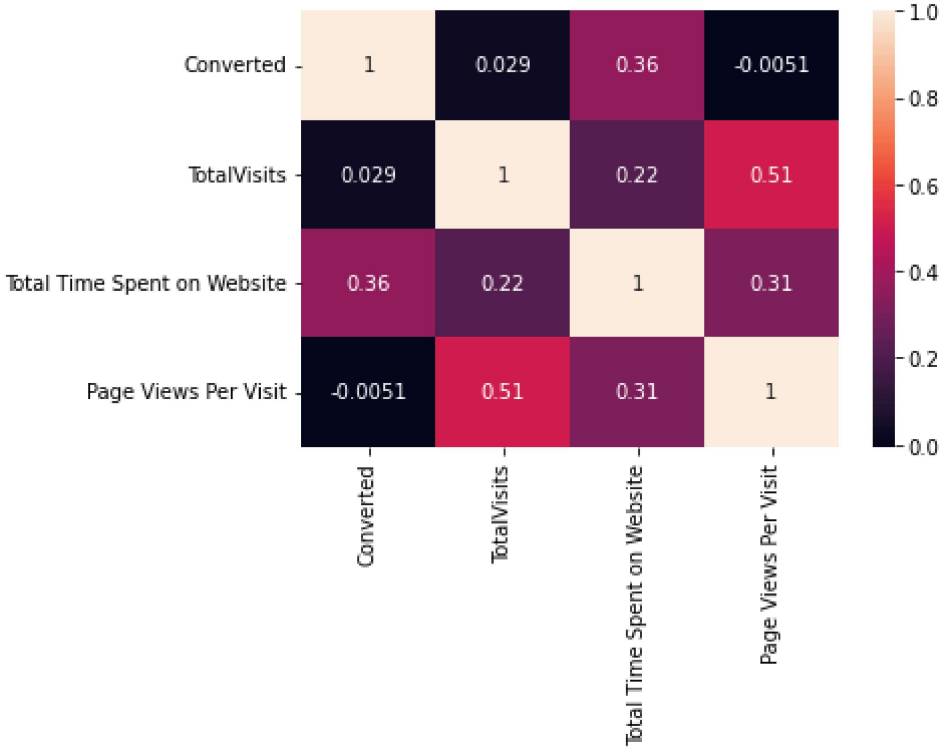
In [36]:
```
plt.figure(figsize=(10,10))

for index, col in enumerate(numerical_columns):
    plt.subplot(3,3,index+1)
    sns.boxplot(preprocessed_dataframe[col])
    plt.title("Plot "+str(index+1)+": "+col)
plt.subplots_adjust(wspace=1,hspace=0.2)

plt.show()
```



Need to remove the outliers from 'TotalVisits' and 'Page Views Per Visit' as per the box plot.

Data Analysis

In [37]:
```
sns.heatmap(preprocessed_dataframe.corr(),annot=True)
plt.show()
```



'TotalVisits' and 'Page Views Per Visit' are highly correlated and removing the columns. 'Total Time Spent on Website' is not having any outlier.

In [38]:
```
preprocessed_dataframe.drop(['TotalVisits', 'Page Views Per Visit'], axis = 1, inplace
```

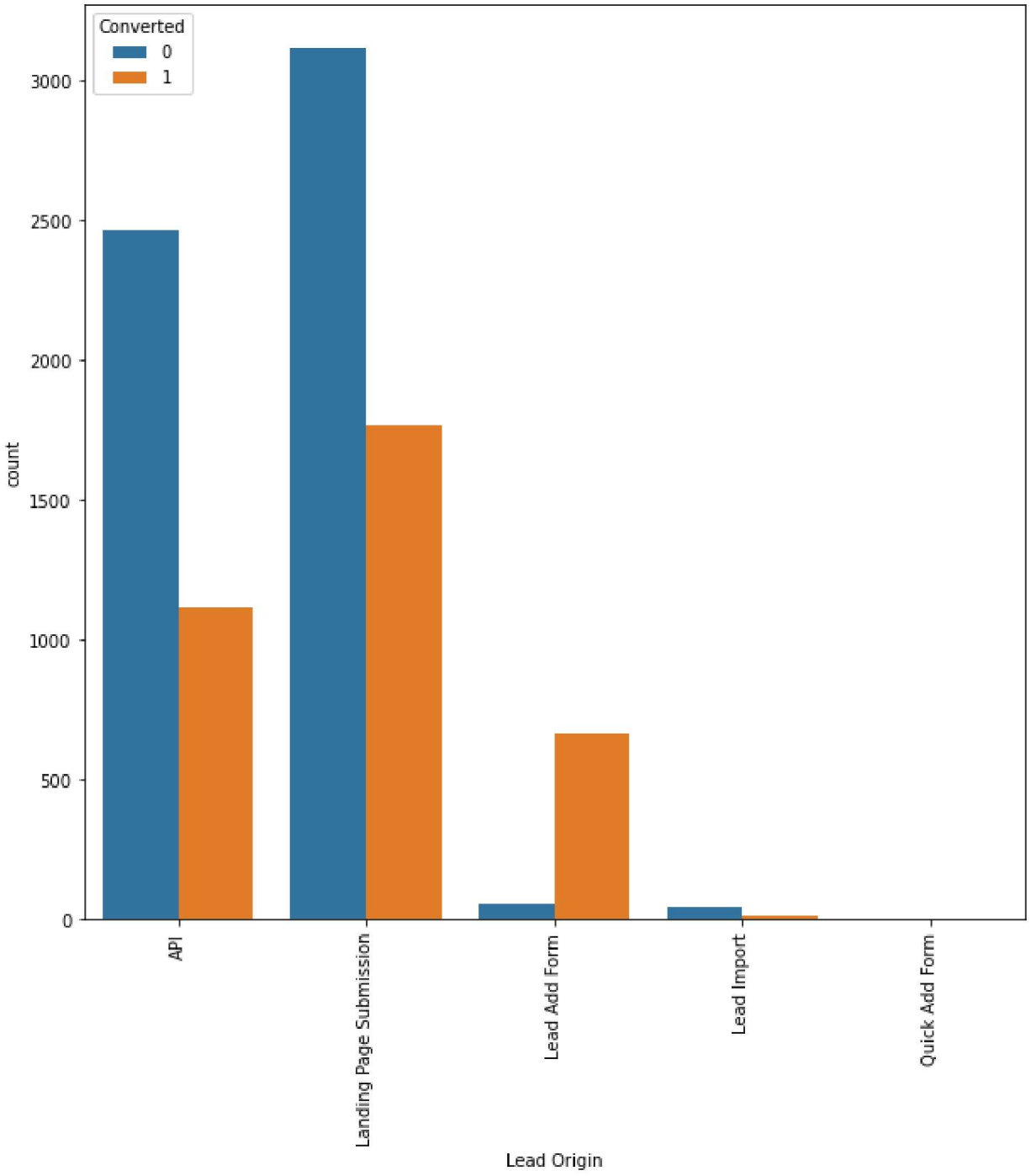
In [39]:
```
numerical_columns = ['Total Time Spent on Website']
```

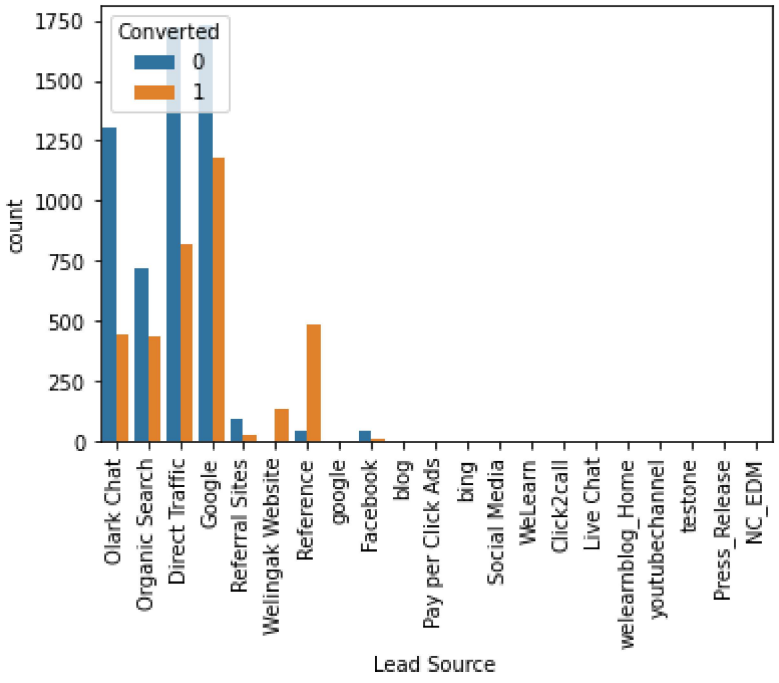
In [40]:
```
categorical_columns = preprocessed_dataframe.loc[:, preprocessed_dataframe.dtypes == 'o
categorical_columns
```

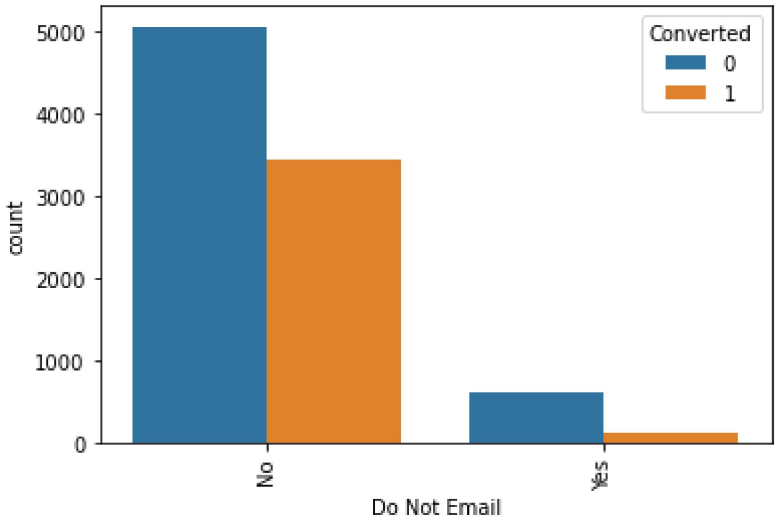
Out[40]:
```
['Lead Origin',
 'Lead Source',
 'Do Not Email',
 'Last Activity',
 'Country',
 'What is your current occupation',
 'A free copy of Mastering The Interview',
 'Last Notable Activity']
```

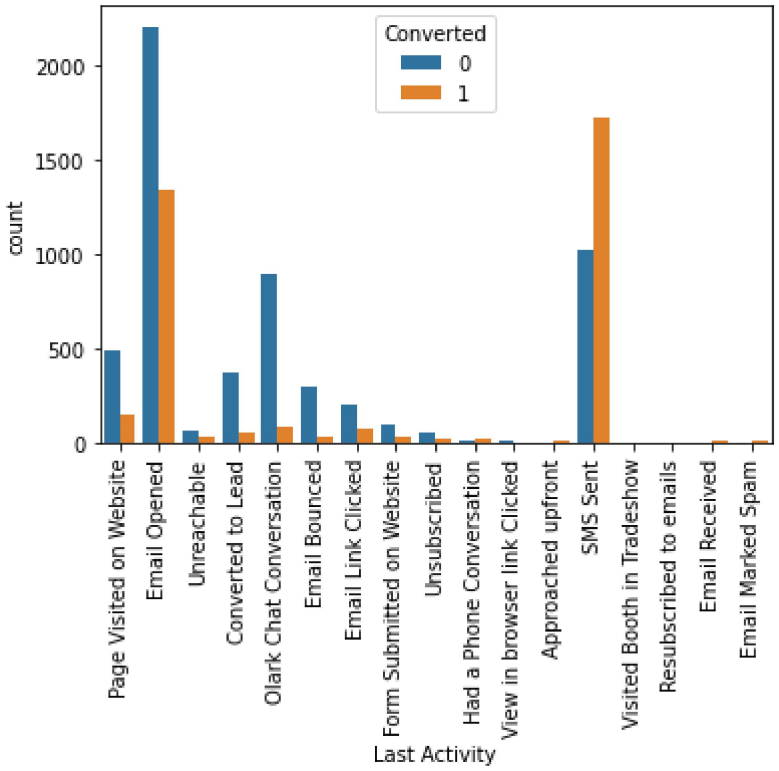
In [41]:
```
plt.figure(figsize=(10,10))
for col in categorical_columns:
    sns.countplot(preprocessed_dataframe[col], hue=preprocessed_dataframe['Converted'])
```

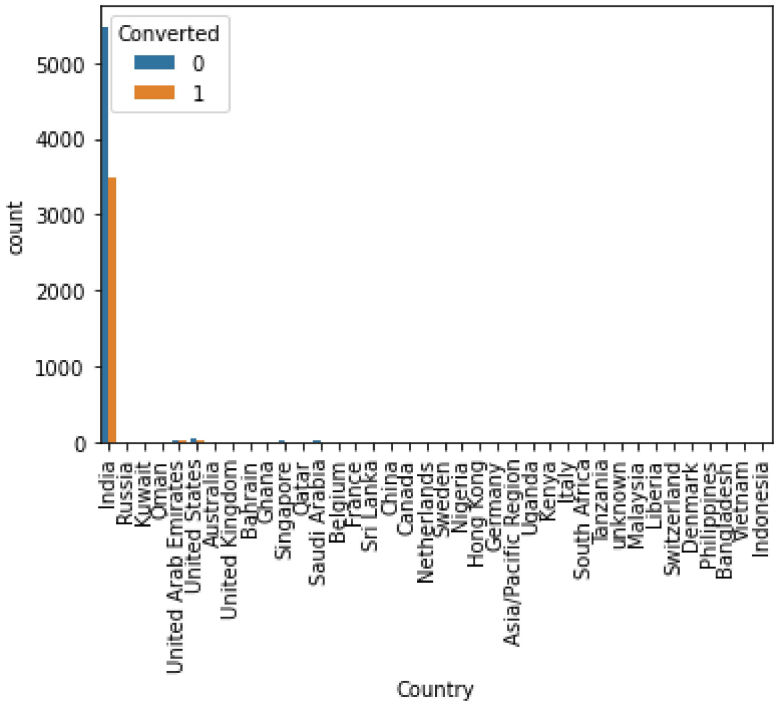
```
    plt.xticks(rotation = 90)
    plt.show()
```

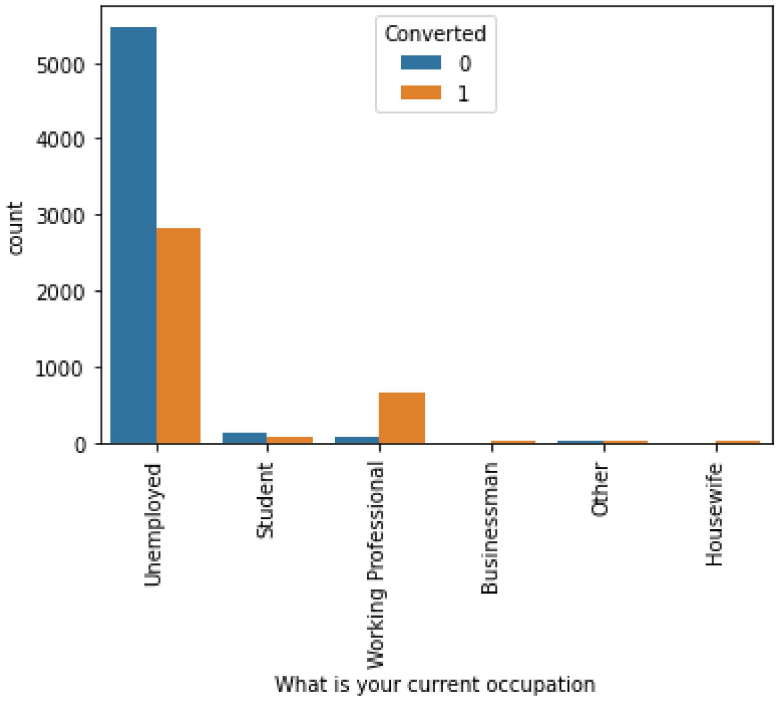


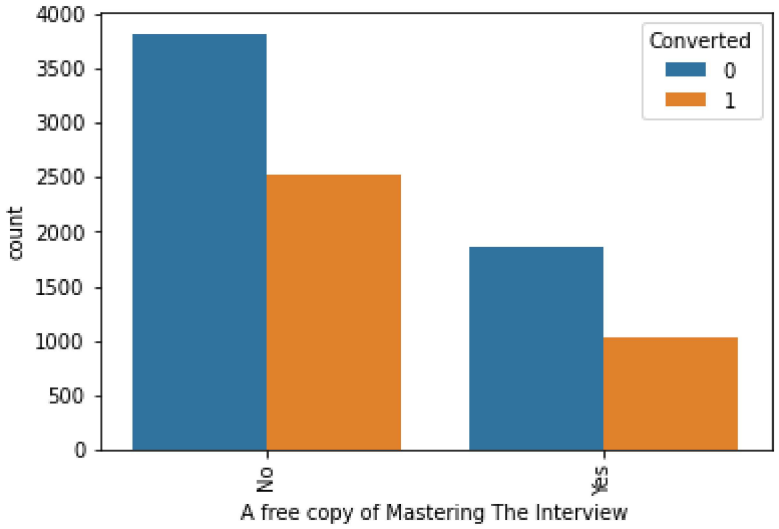

Converted
0
1
count
1750
1500
1250
1000
750
500
250
0
Olark Chat
Organic Search
Direct Traffic
Google
Referral Sites
Welingak Website
Reference
google
Facebook
blog
Pay per Click Ads
bing
Social Media
WeLearn
Click2call
Live Chat
welearnblog_Home
youtubechannel
testone
Press_Release
NC_EDM
Lead Source

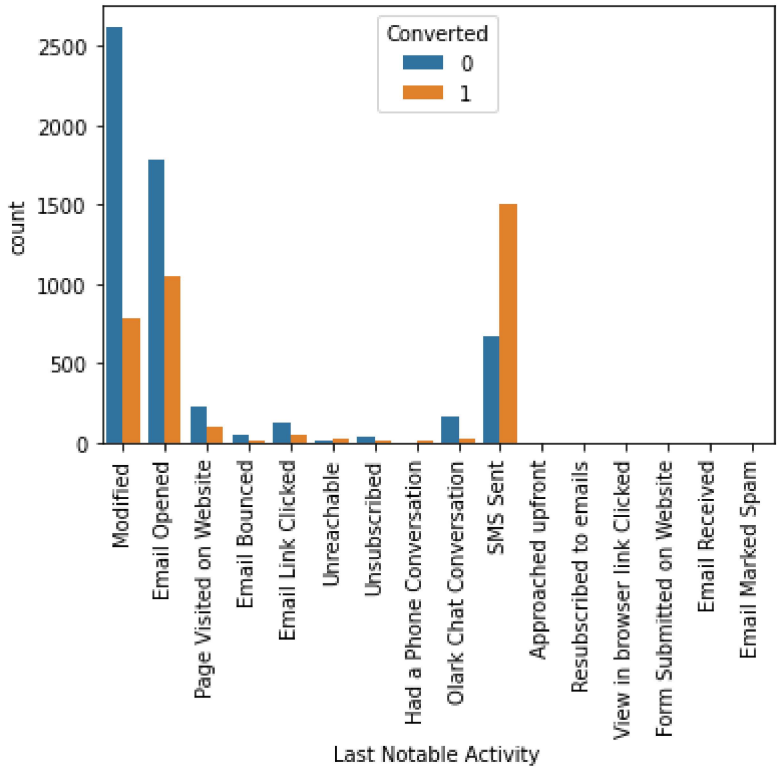

Converted
0
1
count
5000
4000
3000
2000
1000
0
No
Yes
Do Not Email


Converted
0
1
count
2000
1500
1000
500
0
Page Visited on Website
Email Opened
Unreachable
Converted to Lead
Olark Chat Conversation
Email Bounced
Email Link Clicked
Form Submitted on Website
Unsubscribed
Had a Phone Conversation
View in browser link Clicked
Approached upfront
SMS Sent
Visited Booth in Tradeshow
Resubscribed to emails
Email Received
Email Marked Spam
Last Activity


Converted
0
1
count
5000
4000
3000
2000
1000
0
India
Russia
Kuwait
Oman
United Arab Emirates
United States
Australia
United Kingdom
Bahrain
Ghana
Singapore
Qatar
Saudi Arabia
Belgium
France
Sri Lanka
China
Canada
Netherlands
Sweden
Nigeria
Hong Kong
Germany
Asia/Pacific Region
Uganda
Kenya
Italy
South Africa
Tanzania
unknown
Malaysia
Liberia
Switzerland
Denmark
Philippines
Bangladesh
Vietnam
Indonesia
Country


Converted
0
1
count
5000
4000
3000
2000
1000
0
Unemployed
Student
Working Professional
Businessman
Other
Housewife
What is your current occupation


Converted
0
1
count
4000
3500
3000
2500
2000
1500
1000
500
0
No
Yes
A free copy of Mastering The Interview



In [42]:
```
for col in categorical_columns:
    print('\n\n'+col+'\n')
    print('Segment ratio of converted:not converted')
    print(preprocessed_dataframe[preprocessed_dataframe['Converted']==0][col].value_cou
```

```
Lead Origin

Segment ratio of converted:not converted
API                          2.210762
Landing Page Submission      1.763575
Lead Add Form                0.081325
Lead Import                  3.230769
Quick Add Form                    NaN
Name: Lead Origin, dtype: float64


Lead Source

Segment ratio of converted:not converted
Click2call             0.333333
Direct Traffic         2.108802
Facebook               3.230769
Google                 1.469388
Live Chat                   NaN
NC_EDM                      NaN
Olark Chat             2.917411
Organic Search         1.646789
Pay per Click Ads           NaN
Press_Release               NaN
Reference              0.089796
Referral Sites         3.032258
Social Media           1.000000
```

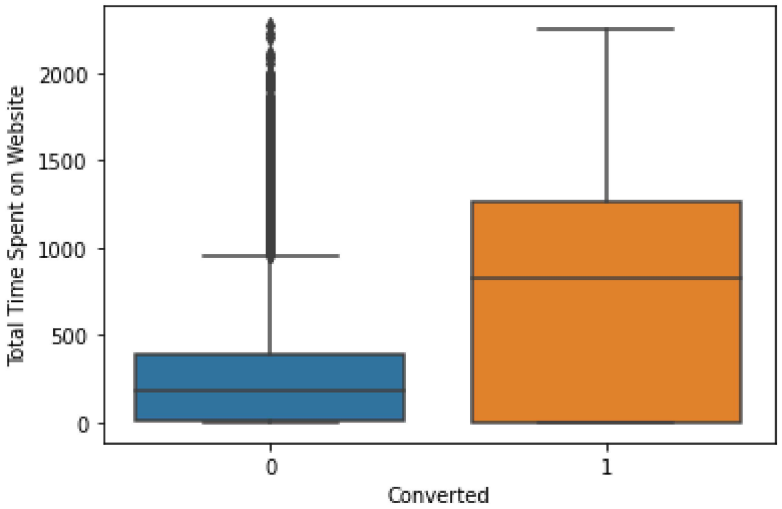
```
WeLearn                   NaN
Welingak Website     0.014286
bing                 5.000000
blog                      NaN
google                    NaN
testone                   NaN
welearnblog_Home          NaN
youtubechannel            NaN
Name: Lead Source, dtype: float64


Do Not Email

Segment ratio of converted:not converted
No     1.470520
Yes    5.220339
Name: Do Not Email, dtype: float64


Last Activity

Segment ratio of converted:not converted
Approached upfront                   NaN
Converted to Lead               6.925926
Email Bounced                  11.538462
Email Link Clicked              2.657534
Email Marked Spam                    NaN
Email Opened                    1.653673
Email Received                       NaN
Form Submitted on Website       3.142857
Had a Phone Conversation        0.363636
Olark Chat Conversation        10.583333
Page Visited on Website         3.238411
Resubscribed to emails               NaN
SMS Sent                        0.589461
Unreachable                     2.000000
Unsubscribed                    2.812500
View in browser link Clicked    5.000000
Visited Booth in Tradeshow           NaN
Name: Last Activity, dtype: float64


Country

Segment ratio of converted:not converted
Asia/Pacific Region     1.000000
Australia               3.333333
Bahrain                 0.750000
Bangladesh              1.000000
Belgium                      NaN
Canada                       NaN
China                        NaN
Denmark                      NaN
France                  1.000000
Germany                 3.000000
Ghana                        NaN
Hong Kong               0.750000
India                   1.574921
Indonesia                    NaN
Italy                        NaN
```

```
Kenya                        NaN
Kuwait                       NaN
Liberia                      NaN
Malaysia                     NaN
Netherlands             1.000000
Nigeria                      NaN
Oman                    1.000000
Philippines                  NaN
Qatar                   9.000000
Russia                       NaN
Saudi Arabia            4.250000
Singapore               1.181818
South Africa            3.000000
Sri Lanka                    NaN
Sweden                  2.000000
Switzerland                  NaN
Tanzania                     NaN
Uganda                       NaN
United Arab Emirates    1.650000
United Kingdom          2.000000
United States           2.833333
Vietnam                      NaN
unknown                 4.000000
Name: Country, dtype: float64


What is your current occupation

Segment ratio of converted:not converted
Businessman             0.600000
Housewife                    NaN
Other                   0.600000
Student                 1.692308
Unemployed              1.949128
Working Professional    0.091190
Name: What is your current occupation, dtype: float64


A free copy of Mastering The Interview

Segment ratio of converted:not converted
No     1.509680
Yes    1.803883
Name: A free copy of Mastering The Interview, dtype: float64


Last Notable Activity

Segment ratio of converted:not converted
Approached upfront                   NaN
Email Bounced                   5.666667
Email Link Clicked              2.844444
Email Marked Spam                    NaN
Email Opened                    1.707854
Email Received                       NaN
Form Submitted on Website            NaN
Had a Phone Conversation        0.076923
Modified                        3.351213
Olark Chat Conversation         6.320000
Page Visited on Website         2.419355
```

```
Resubscribed to emails              NaN
SMS Sent                       0.440318
Unreachable                    0.454545
Unsubscribed                   2.357143
View in browser link Clicked        NaN
Name: Last Notable Activity, dtype: float64
```

1. API and Landing Page submission brings most of the leads whereas the conversion rate is higher for Lead Add Form.
2. Lead Import and Quick Add Form brings very few leads and Quick Add Form is having a zero conversion rate.
3. In lead source, Direct Traffic and Olark Chat brings in huge number of leads but suffers very low conversion whereas Google has good lead inputs and decent conversion.
4. Reference shows the highest conversion rate.
5. In Last Activity, Had a Phone Conversation and SMS sent seems to generate hot leads having good conversion rate.
6. Unemployed people seems to be making up for most of the leads but with low conversion of about half.
7. Businessman and Working Professional contribute for higher conversions.
8. Housewives are having less lead generation percentage, but the generated leads tend to be converted.

In [43]:
```
sns.boxplot(x=preprocessed_dataframe.Converted, y=preprocessed_dataframe['Total Time Sp
plt.show()
```



Leads spending more time in website has higher probability to be converted.

Categorical encoding

In [44]:
```
dummy_df = pd.DataFrame()
for col in categorical_columns:
    dummy_df = pd.concat([dummy_df, pd.get_dummies(preprocessed_dataframe[col], drop_fi
```

In [45]:
```
preprocessed_dataframe.drop(columns=categorical_columns,axis=1,inplace =True)
```

In [46]:
```
preprocessed_dataframe = pd.concat([preprocessed_dataframe, dummy_df], axis = 1)
```

Split and normalization

In [47]:
```
label = 'Converted'
```

In [48]:
```
train_df, test_df = train_test_split(preprocessed_dataframe, train_size = 0.70, test_si
```

In [49]:
```
scaler = StandardScaler()
```

In [50]:
```
train_df[numerical_columns] = scaler.fit_transform(train_df[numerical_columns])
```

Model Building

In [51]:
```
## First split independent and dependent variables
feature_list = train_df.columns.tolist()
feature_list.remove(label)
feature_train_df = train_df[feature_list]
label_train_df = train_df[[label]]
```

In [52]:
```
logreg=LogisticRegression()
rfe=RFE(logreg,20).fit(feature_train_df,label_train_df)
#list(zip(feature_train_df.columns, rfe.support_, rfe.ranking_))
```

In [53]:
```
rfe_feature_list=list(feature_train_df.columns[rfe.support_])
rfe_feature_list
```

Out[53]:
```
['Total Time Spent on Website',
 'Lead Origin_Lead Add Form',
 'Lead Source_Olark Chat',
 'Lead Source_Welingak Website',
 'Do Not Email_Yes',
 'Last Activity_Converted to Lead',
 'Last Activity_Email Bounced',
 'Last Activity_Olark Chat Conversation',
 'Last Activity_Page Visited on Website',
 'Country_Italy',
 'Country_Nigeria',
 'What is your current occupation_Housewife',
 'What is your current occupation_Student',
 'What is your current occupation_Unemployed',
 'What is your current occupation_Working Professional',
 'Last Notable Activity_Email Link Clicked',
 'Last Notable Activity_Email Opened',
 'Last Notable Activity_Had a Phone Conversation',
 'Last Notable Activity_Modified',
 'Last Notable Activity_Olark Chat Conversation']
```

In [54]:
```
#function to calculate VIF
def compute_vif(dataframe_to_check):
    vif_data = pd.DataFrame()
    vif_data['variable'] = dataframe_to_check.columns
    vif_data['vif'] = [variance_inflation_factor(dataframe_to_check.values,
                        d) for d in range(dataframe_to_check.shape[1])]
    return vif_data
```

In [55]:
```
feature_train_df_sm = sm.add_constant(feature_train_df[rfe_feature_list])
log_model = sm.GLM(label_train_df,feature_train_df_sm,family=sm.families.Binomial())
log_model = log_model.fit()
log_model.summary()
```

Out[55]:

Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	6468
Model:	GLM	**Df Residuals:**	6447
Model Family:	Binomial	**Df Model:**	20
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2664.1
Date:	Mon, 15 Apr 2024	**Deviance:**	5328.2
Time:	16:01:55	**Pearson chi2:**	7.54e+03
No. Iterations:	23	**Pseudo R-squ. (CS):**	0.3993
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	1.0995	0.635	1.732	0.083	-0.145	2.344
Total Time Spent on Website	1.1576	0.041	28.429	0.000	1.078	1.237
Lead Origin_Lead Add Form	3.7369	0.200	18.721	0.000	3.346	4.128
Lead Source_Olark Chat	1.3066	0.103	12.654	0.000	1.104	1.509
Lead Source_Welingak Website	22.5732	1.29e+04	0.002	0.999	-2.53e+04	2.53e+04
Do Not Email_Yes	-1.2959	0.191	-6.797	0.000	-1.670	-0.922
Last Activity_Converted to Lead	-0.9850	0.218	-4.521	0.000	-1.412	-0.558
Last Activity_Email Bounced	-1.0622	0.341	-3.111	0.002	-1.731	-0.393
Last Activity_Olark Chat Conversation	-1.3986	0.185	-7.546	0.000	-1.762	-1.035
Last Activity_Page Visited on Website	-1.2549	0.157	-8.009	0.000	-1.562	-0.948
Country_Italy	-26.0181	7.4e+04	-0.000	1.000	-1.45e+05	1.45e+05
Country_Nigeria	-24.1643	6.37e+04	-0.000	1.000	-1.25e+05	1.25e+05
What is your current occupation_Housewife	24.0149	4.19e+04	0.001	1.000	-8.2e+04	8.21e+04
What is your current occupation_Student	-0.6375	0.666	-0.957	0.338	-1.942	0.667

What is your current occupation_Unemployed	-1.0347	0.631	-1.640	0.101	-2.271	0.202
What is your current occupation_Working Professional	1.8511	0.658	2.813	0.005	0.561	3.141
Last Notable Activity_Email Link Clicked	-1.8025	0.250	-7.196	0.000	-2.293	-1.312
Last Notable Activity_Email Opened	-1.3327	0.087	-15.356	0.000	-1.503	-1.163
Last Notable Activity_Had a Phone Conversation	2.1357	1.094	1.953	0.051	-0.008	4.279
Last Notable Activity_Modified	-1.4982	0.098	-15.250	0.000	-1.691	-1.306
Last Notable Activity_Olark Chat Conversation	-1.5230	0.381	-4.000	0.000	-2.269	-0.777

Removing 'Lead Source_Welingak Website', 'Country_Italy', 'Country_Nigeria', 'What is your current occupation_Student', 'What is your current occupation_Housewife', 'What is your current occupation_Unemployed', 'Last Notable Activity_Had a Phone Conversation' because of high p-value.

In [56]:
```
rfe_feature_list = [d for d in rfe_feature_list if d not in ['Lead Source_Welingak Webs
        'Country_Nigeria', 'What is your current occupation_Student',
'What is your current occupation_Housewife', 'What is your current occupation_Unemploye
                                            'Last Notable Activity_Had a Phone Conversati
```

In [57]:
```
feature_train_df_sm = sm.add_constant(feature_train_df[rfe_feature_list])
log_model = sm.GLM(label_train_df,feature_train_df_sm,family=sm.families.Binomial())
log_model = log_model.fit()
log_model.summary()
```

Out[57]:

Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	6468
Model:	GLM	**Df Residuals:**	6454
Model Family:	Binomial	**Df Model:**	13
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2690.7
Date:	Mon, 15 Apr 2024	**Deviance:**	5381.3
Time:	16:02:48	**Pearson chi2:**	8.07e+03
No. Iterations:	6	**Pseudo R-squ. (CS):**	0.3944
Covariance Type:	nonrobust		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	0.0792	0.068	1.171	0.241	-0.053	0.212
Total Time Spent on Website	1.1539	0.041	28.459	0.000	1.074	1.233
Lead Origin_Lead Add Form	4.0715	0.195	20.912	0.000	3.690	4.453
Lead Source_Olark Chat	1.2939	0.103	12.577	0.000	1.092	1.496

Do Not Email_Yes	-1.2895	0.189	-6.814	0.000	-1.660	-0.919
Last Activity_Converted to Lead	-0.9888	0.218	-4.546	0.000	-1.415	-0.563
Last Activity_Email Bounced	-1.0947	0.344	-3.187	0.001	-1.768	-0.421
Last Activity_Olark Chat Conversation	-1.3905	0.185	-7.516	0.000	-1.753	-1.028
Last Activity_Page Visited on Website	-1.2526	0.156	-8.009	0.000	-1.559	-0.946
What is your current occupation_Working Professional	2.8560	0.194	14.702	0.000	2.475	3.237
Last Notable Activity_Email Link Clicked	-1.7813	0.247	-7.212	0.000	-2.265	-1.297
Last Notable Activity_Email Opened	-1.3212	0.086	-15.344	0.000	-1.490	-1.152
Last Notable Activity_Modified	-1.4927	0.098	-15.270	0.000	-1.684	-1.301
Last Notable Activity_Olark Chat Conversation	-1.4772	0.373	-3.959	0.000	-2.208	-0.746

In [58]:
```
vif_df = compute_vif(feature_train_df[rfe_feature_list])
```

In [59]:
```
print(vif_df.sort_values('vif',ascending=False))
print(log_model.summary())
```

```
                                          variable       vif
11                    Last Notable Activity_Modified  2.053099
6               Last Activity_Olark Chat Conversation  2.038582
5                         Last Activity_Email Bounced  1.878718
3                                   Do Not Email_Yes  1.844504
2                             Lead Source_Olark Chat  1.676651
12      Last Notable Activity_Olark Chat Conversation  1.337010
4                     Last Activity_Converted to Lead  1.250938
0                         Total Time Spent on Website  1.213874
1                           Lead Origin_Lead Add Form  1.162495
7               Last Activity_Page Visited on Website  1.117900
8   What is your current occupation_Working Profes...  1.116013
10                  Last Notable Activity_Email Opened  1.098823
9             Last Notable Activity_Email Link Clicked  1.017879
                 Generalized Linear Model Regression Results
==============================================================================
Dep. Variable:              Converted   No. Observations:                 6468
Model:                            GLM   Df Residuals:                     6454
Model Family:                Binomial   Df Model:                           13
Link Function:                  Logit   Scale:                          1.0000
Method:                          IRLS   Log-Likelihood:                -2690.7
Date:                Mon, 15 Apr 2024   Deviance:                       5381.3
Time:                        16:03:01   Pearson chi2:                 8.07e+03
No. Iterations:                     6   Pseudo R-squ. (CS):             0.3944
Covariance Type:            nonrobust
==============================================================================================
                                                 coef    std err          z
P>|z|      [0.025      0.975]
----------------------------------------------------------------------------------------------
const                                          0.0792      0.068      1.171
0.241      -0.053       0.212
Total Time Spent on Website                    1.1539      0.041     28.459
```

```
0.000       1.074       1.233
Lead Origin_Lead Add Form                                4.0715      0.195     20.912
0.000       3.690       4.453
Lead Source_Olark Chat                                   1.2939      0.103     12.577
0.000       1.092       1.496
Do Not Email_Yes                                        -1.2895      0.189     -6.814
0.000      -1.660      -0.919
Last Activity_Converted to Lead                         -0.9888      0.218     -4.546
0.000      -1.415      -0.563
Last Activity_Email Bounced                             -1.0947      0.344     -3.187
0.001      -1.768      -0.421
Last Activity_Olark Chat Conversation                   -1.3905      0.185     -7.516
0.000      -1.753      -1.028
Last Activity_Page Visited on Website                   -1.2526      0.156     -8.009
0.000      -1.559      -0.946
What is your current occupation_Working Professional     2.8560      0.194     14.702
0.000       2.475       3.237
Last Notable Activity_Email Link Clicked                -1.7813      0.247     -7.212
0.000      -2.265      -1.297
Last Notable Activity_Email Opened                      -1.3212      0.086    -15.344
0.000      -1.490      -1.152
Last Notable Activity_Modified                          -1.4927      0.098    -15.270
0.000      -1.684      -1.301
Last Notable Activity_Olark Chat Conversation           -1.4772      0.373     -3.959
0.000      -2.208      -0.746
==============================================================================================
================================
```

VIF and p-value seems to be within limit.

In [60]:
```
# Getting the predicted values on the train set
label_train_pred = log_model.predict(feature_train_df_sm)
```

In [61]:
```
label_train_pred_df = pd.DataFrame({'Converted':label_train_df[label].values, 'Predicte
label_train_pred_df
```

Out[61]:

	Converted	Predicted
0	1	0.861417
1	1	0.984790
2	1	0.960643
3	0	0.239815
4	0	0.072815
...	...	...
6463	1	0.536754
6464	0	0.108398
6465	0	0.094912
6466	1	0.934055
6467	0	0.072815

6468 rows × 2 columns

In [62]:
```
label_train_pred_df['Classified'] = label_train_pred_df.Predicted.map(lambda d: 1 if d

label_train_pred_df.head()
```

Out[62]:

	Converted	Predicted	Classified
0	1	0.861417	1
1	1	0.984790	1
2	1	0.960643	1
3	0	0.239815	0
4	0	0.072815	0

In [63]:
```
# Function to get metrics and performance of the model
def model_performance_check(confusion_matrix):
    TN =confusion_matrix[0,0]
    TP =confusion_matrix[1,1]
    FP =confusion_matrix[0,1]
    FN =confusion_matrix[1,0]
    accuracy = (TP+TN)/(TP+TN+FP+FN)
    speci = TN/(TN+FP)
    sensi = TP/(TP+FN)
    precision = TP/(TP+FP)
    recall = TP/(TP+FN)
    TPR = TP/(TP + FN)
    TNR = TN/(TN + FP)
    FPR = FP/(TN + FP)
    FNR = FN/(TP + FN)
    pos_pred_val = TP /(TP+FP)
    neg_pred_val = TN /(TN+FN)

    print ("Model Accuracy value is              : ", round(accuracy*100,2),"%")
    print ("Model Sensitivity value is           : ", round(sensi*100,2),"%")
    print ("Model Specificity value is           : ", round(speci*100,2),"%")
    print ("Model Precision value is             : ", round(precision*100,2),"%")
    print ("Model Recall value is                : ", round(recall*100,2),"%")
    print ("Model True Positive Rate (TPR)       : ", round(TPR*100,2),"%")
    print ("Model False Positive Rate (FPR)      : ", round(FPR*100,2),"%")
    print ("Model Poitive Prediction Value is    : ", round(pos_pred_val*100,2),"%")
    print ("Model Negative Prediction value is   : ", round(neg_pred_val*100,2),"%")
```

In [64]:
```
confusion_matrix = metrics.confusion_matrix(label_train_pred_df.Converted, label_train_
print(confusion_matrix)
```

```
[[3534  440]
 [ 730 1764]]
```

In [65]:
```
model_performance_check(confusion_matrix)
```

```
Model Accuracy value is              :  81.91 %
Model Sensitivity value is           :  70.73 %
```

```
Model Specificity value is          :  88.93 %
Model Precision value is            :  80.04 %
Model Recall value is               :  70.73 %
Model True Positive Rate (TPR)      :  70.73 %
Model False Positive Rate (FPR)     :  11.07 %
Model Poitive Prediction Value is   :  80.04 %
Model Negative Prediction value is  :  82.88 %
```

In [66]:

```
#Finding Optimal Cutoff Point
cut_off = [float(x)/10 for x in range(10)]
for i in cut_off:
    label_train_pred_df[i]= label_train_pred_df.Predicted.map(lambda d: 1 if d > i else
label_train_pred_df.head()
```

Out[66]:

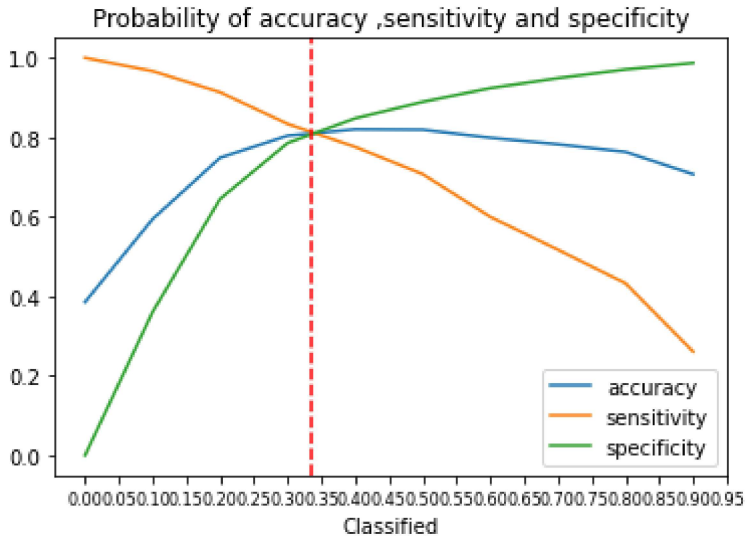
	Converted	Predicted	Classified	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0	1	0.861417	1	1	1	1	1	1	1	1	1	1	0
1	1	0.984790	1	1	1	1	1	1	1	1	1	1	1
2	1	0.960643	1	1	1	1	1	1	1	1	1	1	1
3	0	0.239815	0	1	1	1	0	0	0	0	0	0	0
4	0	0.072815	0	1	0	0	0	0	0	0	0	0	0

In [67]:

```
optimization_df = pd.DataFrame( columns = ['Classified','accuracy','sensitivity','speci

prob = [i/10 for i in range(10)]

for i in prob:
    confusion_matrix = metrics.confusion_matrix(label_train_pred_df.Converted, label_tr
    total1=sum(sum(confusion_matrix))
    accuracy = (confusion_matrix[0,0]+confusion_matrix[1,1])/total1
    specificity = confusion_matrix[0,0]/(confusion_matrix[0,0]+confusion_matrix[0,1])
    sensitivity = confusion_matrix[1,1]/(confusion_matrix[1,0]+confusion_matrix[1,1])
    optimization_df.loc[i] =[i,accuracy,sensitivity,specificity]
```

In [68]:

```
optimization_df.plot.line(x='Classified', y=['accuracy','sensitivity','specificity'])
plt.title('Probability of accuracy ,sensitivity and specificity')
plt.xticks(np.arange(0,1,step=0.05),size=8)
plt.axvline(x=0.335, color='r', linestyle='--') # additing axline
plt.show()
```



In [69]:
```
optimization_df
```

Out[69]:

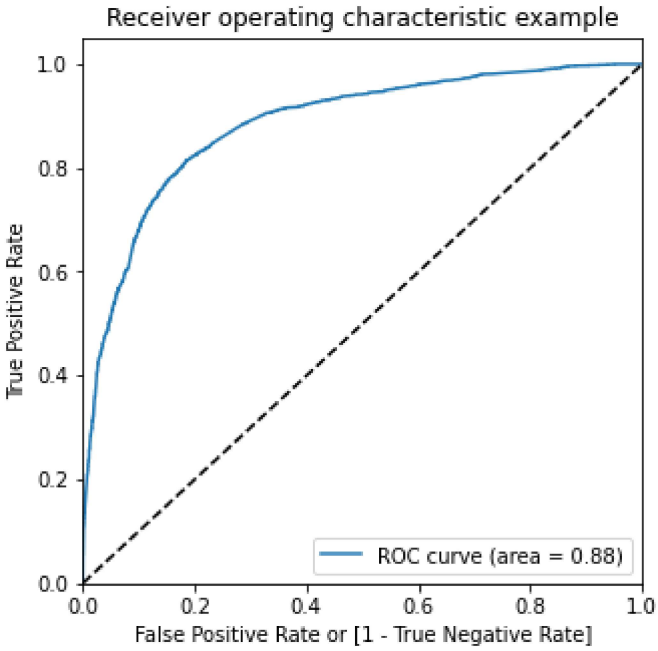
	Classified	accuracy	sensitivity	specificity
0.0	0.0	0.385591	1.000000	0.000000
0.1	0.1	0.594310	0.966720	0.360594
0.2	0.2	0.748609	0.913392	0.645194
0.3	0.3	0.804267	0.833601	0.785858
0.4	0.4	0.820037	0.775862	0.847760
0.5	0.5	0.819109	0.707298	0.889280
0.6	0.6	0.798856	0.599840	0.923754
0.7	0.7	0.782468	0.517241	0.948918
0.8	0.8	0.763296	0.432638	0.970810
0.9	0.9	0.707174	0.261026	0.987167

0.4 seems to be the optimal cut off.

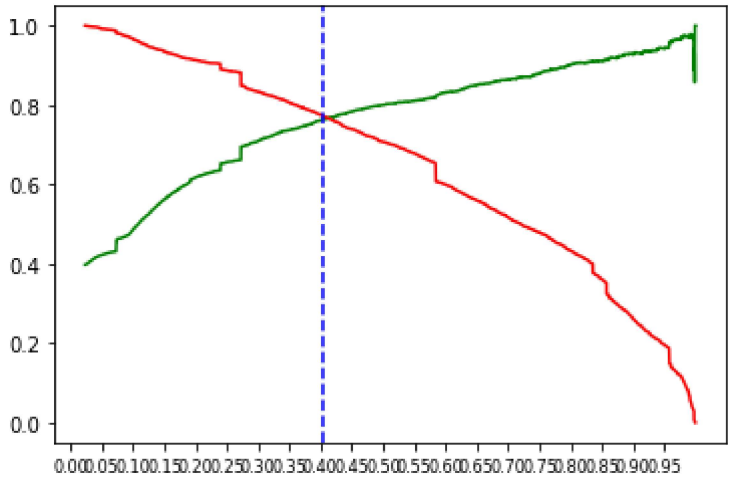
In [70]:
```
label_train_pred_df['Classified'] = label_train_pred_df.Predicted.map(lambda d: 1 if d
confusion_matrix = metrics.confusion_matrix(label_train_pred_df.Converted, label_train_
model_performance_check(confusion_matrix)
```

```
Model Accuracy value is              :  82.0 %
Model Sensitivity value is           :  77.59 %
Model Specificity value is           :  84.78 %
Model Precision value is             :  76.18 %
Model Recall value is                :  77.59 %
Model True Positive Rate (TPR)       :  77.59 %
Model False Positive Rate (FPR)      :  15.22 %
Model Poitive Prediction Value is    :  76.18 %
Model Negative Prediction value is   :  85.77 %
```

In [71]:
```
fpr, tpr, thresholds = metrics.roc_curve(label_train_pred_df.Converted, label_train_pre
                                                    drop_intermediate = False)
auc_score = metrics.roc_auc_score(label_train_pred_df.Converted, label_train_pred_df.Pr
plt.figure(figsize=(5, 5))
plt.plot( fpr, tpr, label='ROC curve (area = %0.2f)' % auc_score )
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate or [1 - True Negative Rate]')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic example')
plt.legend(loc="lower right")
plt.show()
```



In [72]:
```
#Precision and recall tradeoff
p, r, thresholds = precision_recall_curve(label_train_pred_df.Converted, label_train_pr
plt.plot(thresholds, p[:-1], "g-")
plt.plot(thresholds, r[:-1], "r-")
plt.xticks(np.arange(0,1,step=0.05),size=8)
plt.axvline(x=0.404, color='b', linestyle='--') # additing axline
plt.show()
```



Precision recall trade-off also giving cut-off point as 0.4.

Model Prediction

In [73]:
```
# Scaling the test data
test_df[numerical_columns] = scaler.transform(test_df[numerical_columns])
test_df.head()
```

Out[73]:

	Converted	Total Time Spent on Website	Lead Origin_Landing Page Submission	Lead Origin_Lead Add Form	Lead Origin_Lead Import	Lead Origin_Quick Add Form	Lead Source_Direct Traffic	Sour
8087	1	-0.896237	0	1	0	0	0	
5177	0	0.574699	0	0	0	0	0	
1148	1	-0.592536	1	0	0	0	1	
7291	1	-0.799272	1	0	0	0	1	
1979	1	2.078566	1	0	0	0	0	

5 rows × 101 columns

In [74]:
```
#adding contant value
test_df_sm = sm.add_constant(test_df[rfe_feature_list])
test_df_sm.columns
```

Out[74]:
```
Index(['const', 'Total Time Spent on Website', 'Lead Origin_Lead Add Form',
       'Lead Source_Olark Chat', 'Do Not Email_Yes',
       'Last Activity_Converted to Lead', 'Last Activity_Email Bounced',
       'Last Activity_Olark Chat Conversation',
       'Last Activity_Page Visited on Website',
       'What is your current occupation_Working Professional',
       'Last Notable Activity_Email Link Clicked',
       'Last Notable Activity_Email Opened', 'Last Notable Activity_Modified',
       'Last Notable Activity_Olark Chat Conversation'],
      dtype='object')
```

In [75]:
```
test_pred = log_model.predict(test_df_sm)
```

In [76]:
```
test_pred_df = pd.concat([test_df[[label]], pd.DataFrame(test_pred)],axis=1)
```

In [77]:
```
test_pred_df = test_pred_df.rename(columns={ 0 : 'Predicted'})
test_pred_df
```

Out[77]:

	Converted	Predicted
8087	1	0.990541
5177	0	0.375114
1148	1	0.043688
7291	1	0.300885
1979	1	0.922555
...	...	...
5777	1	0.320746
3663	0	0.072815
523	0	0.677102
7346	1	0.741684
6946	1	0.636206

2772 rows × 2 columns

In [78]:
```
test_pred_df['Classified'] = test_pred_df.Predicted.map(lambda d: 1 if d > 0.4 else 0)

test_pred_df.head()
```

Out[78]:

	Converted	Predicted	Classified
8087	1	0.990541	1
5177	0	0.375114	0
1148	1	0.043688	0
7291	1	0.300885	0
1979	1	0.922555	1

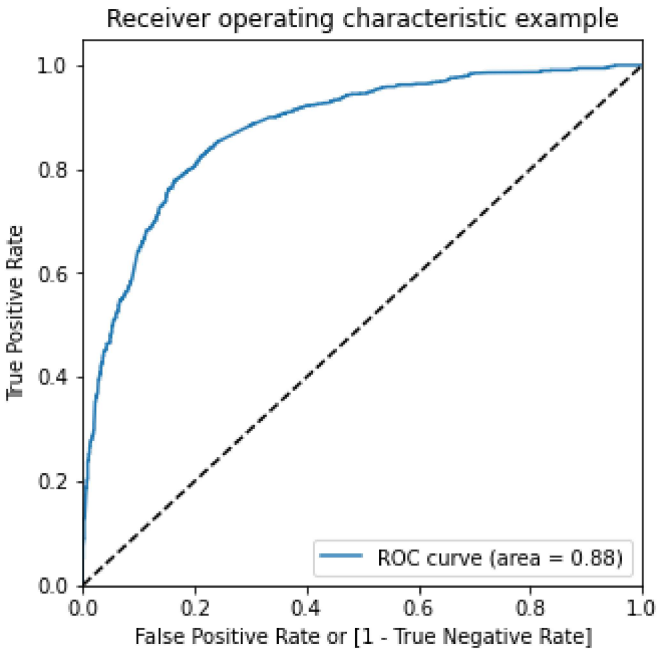
In [79]:
```
confusion_matrix = metrics.confusion_matrix(test_pred_df.Converted, test_pred_df.Classi
model_performance_check(confusion_matrix)
```

```
Model Accuracy value is             :  81.1 %
Model Sensitivity value is          :  76.48 %
Model Specificity value is          :  83.99 %
Model Precision value is            :  74.93 %
```

```
Model Recall value is               :  76.48 %
Model True Positive Rate (TPR)      :  76.48 %
Model False Positive Rate (FPR)     :  16.01 %
Model Poitive Prediction Value is   :  74.93 %
Model Negative Prediction value is  :  85.09 %
```

In [80]:

```
fpr, tpr, thresholds = metrics.roc_curve(test_pred_df.Converted, test_pred_df.Predicted
                                         drop_intermediate = False)
auc_score = metrics.roc_auc_score(test_pred_df.Converted, test_pred_df.Predicted)
plt.figure(figsize=(5, 5))
plt.plot( fpr, tpr, label='ROC curve (area = %0.2f)' % auc_score )
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate or [1 - True Negative Rate]')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic example')
plt.legend(loc="lower right")
plt.show()
```



Lead Score

In [81]:

```
test_pred_df['score']=(test_pred_df['Predicted']*100)
test_pred_df.sort_values(by='score',ascending=False)
test_pred_df
```

Out[81]:

	Converted	Predicted	Classified	score
8087	1	0.990541	1	99.054127
5177	0	0.375114	0	37.511364
1148	1	0.043688	0	4.368835
7291	1	0.300885	0	30.088451

	Converted	Predicted	Classified	score
1979	1	0.922555	1	92.255519
...	...	...	...	...
5777	1	0.320746	0	32.074588
3663	0	0.072815	0	7.281535
523	0	0.677102	1	67.710209
7346	1	0.741684	1	74.168378
6946	1	0.636206	1	63.620635

2772 rows × 4 columns

In [82]:
```
log_model.params.sort_values(ascending=False)
```

Out[82]:
```
Lead Origin_Lead Add Form                                4.071536
What is your current occupation_Working Professional     2.855973
Lead Source_Olark Chat                                   1.293931
Total Time Spent on Website                              1.153877
const                                                    0.079172
Last Activity_Converted to Lead                         -0.988811
Last Activity_Email Bounced                             -1.094667
Last Activity_Page Visited on Website                   -1.252645
Do Not Email_Yes                                        -1.289451
Last Notable Activity_Email Opened                      -1.321220
Last Activity_Olark Chat Conversation                   -1.390532
Last Notable Activity_Olark Chat Conversation           -1.477188
Last Notable Activity_Modified                          -1.492650
Last Notable Activity_Email Link Clicked                -1.781275
dtype: float64
```

In [83]:
```
log_model.params.index.to_list()
```

Out[83]:
```
['const',
 'Total Time Spent on Website',
 'Lead Origin_Lead Add Form',
 'Lead Source_Olark Chat',
 'Do Not Email_Yes',
 'Last Activity_Converted to Lead',
 'Last Activity_Email Bounced',
 'Last Activity_Olark Chat Conversation',
 'Last Activity_Page Visited on Website',
 'What is your current occupation_Working Professional',
 'Last Notable Activity_Email Link Clicked',
 'Last Notable Activity_Email Opened',
 'Last Notable Activity_Modified',
 'Last Notable Activity_Olark Chat Conversation']
```

'Do Not Email_Yes','Last Activity_Converted to Lead','Last Activity_Email Bounced','Last Activity_Olark Chat Conversation','Last Activity_Page Visited on Website','Last Notable Activity_Email Link Clicked','Last Notable Activity_Email Opened','Last Notable Activity_Modified','Last Notable Activity_Olark Chat Conversation'