3. Entity Descriptions and Relationships

3.1 Customer Table

- **Purpose**: Stores information about bank customers who apply for loans.
- **Attributes**:
  - `customerId`: Unique identifier for each customer (Primary Key).
  - `name`: Customer's full name.
  - `address`: Customer's residential address.
  - `phone`: Customer's phone number.
  - `email`: Customer's email address.
  - `nationalId`: Customer's national identification number.
  - `CSI_Score`: Customer Satisfaction Index score, used to assess customer feedback.

Usage Scenario:

- Each customer can submit multiple loan applications. For example, a customer named John Doe can apply for both a personal loan and a car loan.

3.2 LoanApplication Table

- **Purpose**: Captures details of loan applications submitted by customers.
- **Attributes**:
  - `applicationId`: Unique identifier for each loan application (Primary Key).
  - `customerId`: References the `customerId` from the **Customer** table (Foreign Key).
  - `loanAmount`: Amount requested by the customer.
  - `applicationDate`: Date when the loan application was submitted.
  - `status`: Current status of the loan application (e.g., Pending, Approved, Rejected).
  - `branchId`: References the `branchId` from the **Branch** table (Foreign Key).
  - `documentLinks`: Links to supporting documents provided by the customer.

Usage Scenario:

- A customer applies for a loan at a specific bank branch. For example, Olivier applies for a loan of 10,000 FRW at the Kigali branch.

3.3 Loan Table

- **Purpose**: Stores details of approved loans.
- **Attributes**:
  - `loanId`: Unique identifier for each loan (Primary Key).

    - `applicationId`: References the `applicationId` from the **LoanApplication** table (Foreign Key).
    - `loanAmount`: Amount approved by the bank.
    - `interestRate`: Interest rate applied to the loan.
    - `termMonths`: Loan term in months.
    - `approvalDate`: Date when the loan was approved.
    - `fundingDate`: Date when the loan funds were disbursed.
    - `loanStatus`: Status of the loan (e.g., Active, Closed).

Usage Scenario:

- Once a loan application is approved, it is moved to the **Loan** table for disbursement and repayment tracking.

3.4 Branch Table

- **Purpose**: Maintains information about bank branches.
- **Attributes**:
    - `branchId`: Unique identifier for each branch (Primary Key).
    - `branchName`: Name of the bank branch.
    - `location`: Physical address of the branch.
    - `phone`: Contact number of the branch.

Usage Scenario:

- Loan applications are submitted at different branches. For example, a loan application can be processed at the Kigali branch.

3.5 CreditCheck Table

- **Purpose**: Tracks credit checks performed for loan applications.
- **Attributes**:
    - `checkId`: Unique identifier for each credit check (Primary Key).
    - `applicationId`: References the `applicationId` from the **LoanApplication** table (Foreign Key).
    - `creditScore`: Customer's credit score as determined by the bank.
    - `remarks`: Additional notes on the credit evaluation.
    - `checkDate`: Date the credit check was performed.

Usage Scenario:

- Before approving a loan, a credit check is performed to assess the customer's creditworthiness.

3.6 Feedback Table

- **Purpose**: Stores customer feedback regarding their loan experience.
- **Attributes**:
  - `feedbackId`: Unique identifier for each feedback entry (Primary Key).
  - `customerId`: References the `customerId` from the **Customer** table (Foreign Key).
  - `loanId`: References the `loanId` from the **Loan** table (Foreign Key).
  - `score`: Rating provided by the customer (e.g., 1 to 5 stars).
  - `comments`: Feedback comments provided by the customer.
  - `feedbackDate`: Date the feedback was submitted.

Usage Scenario:

- Customers provide feedback after their loan application is processed, which helps the bank improve its services.

3.7 ApplicationProcessing Table

- **Purpose**: Logs the various stages of loan application processing.
- **Attributes**:
  - `processId`: Unique identifier for each processing step (Primary Key).
  - `applicationId`: References the `applicationId` from the **LoanApplication** table (Foreign Key).
  - `stepName`: Name of the processing step (e.g., Document Verification, Approval).
  - `startTime`: Start time of the processing step.
  - `endTime`: End time of the processing step.
  - `status`: Status of the processing step (e.g., Completed, Pending).

Usage Scenario:

- Tracks each step of the loan application process to ensure efficient handling and transparency.

3.8 FinancialLoss Table

- **Purpose**: Records potential financial losses due to abandoned or rejected loan applications.
- **Attributes**:
  - `lossId`: Unique identifier for each financial loss entry (Primary Key).
  - `applicationId`: References the `applicationId` from the **LoanApplication** table (Foreign Key).
  - `lossAmount`: Estimated loss amount due to a rejected or abandoned loan.
  - `reason`: Reason for the financial loss.

- `recordDate`: Date the financial loss was recorded.

Usage Scenario:

- Helps the bank analyze reasons for loan rejection and assess financial risks.

---

4. Relationships and Constraints

Primary and Foreign Key Relationships

- **Customer → LoanApplication**: A customer can have multiple loan applications.
- **LoanApplication → Loan**: Each loan is associated with a single application.
- **Branch → LoanApplication**: Applications are submitted at specific branches.
- **LoanApplication → CreditCheck**: Multiple credit checks can be conducted for a single application.
- **Loan → Feedback**: Feedback is collected for loans that have been disbursed.
- **LoanApplication → ApplicationProcessing**: Each application goes through multiple processing steps.
- **LoanApplication → FinancialLoss**: Records losses due to abandoned or rejected applications.

Constraints

- **Unique Keys**: The `customerId`, `applicationId`, `loanId`, etc., are unique and auto-incremented.
- **Foreign Keys**: Ensure data integrity by linking related records across tables.
- **Data Types**: Ensure consistency in how data is stored, e.g., using `varchar` for text, `int` for IDs, and `decimal` for financial amounts.

---

5. Use Case Scenarios

Scenario 1: Submitting a Loan Application

1. A customer visits a branch to apply for a loan.
2. The branch officer collects customer details and documents, which are stored in the **Customer** and **LoanApplication** tables.
3. A credit check is performed and recorded in the **CreditCheck** table.
4. The application goes through multiple processing steps tracked in the **ApplicationProcessing** table.
5. If approved, a loan entry is created in the **Loan** table.

Scenario 2: Customer Feedback Collection

1. After loan disbursement, the bank sends a feedback request to the customer.

2. The customer provides feedback, which is recorded in the **Feedback** table.
3. The feedback is used to improve customer service.

Scenario 3: Analyzing Financial Losses

1. If a loan application is abandoned or rejected, the bank records the potential loss in the **FinancialLoss** table.
2. This data helps the bank identify reasons for loan rejection and improve its risk management policies.

---

6. Conclusion

This data model is designed to streamline the loan origination process, improve customer service, and provide insights into application processing, credit evaluation, and financial risks. It ensures data integrity through well-defined relationships and constraints, making it scalable and efficient for the bank's operational needs.