**BANK OF SAIT  
ARTICLE  
Written by: Amritpal Singh**

**Introduction:**

SAIT Bank is a trusted financial institution dedicated to providing secure, innovative, and customer-focused banking solutions, including savings, loans, and digital transaction services.

**Mission Statement**

To design and maintain a reliable, efficient, secure database system to serve all banking operations in handling customer data, financial transactions, and business processes for optimal service delivery, regulatory compliance, and decision-making.

**OBJECTIVES:**

**1.Centralized Data Management:** Ensure all customers, account, and transaction data are stored in a unified and easily accessible system to avoid redundancy and maintain data integrity.

**2.Enhance Transaction Accuracy and Security:** Automate the recording and tracking of deposits, withdrawals, and transfers while safeguarding sensitive data against unauthorized access.

**3.Improve Operational Efficiency**: Automate routine tasks such as account management, transaction recording, and loan processing to minimize errors and save time.

**4.Enable Data-Driven Decision-Making**: Use the database to generate insights through analytics, such as identifying trends in customer behavior, evaluating loan performance, and optimizing branch operations for strategic growth.

**BUSINESS TRANSACTION RULES:**

* + A transaction must be fully completed or fully rolled back if any part fails.
  + Transactions must maintain data integrity and follow banking constraints
  + An account must exist and be active before any transaction can be made.
  + Transactions should expire if not completed within a certain time
  + Transactions above a certain limit must trigger an automatic security check.
  + Transactions should be processed in real time for customer satisfaction.

**Data Dictionary**

**PRELIMINARY LIST OF TABLES:**

•BRANCH

•Accounts

•Customers

•Bankers

•Loan

•Borrowers

• Transactions

• Loan Payments

•Credit Cards

**FINAL LIST OF TABLES:**

•BRANCH

•Accounts

•Customers

•Employees

•Loan

•Transactions

• Credit cards  
  
**1.Branches**

|  |  |
| --- | --- |
| Branches |  |
| Branchid | int |
| BranchName | Char |
| BranchAddress | Varchar |
| Asset | decimal |

**2.Account:**

|  |  |
| --- | --- |
| Account |  |
| AccountId | int |
| AccountBalance | Decimal |
| AccountType | varchar |

**3.Employee:**

|  |  |
| --- | --- |
| Emoloyee |  |
| EmployeeId | int |
| Employee Name | varchar |
| Job title | varchar |
| Address | text |
| EmailId | varchar |
| HireDate | date |

**4.Transaction:**

|  |  |
| --- | --- |
| Transaction |  |
| TransactionId | int |
| TransactionType | varchar |
| TransactionMethod | varchar |
| TransactionDate | date |
| Amount | decimal |

**5.Loans:**

|  |  |
| --- | --- |
| Loans |  |
| LoanID | int |
| IssuedAmount | Decimal |
| RemainingAmount | decimal |

**6.Customer:**

|  |  |
| --- | --- |
| Customer |  |
| CustomerId | int |
| CustomerName | varchar |
| DateOfBirth | Date |
| MobileNumber | Varchar |
| Proofid | varchar |

**7.Credit Card**

|  |  |
| --- | --- |
| Credit Card |  |
| CreditCardId | int |
| ExpiryDate | date |
| CardLimit | decimal |

**Explanation:   
1. Branches Table**

* BranchID (int): Unique identifier for each bank branch.
* BranchName (char): Name of the branch.
* BranchAddress (varchar): Physical location of the branch.
* Asset (decimal): Total financial assets held by the branch.

**2. Account Table**

* AccountID (int): Unique identifier for a bank account.
* AccountBalance (decimal): Current balance available in the account.
* AccountType (varchar): Type of account (e.g., savings, checking).

**3. Employee Table**

* EmployeeID (int): Unique identifier for each employee.
* EmployeeName (varchar): Full name of the employee.
* JobTitle (varchar): Role or designation of the employee in the bank (e.g., manager, teller).
* Address (text): Residential address of the employee.
* EmailID (varchar): Contact email address of the employee.
* HireDate (date): Date the employee was hired.

**4. Transaction Table**

* TransactionID (int): Unique identifier for each transaction.
* TransactionType (varchar): Type of transaction (e.g., deposit, withdrawal).
* TransactionMethod (varchar): Mode of transaction (e.g., online, cash).
* TransactionDate (date): Date the transaction was made.
* Amount (decimal): Monetary value involved in the transaction.

**5. Loans Table**

* LoanID (int): Unique identifier for each loan issued.
* IssuedAmount (decimal): Total amount issued for the loan.
* RemainingAmount (decimal): Balance left to be paid for the loan.

**6. Customer Table**

* CustomerID (int): Unique identifier for each customer.
* CustomerName (varchar): Full name of the customer.
* DateOfBirth (date): Customer’s date of birth.
* MobileNumber (varchar): Contact phone number of the customer.
* ProofID (varchar): Identity proof submitted by the customer (e.g., driver's license).

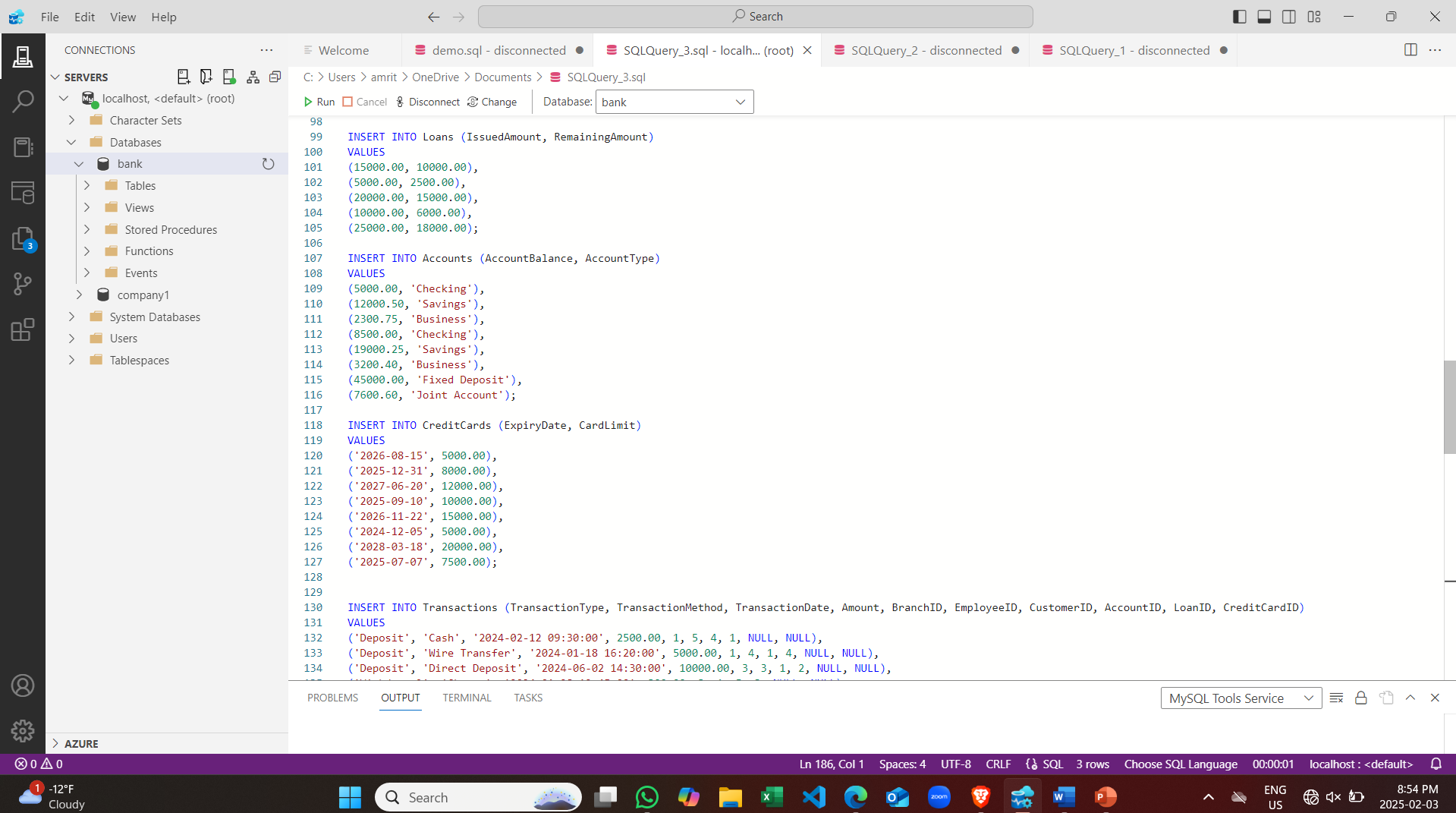
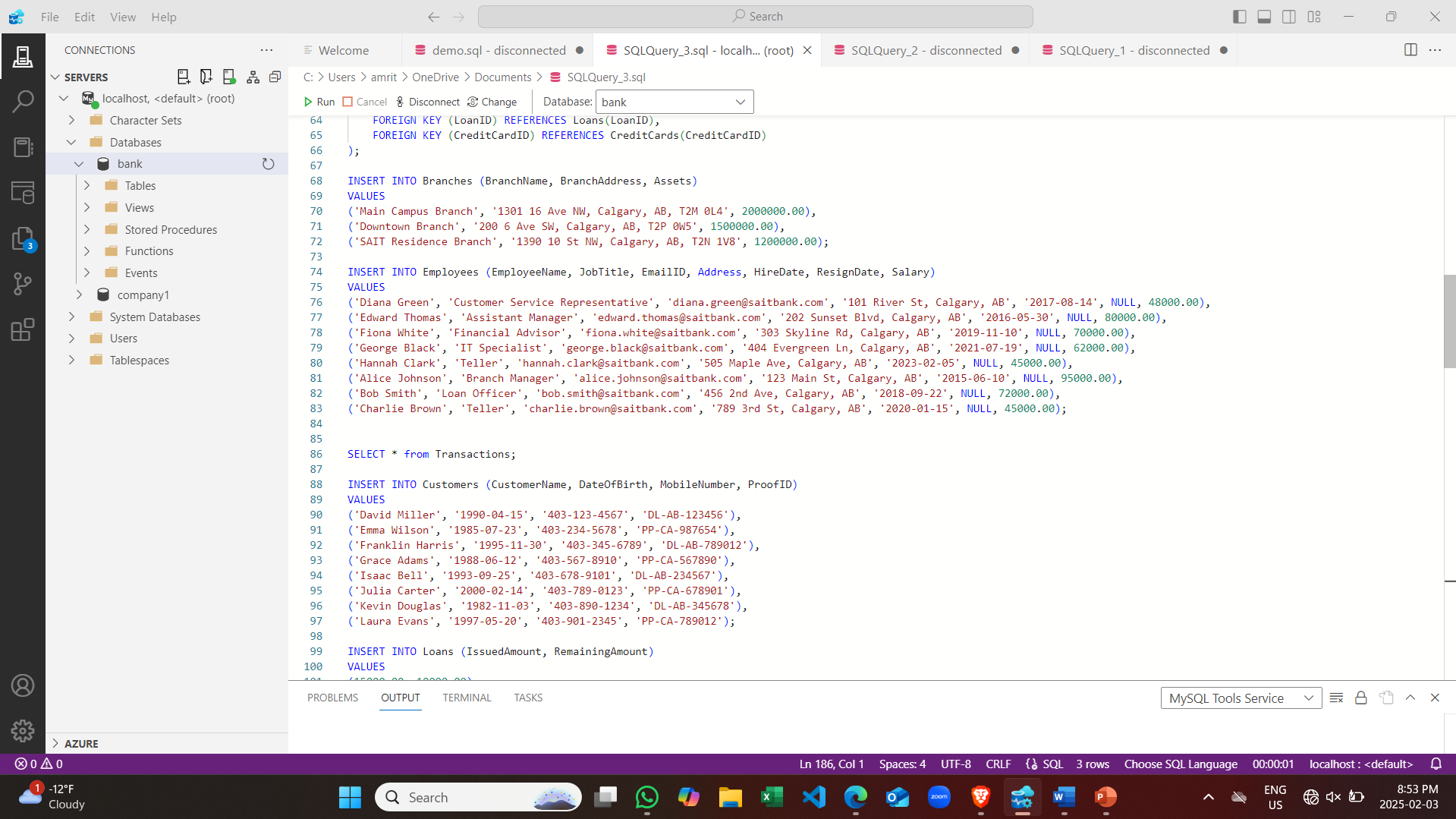
**7. Credit Card Table**

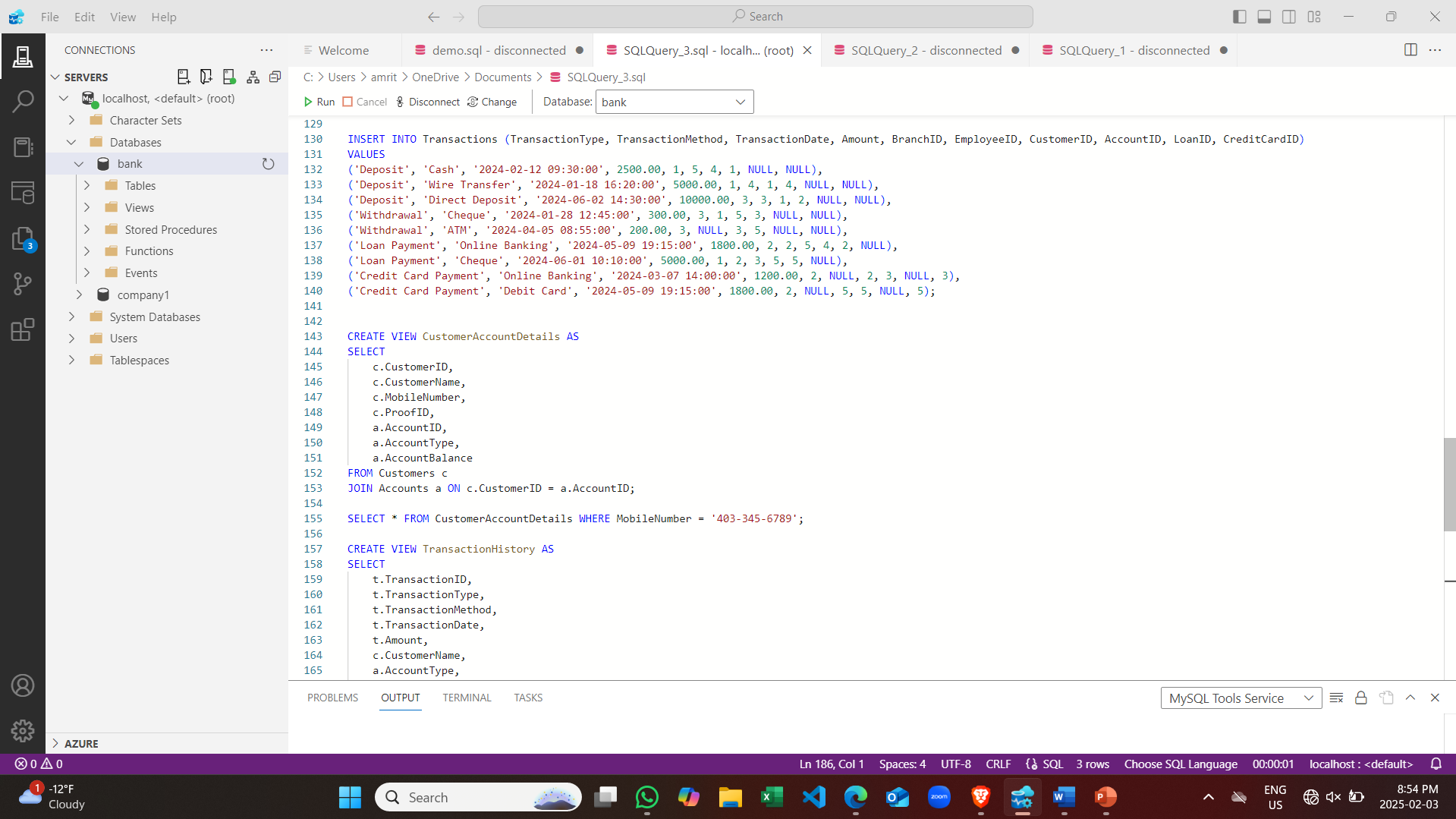
* CreditCardID (int): Unique identifier for each credit card.
* ExpiryDate (date): Expiration date of the credit card.
* CardLimit (decimal): Maximum spending limit of the credit card.

**RELATIONSHIPS:**

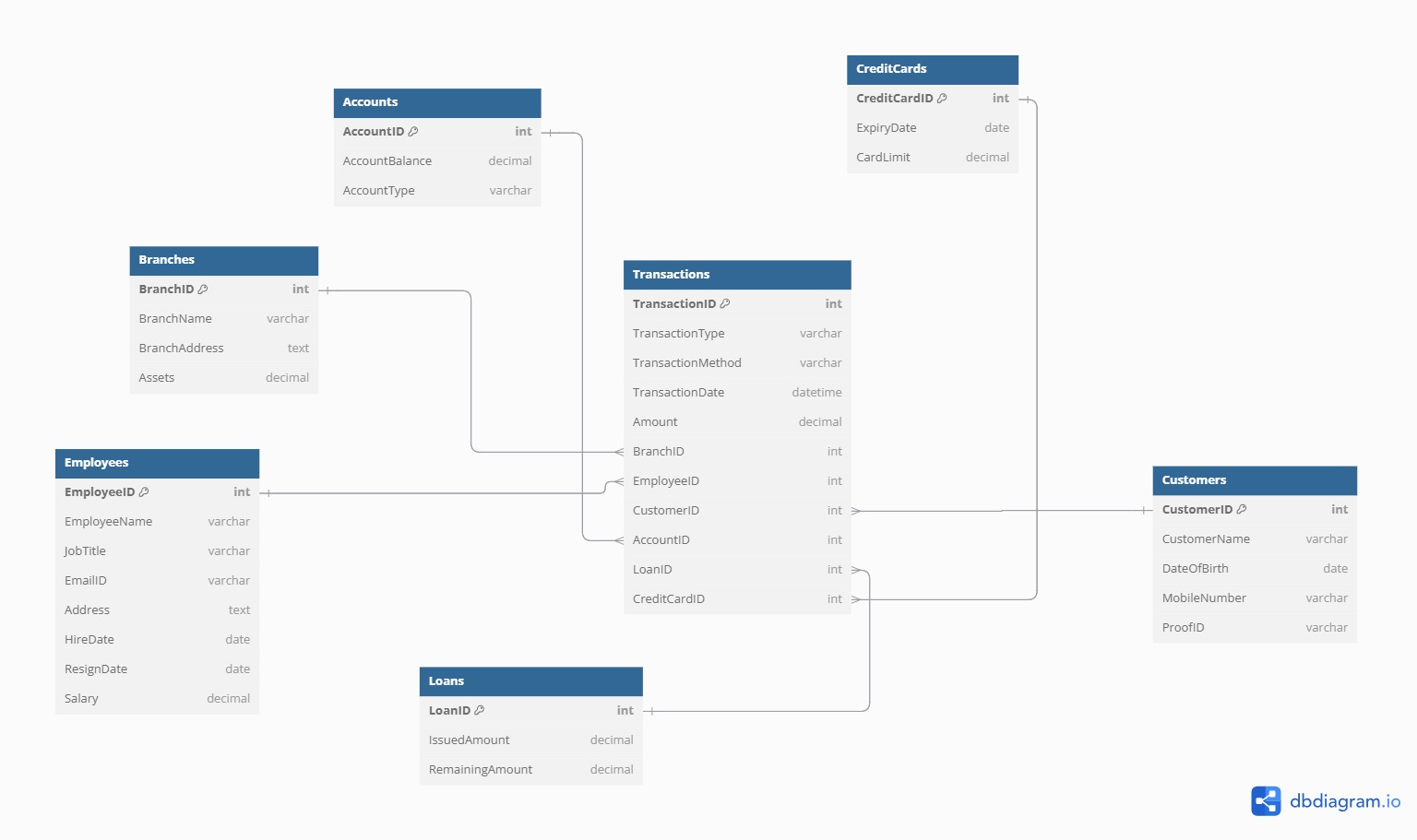
|  |  |  |  |
| --- | --- | --- | --- |
| Relationship | Parent table | Child Table | Relationship Type |
| One Branch to Many Transactions | Branches | Transactions | One-to-Many (1:N) |
| One Employee to Many Transactions | Employees | Transactions | One-to-Many (1:N) |
| One Customer to Many Transactions | Customers | Transactions | One-to-Many (1:N) |
| One Account to Many Transactions | Accounts | Transactions | One-to-Many (1:N) |
| One Loan to Many Transactions | Loans | Transactions | One-to-Many (1:N) |
| One Credit Card to Many Transactions | Credit Cards | Transactions | One-to-Many (1:N) |

**Designing database:**





**Entity Relationship Diagram**



**Conclusion:**This project focused on building a secure and efficient database system for banking operations. It ensures centralized data management, improves transaction accuracy, and automates routine tasks like account management and loan processing. The system supports real-time transactions, protects data, and follows banking rules to maintain integrity. With a well-structured table design, it helps the bank manage branches, accounts, customers, loans, and transactions efficiently while enabling better decision-making and growth.