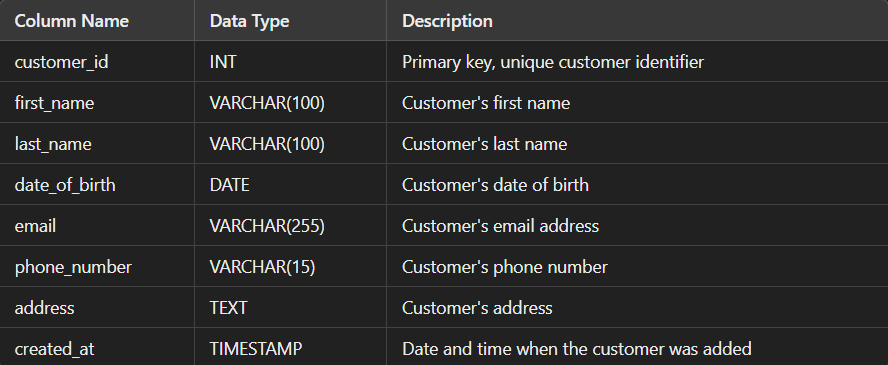
ASSIGNMENT 1

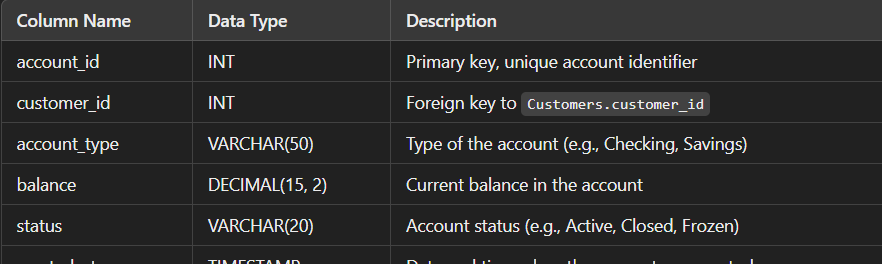
1. **Customers Table**

**This table stores information about each customer.**



2.Account Table

This table stores information about each account held by a customer.

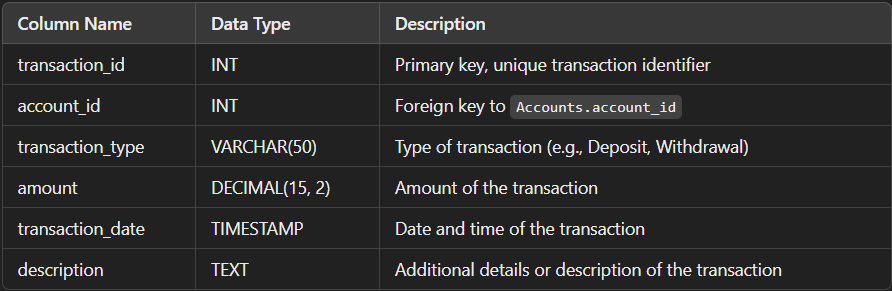


**Relationships:**

* Accounts.customer\_id references Customers.customer\_id (one-to-many: one customer can have many accounts).

3.Transaction Table

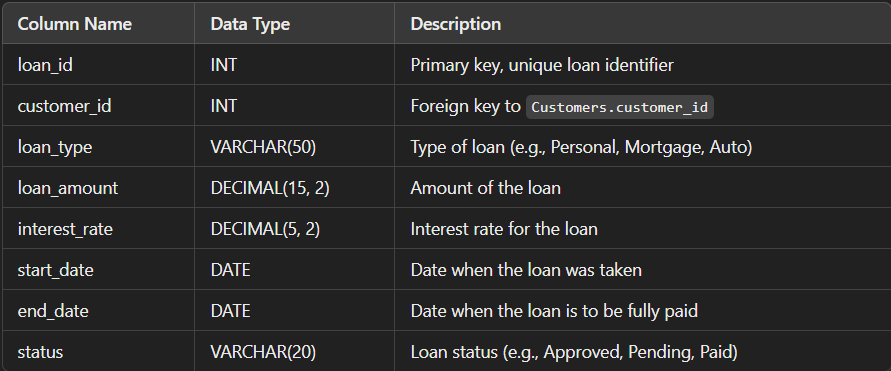
This table stores information about all transactions made on accounts.



**Relationships:**

* Transactions.account\_id references Accounts.account\_id (one-to-many: one account can have many transactions).
* 4.Loan Table

This table stores details about loans given to customers.

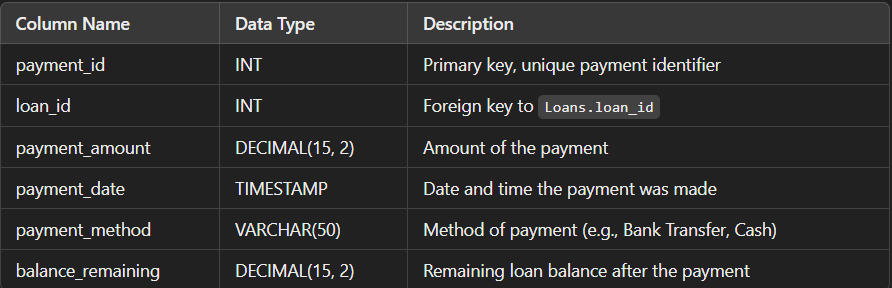


**Relationships:**

* Loans.customer\_id references Customers.customer\_id (one-to-many: one customer can have many loans).

5.Loan Payments Table

This table stores information about payments made towards loans.

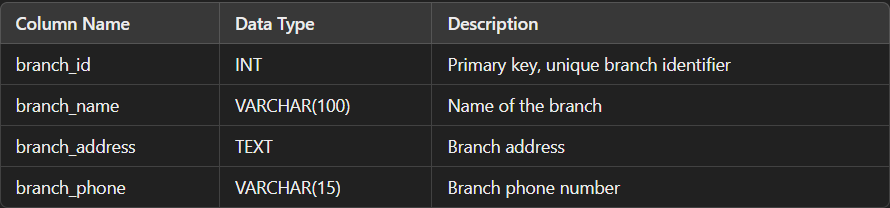


**Relationships:**

* Loan Payments.loan\_id references Loans.loan\_id (one-to-many: one loan can have many payments).

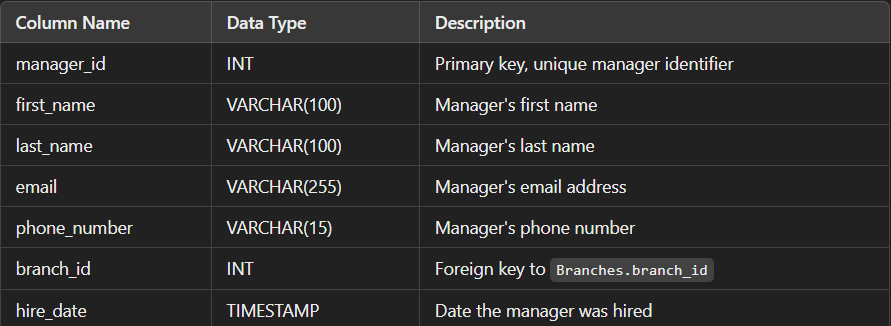
6.Branches Table

This table stores details about the bank branches.



7.Account Managers Table

This table stores information about bank employees who manage accounts and customer relationships.



**Relationships:**

* Account Managers.branch\_id references Branches.branch\_id (one-to-many: one branch can have many account managers).

**Example Relational Model:**

* **Customers** can have multiple **Accounts** (one-to-many relationship).
* **Accounts** can have multiple **Transactions** (one-to-many relationship).
* **Customers** can have multiple **Loans** (one-to-many relationship).
* **Loans** can have multiple **Loan Payments** (one-to-many relationship).
* Each **Account Manager** works at a **Branch**, but one branch can have multiple managers (one-to-many relationship).