## 

## 

## 

## 

**Bank Management Database System**

**Abstract**

This project aims to develop a comprehensive and efficient system to manage bank operations ensuring accuracy, consistency, and ease of access to information

**Khushboo Surana**

## 

## Table of Contents

[**Table of Contents 2**](#_w6350s9skepu)

[**1. Business Description 3**](#_1keh2wksucsv)

[1.1 Overview: Nature and Scope 3](#_z5lddq1p8ev5)

[1.2. Purpose, Benefits and Anticipated Functionalities 3](#_eitxkulrg84n)

[**2. Enhanced Entity-Relationship Diagram (Conceptual Design): Link to EER Diagram 5**](#_4hmd12f6alg)

[2.1 Data Entities and their Attributes 6](#_go8vkc6u5ojy)

[2.2. Entity-Relationships 8](#_nz978j9r3ufq)

[**3. Relational Model (Logical Diagram): Link to Relational Model (Logical Diagram) 9**](#_59ae9lhps1jq)

[3.1 Normalization (Third Normal Form) 9](#_q2frohax22le)

[**4. SQL Database Implementation 10**](#_c0wkbogmn7un)

[4.1. SQL Database Implementation: Without Joins 10](#_bcfzsq1p976e)

[4.2. SQL Database Implementation: Employing Joins 12](#_uqgadrib45c2)

[**5. Administrative Functions 14**](#_gt98o0sqz3g7)

[5.1 Potential Issues and Measures to Address 15](#_3kz57oacdikj)

[**6. Conclusion 15**](#_t11n843dgf6)

## 

## 

## 

## 

## 

## 

## 1. Business Description

### 1.1 Overview: Nature and Scope

The chosen business is a bank, which operates as a financial institution providing a wide array of services to individual account holders. The primary functions of the bank include accepting deposits, granting loans, offering savings and checking accounts, managing financial transactions, and providing investment and wealth management services.

The scope of the bank encompasses various branches and departments, each responsible for different aspects of banking operations: retail banking, credit and loan management, and financial advisory services. The bank serves a diverse customer base, including individual account holders, small businesses, large corporations, and institutional clients. However, for our project, we are focusing exclusively on **retail banking** for individual account holders, excluding small businesses, large corporations, and institutional clients. The project exclusively considers retail banking to streamline and optimize services for individual account holders, ensuring a targeted and efficient system implementation. It operates both physical branches and online banking platforms to cater to the needs of its customers.

### 1.2. Purpose, Benefits and Anticipated Functionalities

**Purpose:** The Bank Management Database System (BMDS) aims to optimize bank operations by centralizing data related to customers, accounts, transactions, and branches. This ensures data accuracy, enhances operational efficiency, and provides real-time access to critical information, enabling better service delivery and informed decision-making.

**Benefits and Anticipated Functionalities:**

**1. Centralized Data Management:** A unified repository ensures data consistency, accuracy, and integrity across all branches and departments, reducing redundancy and ensuring regulatory compliance.

* **Customer Information Management:** Comprehensive records, including personal details and contact information, enable efficient service and personalized banking experiences.
* **Account Management:** Supports various account types (savings, checking, fixed deposits), facilitating creation, maintenance, closure, and real-time transaction tracking for accurate balance information.
* **Loan Management:** Streamlines loan processing from application to repayment, with tools for creditworthiness assessment and interest calculation, reducing default risks.
* **Branch Information Management:** Stores branch details, supporting branch-level reporting, decision-making, and performance monitoring.

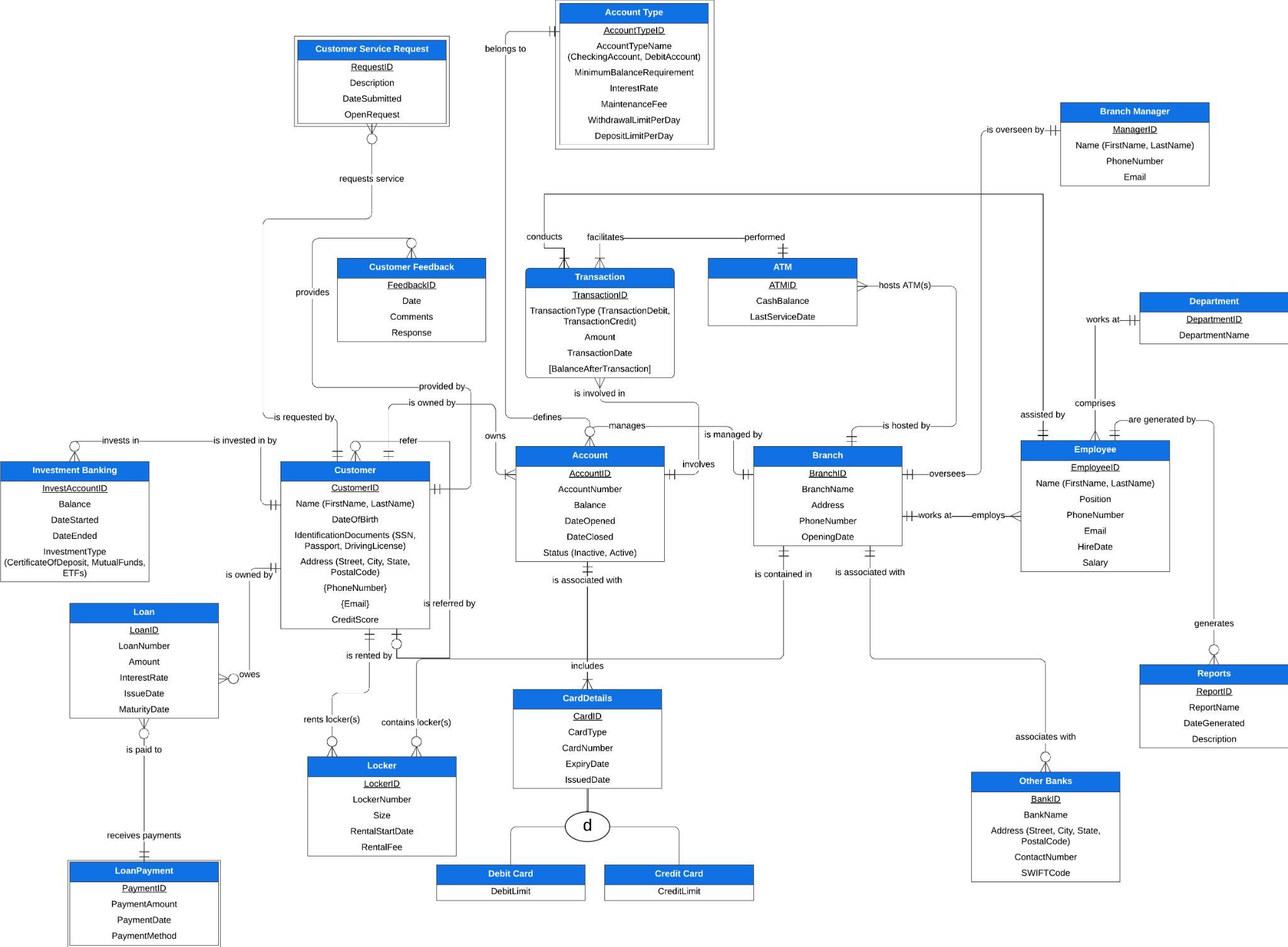
**2. Transaction Processing:** Manages all types of transactions (deposits, withdrawals, transfers, payments), ensuring accuracy and real-time updates, and helping detect and prevent fraud.

**3. Reporting and Analytics:** Provides robust reporting capabilities for financial insights, customer data, and performance metrics, supporting strategic decisions and regulatory compliance.

**4. Security and Access Control:** Implements strong security measures, user authentication, and role-based access control to protect sensitive data and ensure authorized access.

By integrating these functionalities, BMDS enhances operational efficiency, improves customer service, supports strategic decision-making, and ensures robust security and compliance.

## 2. Enhanced Entity-Relationship Diagram (Conceptual Design): [Link to EER Diagram](https://lucid.app/lucidchart/1526d398-3e05-4eb4-b1b0-86a011eb8ca0/edit?viewport_loc=667%2C-4072%2C4500%2C2454%2CjSjITa0wlXQf&invitationId=inv_f246b15b-2168-4961-8e0b-d2f8f0018253)



1. **Unary Relationships**:
   1. Example: A Customer can refer another Customer (self-referential relationship).
2. **Associative Entities**:
   1. Example: Transaction connects Account to ATM and acts as an associative entity.
3. **Disjoint Constraints**:
   1. Example: Card can either be a Debit Card or a Credit Card (disjoint subtypes).
4. **Multivalued Attributes**:
   1. Example: Customers can have multiple phone numbers and email IDs.
5. **Composite Attributes**:
   1. Example: Customer/Employee Name consists of FirstName and LastName; Address consists of Street, City, State, PostalCode.
6. **Weak Entities**:
   1. Example: LoanPayment is a weak entity dependent on Loan. AccountType is a weak entity dependent on Account.
7. **Supertype and Subtype**:
   1. Example: CardDetails as a supertype can have subtypes such as Debit Card and Credit Card.
8. **Completeness and Disjointness**:
   1. Completeness: Total completeness where each CardDetails must be either a Debit Card or Credit Card.
   2. Disjointness: Disjointness constraint ensuring a CardDetails cannot be both Debit Card and Credit Card simultaneously.
9. **Subtype Discriminator**:
   1. Example: CardType attribute to distinguish between Debit Card and Credit Card.

### 2.1 Data Entities and their Attributes

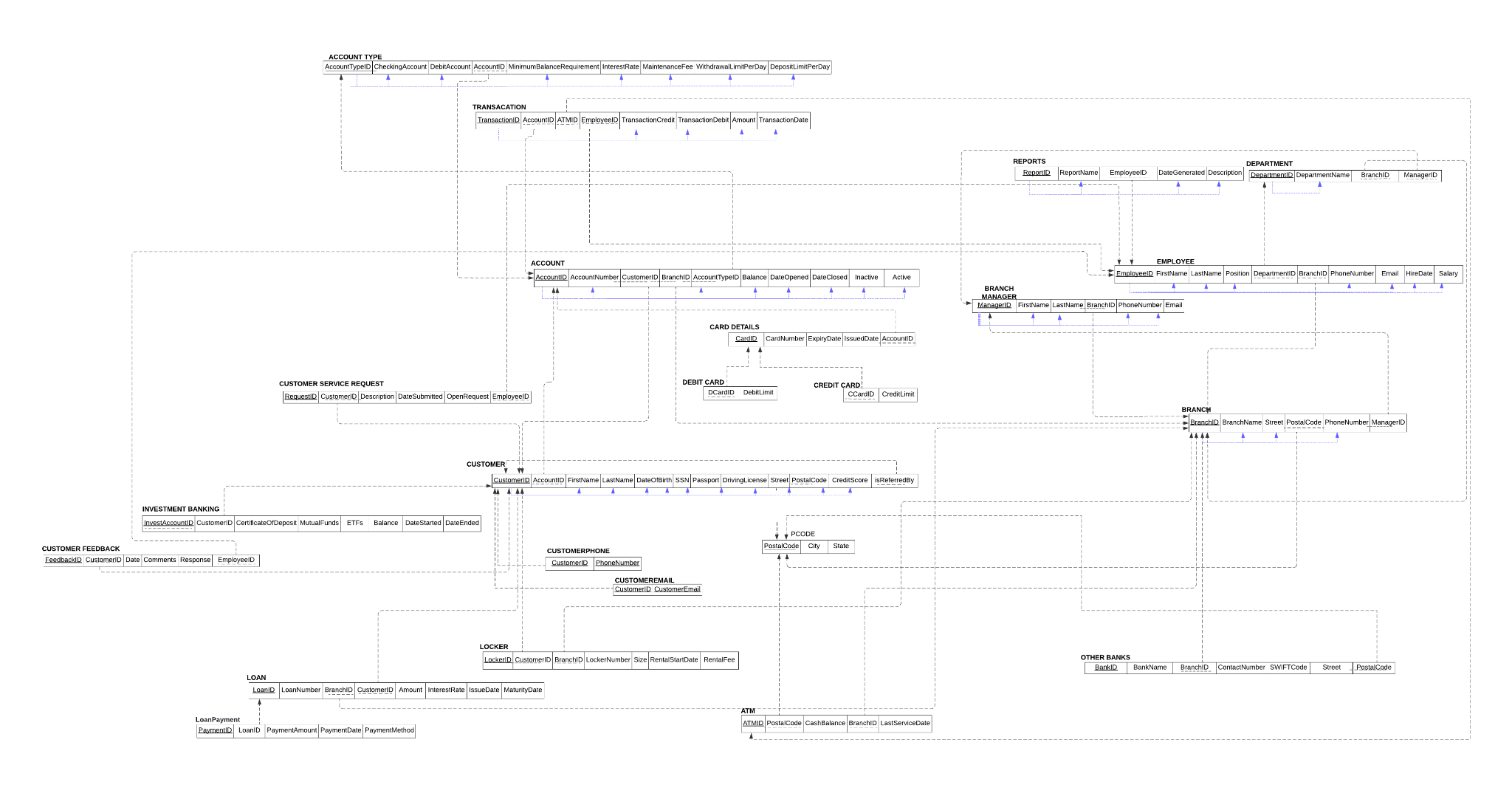
|  |  |  |
| --- | --- | --- |
| **Entity** | **Entity Definition** | **Attributes** |
| Customer | An individual or business entity that maintains an account with a bank | CustomerID (PK), Name,  Date of Birth, IdentificationDocuments,  Address, Phone Number,  Email, CreditScore |
| Account | A financial relationship that a customer holds with a bank | AccountID (PK), AccountNumber, Balance, DateOpened, DateClosed, Status |
| Branch | A physical location of a bank | BranchID (PK), BranchName, Address, PhoneNumber, OpeningDate |
| Transaction | An action performed on an account, such as a deposit, withdrawal, or transfer of funds | TransactionID (PK),TransactionType, Amount, TranscationDate, [BalanceAfterTransaction] |
| Employee | An individual who works for the bank | EmployeeID (PK), Name, Position, PhoneNumber, Email, HireDate, Salary |
| Loan | A financial agreement in which the bank lends money to a customer | LoanID (PK), LoanNumber, Amount , InterestRate, IssueDate, MaturityDate |
| LoanPayment | This entity represents the details related to the payment of loans. | PaymentID (PK), PaymentAmount, PaymentDate, PaymentMethod |
| Locker | This entity represents a secure storage unit provided by the bank for customers to store valuable items. | LockerID (PK), LockerNumber, Size, RentalStartDate, RentalFee |
| Department | This entity refers to different divisions within the bank. | DepartmentID (PK), DepartmentName |
| ATM | This entity represents Automated Teller Machines provided by the bank for customer transactions. | ATMID (PK), CashBalance, LastServiceDate |
| ​​Investment Banking | This entity pertains to the services offered by the bank related to investments. | InvestAccountID (PK), Balance, DateStarted, DateEnded, InvestmentType, |
| Other Banks | This entity represents details of other banks that interact with the primary bank. | BankID (PK), BankName, Address, ContactNumber, SWIFTCode |
| Account Type | This entity defines the different types of accounts offered by the bank. | AccountTypeID (PK), AccountTypeName, MinimumBalanceRequirement, InterestRate, MaintenanceFee, WithdrawalLimitPerDay, DepositLimitPerDay |
| Branch Manager | This entity represents the manager of a bank branch. | ManagerID (PK), Name, PhoneNumber, Email |
| Customer Feedback | This entity captures feedback from customers regarding their experiences with the bank’s services. | FeedbackID (PK), Date, Comments, Response, |
| Reports | This entity refers to various reports generated by the bank for internal and external use. | ReportID (PK), ReportName,  DateGenerated, Description |
| CardDetails | This entity represents the specific attributes and information of a card, distinguished by its type (Debit or Credit), linked to an account. | CardID (PK), CardType, CardNumber, ExpiryDate, IssuedDate |
| Customer Service Request | This entity represents requests made by customers for assistance or services. | RequestID (PK), Description, DateSubmitted, OpenRequest |

*\*\* Note: PK: Primary Key*

### 2.2. Entity-Relationships

1. A Customer can hold multiple Accounts. Each Account is held by one Customer.
2. A Customer can request multiple Customer Service actions. Each Customer Service action is requested by one Customer
3. A Customer can have multiple Loans. Each Loan is held by one Customer
4. An Account Type can include multiple Accounts. Each Account belongs to one Account Type
5. A Transaction occurs in one Account. Each Account can have multiple Transactions
6. A Branch manages multiple Employees. Each Employee is managed by one Branch.
7. A Branch operates multiple Lockers. Each Locker is operated by one Branch
8. A Branch hosts multiple ATMs. Each ATM is hosted by one Branch.
9. A Branch has multiple Departments. Each Department belongs to one Branch
10. A Loan includes multiple Loan Payments. Each Loan Payment is part of one Loan
11. An Investment Banking account belongs to one Customer. Each Customer can have multiple Investment Banking accounts
12. Each Employee can write multiple Reports. Each Report is written by one Employee
13. An Account can have multiple CardDetails; Each CardDetails belongs to one Account

## 3. Relational Model (Logical Diagram): [Link to Relational Model (Logical Diagram)](https://lucid.app/lucidchart/1526d398-3e05-4eb4-b1b0-86a011eb8ca0/edit?viewport_loc=5993%2C-4802%2C5465%2C2980%2COmCJpiiDr1dQ&invitationId=inv_f246b15b-2168-4961-8e0b-d2f8f0018253)



* Primary keys are indicated by solid underlines
* Foreign keys are indicated by dotted underlines
* All the relations in the diagram are normalized to the third normal form

### 3.1 Normalization (Third Normal Form)

* **Ensured Atomicity:** Verified that each attribute held indivisible values, ensuring atomicity. Attributes such as FirstName, LastName, and DateOfBirth in the Customer entity were confirmed to be atomic. Multivalued attributes like Phone Number and Email were separated into their own relations to achieve the First Normal Form (1NF).
* **Eliminated Partial Dependencies:** Decomposed the "Status" relation into separate "Active" and "Inactive" relations. This step removed partial dependencies, ensuring that every non-key attribute is fully functionally dependent on the primary key, thus achieving the Second Normal Form (2NF)
* **Resolved Transitive Dependencies:** Reviewed and addressed transitive dependencies within the relations. Attributes like "PostalCode" and "Street" were normalized by restructuring them into separate relations, eliminating transitive dependencies and bringing the design to the Third Normal Form (3NF)
* **Verified Full Functional Dependencies:** Confirmed that every non-key attribute is fully functionally dependent on the entire primary key. This step was crucial for maintaining data integrity, ensuring each attribute is uniquely determined by the primary key. For example, attributes like Amount, InterestRate, and IssueDate in the Loan entity were confirmed to be fully dependent on LoanID
* **Minimized Redundancy:** Conducted thorough checks to eliminate redundant data across relations, enhancing data consistency and storage efficiency within the relational model. For instance, the Branch entity was designed to prevent duplicate information about branches from being stored in multiple entities

## 4. SQL Database Implementation

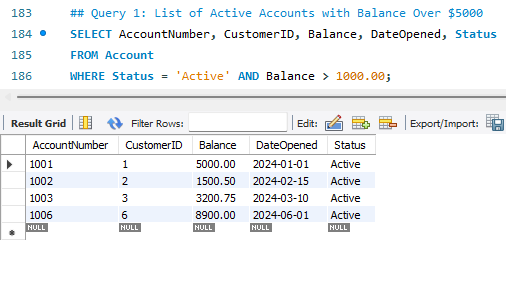
SQL (Structured Query Language) plays a crucial role in the implementation and management of our relational database. It serves as the primary language for interacting with the database, providing a range of functionalities that are essential for the creation, manipulation, and maintenance of data. SQL allows us to create tables with specific columns and data types, ensuring that the data stored is structured and organized.

We have used the INSERT statement to feed sample values into the table for our purposes ([**Link to Sample Data**](https://docs.google.com/spreadsheets/d/1Yv4oIwSD8HUamY2Sak7Dq7O9H_R9ggcYsGxWnDz2JFs/edit?usp=sharing)).

### 4.1. SQL Database Implementation: Without Joins

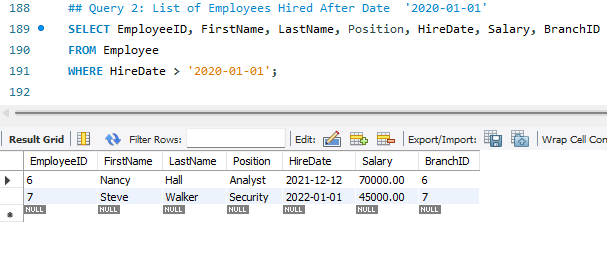
1. **To identify customers active accounts with balance more than $1000**

**Business Justification:** By filtering accounts with a balance greater than $1000, the bank can identify customers who maintain significant balances. These customers are often more valuable to the bank due to the higher potential for revenue through interest and fees.



1. **To identify employees hired after a specific date**

**Business Justification:** The HR department can analyze recent hiring trends and patterns. This helps in understanding the growth and expansion of the workforce. The query provides insights into the salaries of recently hired employees. This information is crucial for budgeting and financial planning, ensuring that salary expenditures are aligned with organizational goals.

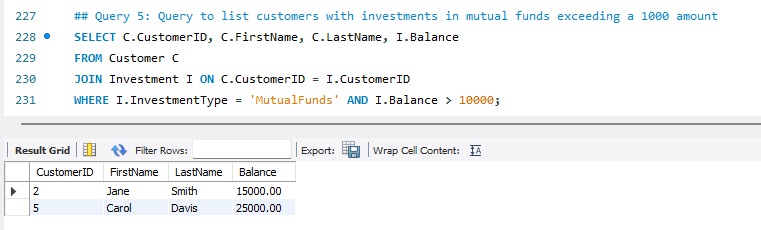
**Since ‘BalanceAfterTransaction’ is d**

### 

### 4.2. SQL Database Implementation: Employing Joins

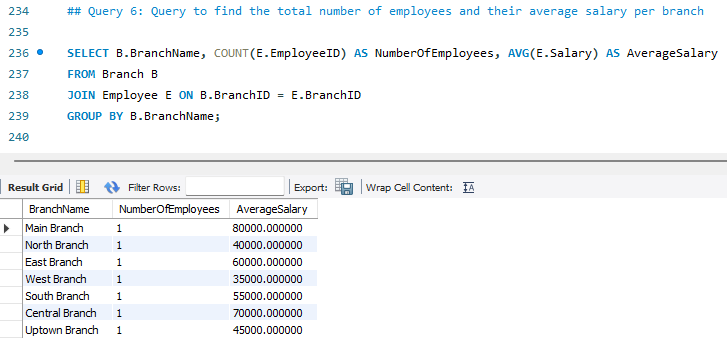
1. **Query to list customers with investment in Mutual Funds exceeding 1000 amount**

**Business Justification:** The result set provides a focused view of high-value customers with significant mutual fund investments, enabling the bank to enhance customer service, offer targeted financial products, and manage risk effectively.



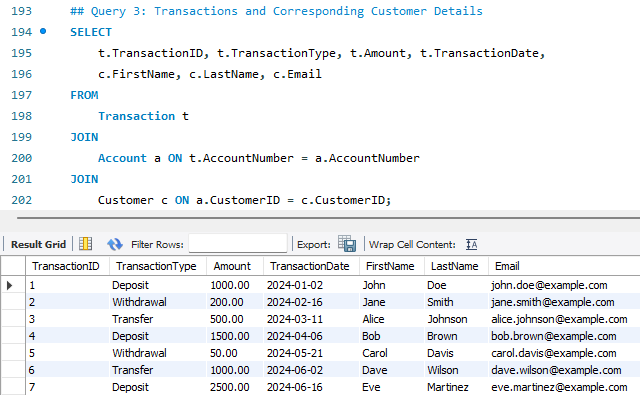
1. **Find no. of employees and their avg salary per branch**

**Business Justification:** Understanding the number of employees per branch and their average salary helps in efficient resource allocation. The bank can ensure that each branch is adequately staffed and that resources are distributed according to the needs of each branch.



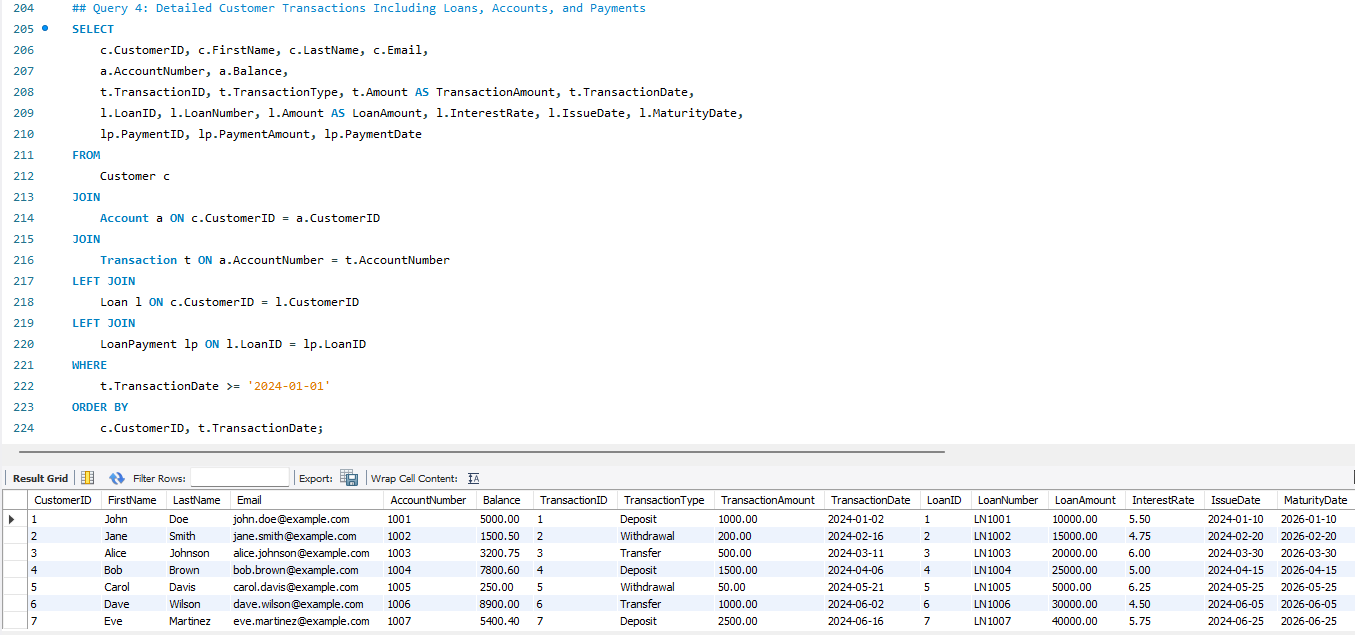
1. **To see customers details and complete transactions**

**Business Justification:** By combining transaction data with customer details, the bank can gain deeper insights into customer behavior and transaction patterns. This information is invaluable for understanding how customers interact with their accounts and which services they use most frequently. The query performs joins to combine data from the Transaction, Account, and Customer tables.



1. **To see customer details and transactions, loans, accounts and payments**

**Business Justification:** By aggregating data from multiple tables (Customer, Account, Transaction, Loan, LoanPayment), the bank gains a holistic view of a customer's financial activities. This comprehensive perspective helps in understanding customer behavior and financial health.



## 5. Administrative Functions

Maintaining a database requires a range of administrative functions to ensure its performance, integrity, security, and availability. Here are some key administrative functions:

**Database Backup and Recovery**

* Schedule and perform backups to ensure data restoration in case of loss or corruption.
* Develop and test disaster recovery plans for quick restoration after failures.

**Performance Monitoring and Tuning:**

* Analyze and optimize SQL queries for better performance.
* Create and maintain indexes to improve data retrieval speed.
* Monitor and adjust CPU, memory, and disk space for optimal performance.

**Security Management:**

* Implement role-based access control (RBAC) for appropriate user permissions.
* Encrypt sensitive data at rest and in transit to prevent unauthorized access.
* Maintain audit logs to track transactions and access for monitoring and compliance.

**Data Integrity and Consistency:**

* Use primary keys, foreign keys, and unique constraints to maintain integrity.
* Implement rules to ensure data meets required standards and formats.

### 5.1 Potential Issues and Measures to Address

1. **Data Integrity Issues:** Prevent inconsistent or duplicate entries and unauthorized modifications by normalizing the database, enforcing foreign key constraints, and implementing unique constraints.
2. **Data Security and Privacy:** Prevent unauthorized access and data breaches by using role-based access control (RBAC), encrypting data, and maintaining audit trails.
3. **Performance and Scalability:** Improve query performance and handle large data volumes by using appropriate indexes, partitioning tables, and implementing load balancing.
4. **Backup and Recovery:** Prevent data loss by performing regular backups and testing disaster recovery plans regularly.

## 6. Conclusion

The Bank Management Database System (BMDS) aims to revolutionize banking operations by ensuring accuracy, consistency, and ease of access to information. By centralizing data management, the BMDS will keep information up-to-date across all branches and departments, reducing redundancy and enhancing data integrity. This will lead to efficient and personalized customer service, improved workflows in account and loan management, and better decision-making.

The system will handle all types of transactions accurately and in real-time, preventing fraud and enhancing the reliability of financial operations. Regular maintenance and administrative functions, including performance monitoring, backup and recovery, and security management, will ensure the system remains efficient and secure.

Overall, the BMDS will enhance operational efficiency, customer service, and support strategic decision-making while ensuring robust security and compliance, significantly optimizing the bank's operations and services.