**Customer Segmentation for Targeted Marketing**

**Project Documentation – Phase 1 & 2**

**Technologies Used**: MySQL, Python, Pandas

**Project Objective**

The goal of this project is to analyze and segment customers of a bank based on demographic and behavioral attributes to support **targeted marketing strategies**. The data pipeline includes SQL-based extraction, Python-based data cleaning, transformation, and visualization.

**Dataset Source**

**Dataset**: Bank Marketing Dataset – Kaggle  
**File**: bank.csv  
**Records**: 11,162  
**Columns**: 17  
**Context**: Real marketing campaign data from a Portuguese banking institution.

**Step 1: Data Extraction from MySQL**

**1. Load CSV into MySQL**

* A local MySQL database named bank\_marketing was created.
* A table bank\_data was defined based on the dataset structure.
* The CSV file was imported using either the MySQL Workbench import wizard or the LOAD DATA INFILE command.

**MySQL Table Definition:**

CREATE TABLE bank\_data (

age INT,

job VARCHAR(50),

marital VARCHAR(20),

education VARCHAR(50),

default\_status VARCHAR(10),

balance INT,

housing VARCHAR(10),

loan VARCHAR(10),

contact VARCHAR(20),

day INT,

month VARCHAR(10),

duration INT,

campaign INT,

pdays INT,

previous INT,

poutcome VARCHAR(20),

y VARCHAR(10)

);

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**2. Connect MySQL to Python**

Using mysql-connector-python and pandas.read\_sql():

import pandas as pd

import mysql.connector

conn = mysql.connector.connect(

host='localhost',

user='root',

password='admin',

database='bank\_marketing'

)

query = "SELECT \* FROM bank\_data"

df = pd.read\_sql(query, conn)

Successfully extracted **11,162 records** into a pandas DataFrame for further cleaning.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

**Step 2: Data Cleaning using Pandas**

**1. Renaming Reserved SQL Column**

Column default\_status was renamed to default in pandas for consistency.

**2. Categorical Cleaning**

* All categorical text columns were:
  + Trimmed of extra spaces
  + Lowercased for consistency

**3. Binary Field Conversion**

Fields like default, housing, loan, and y (campaign response) were converted:

binary\_map = {'yes': 1, 'no': 0}

df['default'] = df['default'].map(binary\_map)

df['housing'] = df['housing'].map(binary\_map)

df['loan'] = df['loan'].map(binary\_map)

df['y'] = df['y'].map(binary\_map)

**4. Outlier Detection**

Potential outliers in balance were flagged for further review:

df[df['balance'] < -1000]

**5. Export Cleaned Data**

The cleaned DataFrame was saved to a file for the next transformation stage:

df.to\_csv("data/cleaned\_bank\_data.csv", index=False)

**Cleaned dataset** is ready with consistent formats and valid data types.

**Project Directory Structure**

customer\_segmentation\_project/

│

├── data/

│ ├── bank.csv

│ └── cleaned\_bank\_data.csv

│

├── load\_and\_explore.py

├── clean\_data.py

└── README.md (this file)

**Summary**

| **Stage** | **Outcome** |
| --- | --- |
| SQL Import | Bank data loaded into MySQL for flexible querying |
| Extraction | Pulled into Python using pandas and MySQL connector |
| Cleaning | Text normalized, binary fields encoded, outliers flagged |
| Output | Clean CSV for use in feature engineering and visualization |

**A screen shot of a computer

AI-generated content may be incorrect.**

**Step 3: Data Transformation**