Summary Report – Steps Followed
To build a Machine Learning logistic regression model for lead generation, I have followed these steps:

1. **Exploratory Data Visualization** – Performing EDA on the dataset that includes historical information on leads. The dataset contains features like 'Lead Origin', 'Lead Source', 'Do Not Email', 'Do Not Call', and others listed.

   There are 29 categorical variables
   The categorical variables are : 'Lead Origin', 'Lead Source', 'Do Not Email', 'Do Not Call', 'Last Activity', 'Country', 'Specialization', 'How did you hear about X Education', 'What is your current occupation', 'What matters most to you in choosing a course', 'Search', 'Magazine', 'Newspaper Article', 'X Education Forums', 'Newspaper', 'Digital Advertisement', 'Through Recommendations', 'Receive More Updates About Our Courses', 'Tags', 'Lead Quality', 'Update me on Supply Chain Content', 'Get updates on DM Content', 'Lead Profile', 'City', 'Asymmetrique Activity Index', 'Asymmetrique Profile Index', 'I agree to pay the amount through cheque', 'A free copy of Mastering The Interview', 'Last Notable Activity'
   Numerical variables are –
   'TotalVisits', 'Total Time Spent on Website', 'Page Views Per Visit',
   'Asymmetrique Activity Score', 'Asymmetrique Profile Score']

1. **Data Preprocessing:**
   - Handle Missing Data: Checking for missing values in the dataset and decide on appropriate strategies to impute or remove them. Used median filling technique to remove those- Missing values in the data are
     - TotalVisits -137
     - Total Time Spent on Website - 0
     - Page Views Per Visit - 137
     - Asymmetrique Activity Score - 4218
     - Asymmetrique Profile Score - 4218
   - Categorical Variables: Encode categorical variables using techniques like one-hot encoding to convert them into numerical format.
   - Feature Scaling: Scale numerical features like 'TotalVisits', 'Total Time Spent on Website', and 'Page Views Per Visit' to ensure they have similar scales.
1. **Data Splitting:** Splitting the dataset into training and testing sets to evaluate the model's performance effectively. I have performed a 80-20 split for training and testing sets.
1. **Model Building:**
   - Selected Features: Choosing the relevant features from the dataset to be used as input variables. All given features are not important and relevant such as 'Prospect ID', 'Lead Number' etc.
   - Logistic Regression: Building a logistic regression model, which is suitable for binary classification tasks like lead generation where the given target variable is 'Converted.'
1. **Model Training:** Train the logistic regression model using the training data. Setting and optimizing the model for the best outcome. Also, performing Hyper Parameter optimization to tune the model.
1. **Model Evaluation:**
   - Performance Metrics: Evaluate the model using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to assess its effectiveness in lead generation. Below are the results attached. “0” means lead not converted and “1” means leads converted in the below classification report

```
Test Result:
================================================
Accuracy Score: 81.28%

________________________________________________
CLASSIFICATION REPORT:
                    0           1  accuracy    macro avg  weighted avg
precision    0.830926    0.780924  0.812771     0.805925      0.811364
recall       0.869333    0.724758  0.812771     0.797046      0.812771
f1-score     0.849696    0.751793  0.812771     0.800745      0.811393
support    1125.000000  723.000000  0.812771  1848.000000   1848.000000

________________________________________________
Confusion Matrix:
 [[978 147]
 [199 524]]
```

Building a logistic regression model for lead generation can help any company make data-driven decisions, prioritize leads, and allocate resources more effectively, ultimately leading to improved conversion rates and higher sales efficiency.