Assignment 1

Problem Statement

A person's creditworthiness is often associated (conversely) with the likelihood they may default on loans. Here, we ask you to look at data on loan applicants, extract useful insights and guide a business towards decisions. We're giving you anonymized data on about 1000 loan applications, along with a certain set of attributes about the applicant itself, and whether they were considered high risk. We'd like you to work your magic on this. Do note that it is worse to state an applicant as a low credit risk when they are actually a high risk (cost=4 below), than it is to state an applicant to be a high credit risk when they aren't (cost=1 below).

		Predicted Outcome	
		0	1
Actual Outcome	0	0	1
	1	4	0

This table contains a possible cost matrix where rows represent the actual classification and the columns the predicted classification with classes: 0 = Low credit risk, 1 = High credit risk

Instructions

Answer the questions (using SQL, Python) and explain your rationale in the write-up :

1. How would you segment customers based on their risk (of default)
2. Which of these segments / sub-segments would you propose be approved?
    - For e.g. Would a person with critical credit history be more creditworthy? Are young people more creditworthy? Would a person with more credit accounts be more creditworthy?
3. What other insights can you share about the general creditworthiness of these segments?
4. Tell us what your observations were on the data itself (completeness, skews) and how you would treat any anomalies (for eg - missing data)

1.How would you segment customers based on their risk (of default)

- Customers have been segmented into four risk categories ('Low', 'Medium', 'High', and 'Very High') based on their age, with the majority of the customers falling into the 'Low' risk category.
- The 'grouped' dataframe shows the percentage of high-risk applications for each combination of attributes, with some combinations having a higher percentage of high-risk applications than others.
- The heatmap shows the percentage of high-risk applications for each combination of attributes in the pivot table, with some combinations having a higher percentage of high-risk applications than others. The heatmap helps to identify the combinations of attributes that are associated with high-risk applications.

Process:

- The code merges two dataframes (applicant_df and loan_df) on the common column 'applicant_id', resulting in a new dataframe named 'df'.
- Unnecessary columns (such as 'Months_loan_taken_for', 'Purpose', 'Property', etc.) are dropped from the 'df' dataframe using the drop() method.
- The info() method is called to get a summary of the 'df' dataframe, including its shape, data types, and non-null values for each column.
- The isna().sum() method is used to check if there are any missing values in the 'df' dataframe, and the results are displayed.
- The missing values in the 'df' dataframe are replaced with the median value of each column using the fillna() method.
- The skewness of each column in the 'df' dataframe is calculated using the skew() method.
- Outliers are removed from the 'Primary_applicant_age_in_years' column in the 'df' dataframe by keeping only the rows where the age is less than 100.
- Customers are segmented into risk categories based on their 'Primary_applicant_age_in_years' using the qcut() method, and a new column 'Risk' is added to the 'df' dataframe to store this information.
- The number of applicants in each risk category is visualized using a bar plot created with the help of the plot() method of the 'groupby()' object in the 'df' dataframe.
- The code merges the two dataframes (applicant_df and loan_df) again on the common column 'applicant_id', resulting in a new dataframe named 'df'.
- The percentage of high-risk applications for each combination of attributes is calculated by grouping the 'df' dataframe based on certain attributes (such as 'Employment_status', 'Purpose', etc.) and taking the mean of the 'high_risk_applicant' column.
- The pivot_table() method is used to create a pivot table with counts of high-risk applications for each combination of attributes, and the resulting table is used to calculate the percentage of high-risk applications for each combination of attributes.
- A heatmap is plotted using the seaborn library to visualize the percentage of high-risk applications for each combination of attributes in the pivot table. The heatmap is customized by specifying the colormap, annotation format, and figure size.

3. What other insights can you share about the general creditworthiness of these segments?

- Younger customers are more likely to be approved for a loan than older customers.
- Married customers are more likely to be approved for a loan than single customers.
- Customers with no dependents are more likely to be approved for a loan than customers with dependents.
- Customers who own their own home are more likely to be approved for a loan than customers who rent.
- Customers who are employed full-time are more likely to be approved for a loan than customers who are unemployed or part-time employed.
- Customers with a high savings account balance are more likely to be approved for a loan than customers with a low savings account balance.

4.Tell us what your observations were on the data itself (completeness, skews) and how you would treat any anomalies (for eg - missing data)

- The data is relatively complete. There are only a few missing values.
- The data is slightly skewed. There are more customers with a low risk score than customers with a high risk score.