MACHINE LEARNING BASED BANK CUSTOMER CHURN PREDICTION

1. Collect the dataset containing customer features and churn labels.
2. Load and preprocess the data using Python libraries like Pandas.
3. Handle missing values, encode categorical variables, and normalize numerical features using Python.
4. Perform exploratory data analysis (EDA) to identify patterns and visualize data using Python libraries such as Matplotlib and Seaborn.
5. Address class imbalance using Borderline-SMOTE from the Imbalanced-learn library in Python.
6. Split the dataset into training and testing sets using train_test_split from Scikit-learn in Python.
7. Train machine learning models such as Gradient Boosting Machine (GBM) and XGBoost using Python in Jupyter Notebook.
8. Evaluate model performance using Python with metrics like Accuracy, Precision, Recall, F1-Score, and ROC-AUC.
9. Save the trained model using joblib or pickle in Python for deployment.
10. Deploy the model in a Flask web application written in Python for real-time churn prediction.
11. Store details of churn-predicted customers in an Excel file (risk_customers.xlsx) using Pandas in Python.
12. Test the system using Python for accuracy, functionality, and performance.