EduFin Credit Solutions –

Synthetic Loan Dataset - Technical Documentation

{8 EduFin Credit Solutions

1. Introduction

‘The **EduFin Credit Solutions** dataset is a comprehensive synthetic loan ecosystem designed specifically for corporate analytics training and educational purposes. This end-to-end simulation encompasses the complete student loan lifecycle, from initial application through disbursement, repayment, and potential default scenarios.

Dataset Scope

This synthetic dataset provides realistic financial data patterns while maintaining complete privacy compliance. It covers:

‘+ Educational Institution Partnerships - University and college collaboration details

‘+ Customer Onboarding - KYC, demographics, and financial profiles

+ Loan Lifecycle Management - Application, approval, disbursement, and terms

‘+ Payment Processing - EMI transactions, delays, and payment behaviors

Default Management - Risk assessment, collection stages, and recovery analytics

Key Benefits

‘+ Risk-Free Training Environment - Practice with realistic date without privacy concerns

‘+ Complete Ecosystem View - End-to-end loan operations in a single dataset

‘+ Scalable Analytics - Suitable for both individual leaning and team training

‘+ Industry-Standard Structure - Mirrors real-world financial data architectures

‘il 2. Dataset Architecture

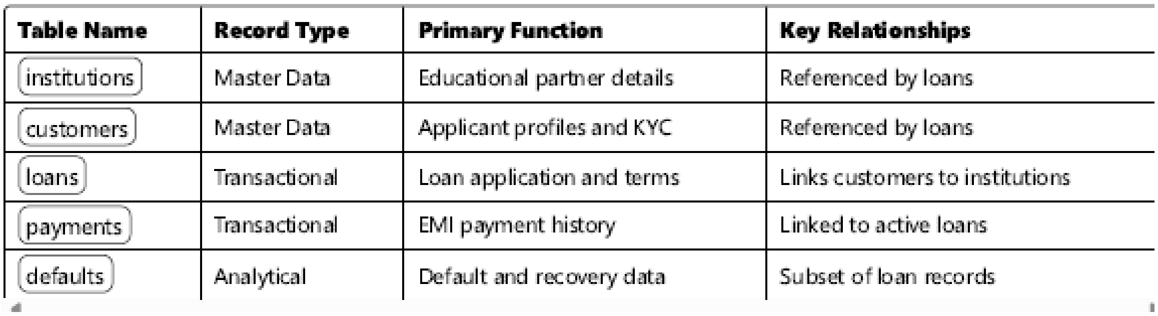


Table Name	Record Type	Primary Function	Key Relationships
institutions	Master Data	Educational partner details	Referenced by loans
customers	Master Data	Applicant profiles and KYC	Referenced by loans
loans	Transactional	Loan application and terms	Links customers to institutions
payments	Transactional	EMI payment history	Linked to active loans
defaults	Analytical	Default and recovery data	Subset of loan records

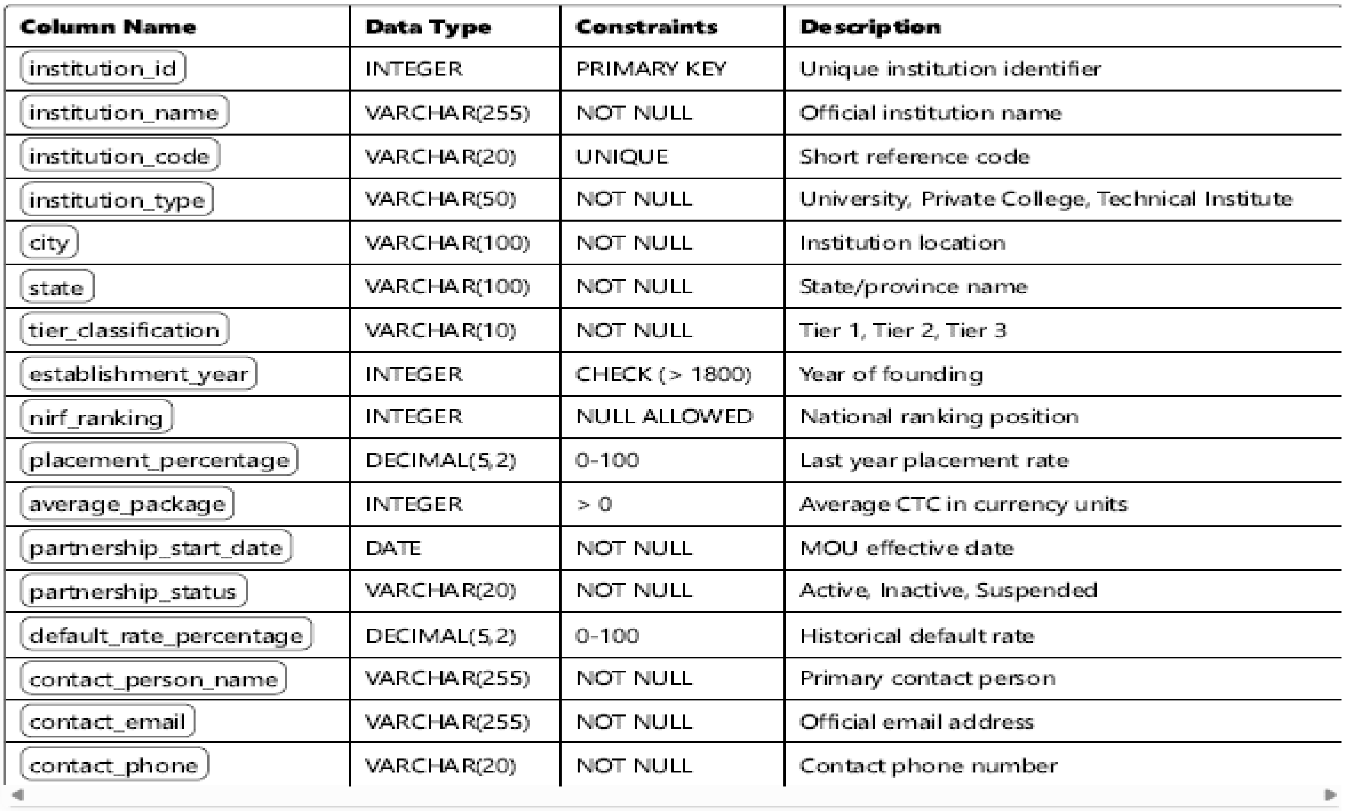
2.2 Data Volume Estimates

- **Institutions**: 500–1,000 records
- **Customers**: 10,000–50,000 records
- **Loans**: 8,000–40,000 records
- **Payments**: 100,000–500,000 records
- **Defaults**: 1,000–5,000 records

3. Table Schema Specifications

3.1 Institutions Table

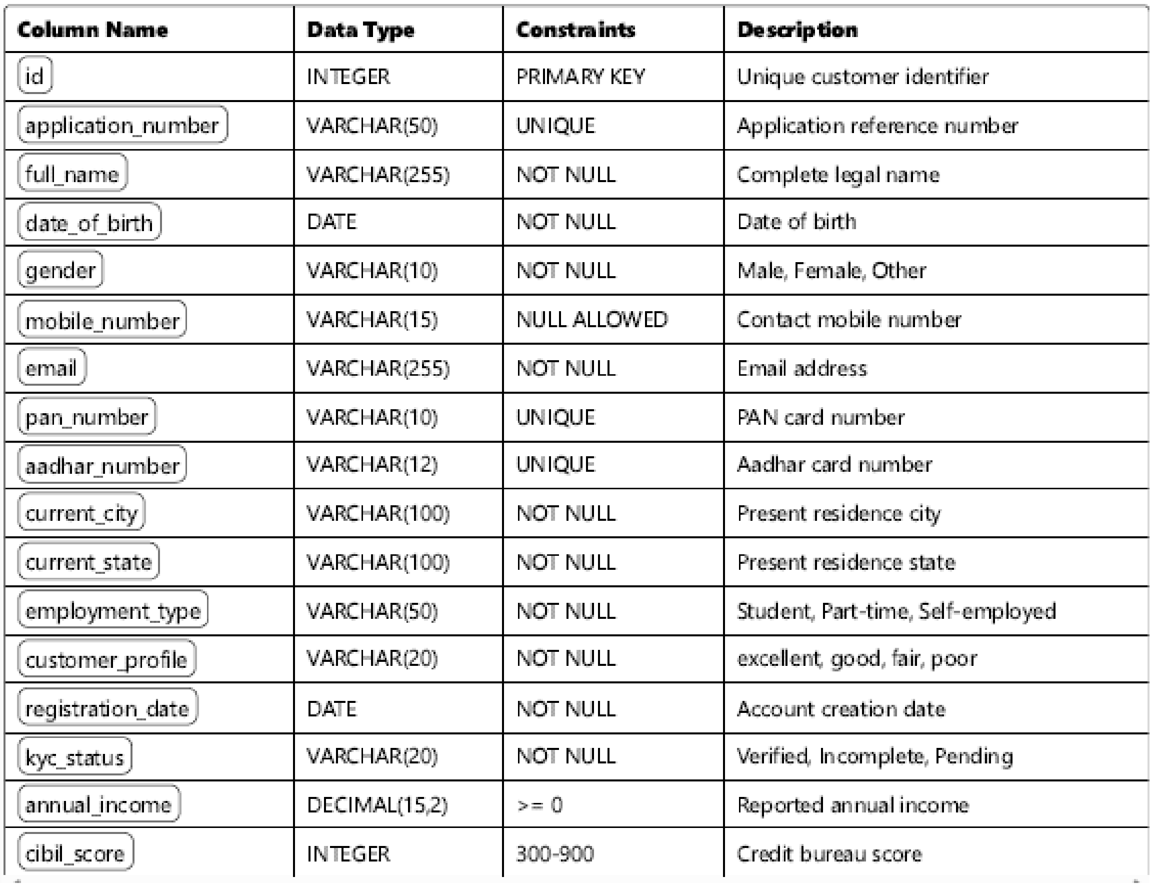
Educational institutions in partnership with EduFin



Column Name	Data Type	Constraints	Description
institution_id	INTEGER	PRIMARY KEY	Unique institution identifier
institution_name	VARCHAR(255)	NOT NULL	Official institution name
institution_code	VARCHAR(20)	UNIQUE	Short reference code
institution_type	VARCHAR(50)	NOT NULL	University, Private College, Technical Institute
city	VARCHAR(100)	NOT NULL	Institution location
state	VARCHAR(100)	NOT NULL	State/province name
tier_classification	VARCHAR(10)	NOT NULL	Tier 1, Tier 2, Tier 3
establishment_year	INTEGER	CHECK (> 1800)	Year of founding
nirf_ranking	INTEGER	NULL ALLOWED	National ranking position
placement_percentage	DECIMAL(5,2)	0-100	Last year placement rate
average_package	INTEGER	> 0	Average CTC in currency units
partnership_start_date	DATE	NOT NULL	MOU effective date
partnership_status	VARCHAR(20)	NOT NULL	Active, Inactive, Suspended
default_rate_percentage	DECIMAL(5,2)	0-100	Historical default rate
contact_person_name	VARCHAR(255)	NOT NULL	Primary contact person
contact_email	VARCHAR(255)	NOT NULL	Official email address
contact_phone	VARCHAR(20)	NOT NULL	Contact phone number

3.2 Customer Table

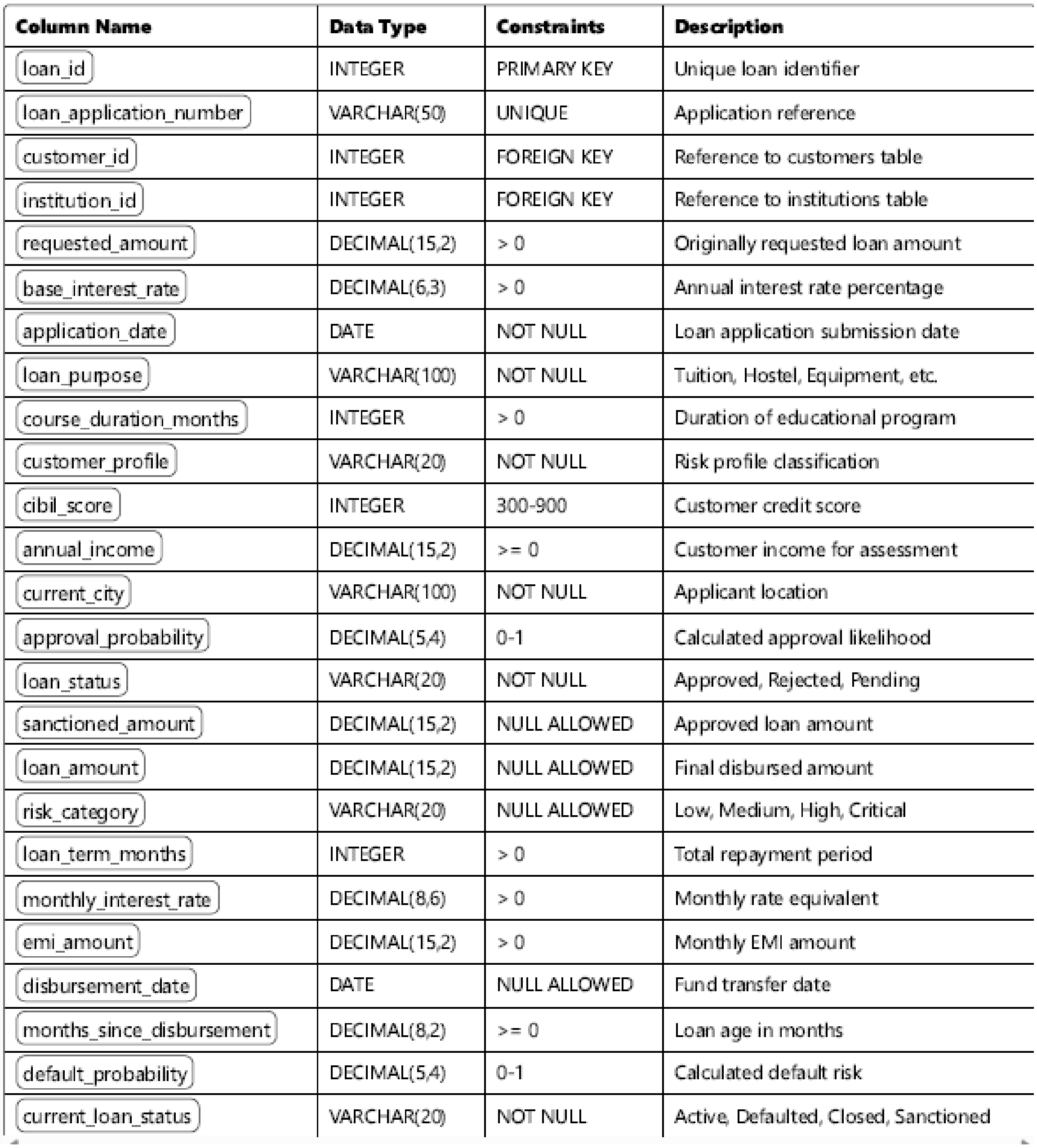
Student loan applicant profiles and KYC data.



Column Name	Data Type	Constraints	Description
id	INTEGER	PRIMARY KEY	Unique customer identifier
application_number	VARCHAR(50)	UNIQUE	Application reference number
full_name	VARCHAR(255)	NOT NULL	Complete legal name
date_of_birth	DATE	NOT NULL	Date of birth
gender	VARCHAR(10)	NOT NULL	Male, Female, Other
mobile_number	VARCHAR(15)	NULL ALLOWED	Contact mobile number
email	VARCHAR(255)	NOT NULL	Email address
pan_number	VARCHAR(10)	UNIQUE	PAN card number
aadhar_number	VARCHAR(12)	UNIQUE	Aadhar card number
current_city	VARCHAR(100)	NOT NULL	Present residence city
current_state	VARCHAR(100)	NOT NULL	Present residence state
employment_type	VARCHAR(50)	NOT NULL	Student, Part-time, Self-employed
customer_profile	VARCHAR(20)	NOT NULL	excellent, good, fair, poor
registration_date	DATE	NOT NULL	Account creation date
kyc_status	VARCHAR(20)	NOT NULL	Verified, Incomplete, Pending
annual_income	DECIMAL(15,2)	>= 0	Reported annual income
cibil_score	INTEGER	300-900	Credit bureau score

3.3 Loans Table

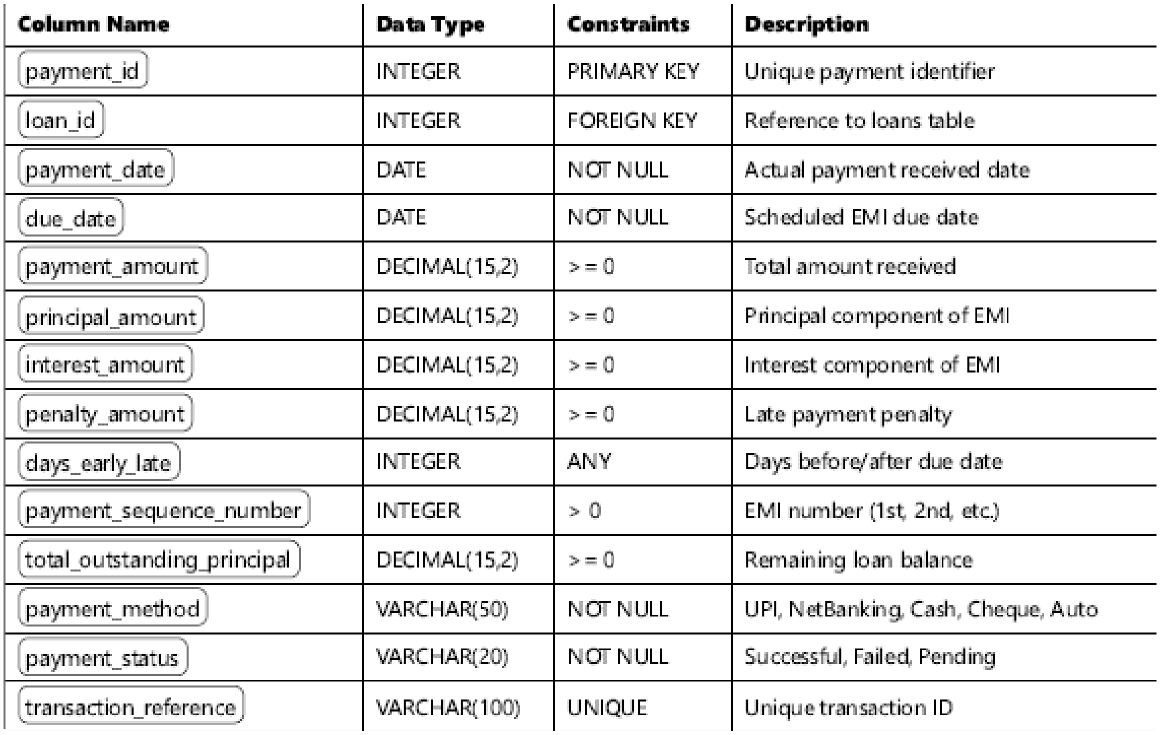
Complete loan application approval, and disbursement records



Column Name	Data Type	Constraints	Description
loan_id	INTEGER	PRIMARY KEY	Unique loan identifier
loan_application_number	VARCHAR(50)	UNIQUE	Application reference
customer_id	INTEGER	FOREIGN KEY	Reference to customers table
institution_id	INTEGER	FOREIGN KEY	Reference to institutions table
requested_amount	DECIMAL(15,2)	> 0	Originally requested loan amount
base_interest_rate	DECIMAL(6,3)	> 0	Annual interest rate percentage
application_date	DATE	NOT NULL	Loan application submission date
loan_purpose	VARCHAR(100)	NOT NULL	Tuition, Hostel, Equipment, etc.
course_duration_months	INTEGER	> 0	Duration of educational program
customer_profile	VARCHAR(20)	NOT NULL	Risk profile classification
cibil_score	INTEGER	300-900	Customer credit score
annual_income	DECIMAL(15,2)	>= 0	Customer income for assessment
current_city	VARCHAR(100)	NOT NULL	Applicant location
approval_probability	DECIMAL(5,4)	0-1	Calculated approval likelihood
loan_status	VARCHAR(20)	NOT NULL	Approved, Rejected, Pending
sanctioned_amount	DECIMAL(15,2)	NULL ALLOWED	Approved loan amount
loan_amount	DECIMAL(15,2)	NULL ALLOWED	Final disbursed amount
risk_category	VARCHAR(20)	NULL ALLOWED	Low, Medium, High, Critical
loan_term_months	INTEGER	> 0	Total repayment period
monthly_interest_rate	DECIMAL(8,6)	> 0	Monthly rate equivalent
emi_amount	DECIMAL(15,2)	> 0	Monthly EMI amount
disbursement_date	DATE	NULL ALLOWED	Fund transfer date
months_since_disbursement	DECIMAL(8,2)	>= 0	Loan age in months
default_probability	DECIMAL(5,4)	0-1	Calculated default risk
current_loan_status	VARCHAR(20)	NOT NULL	Active, Defaulted, Closed, Sanctioned

3.4 Payments Table

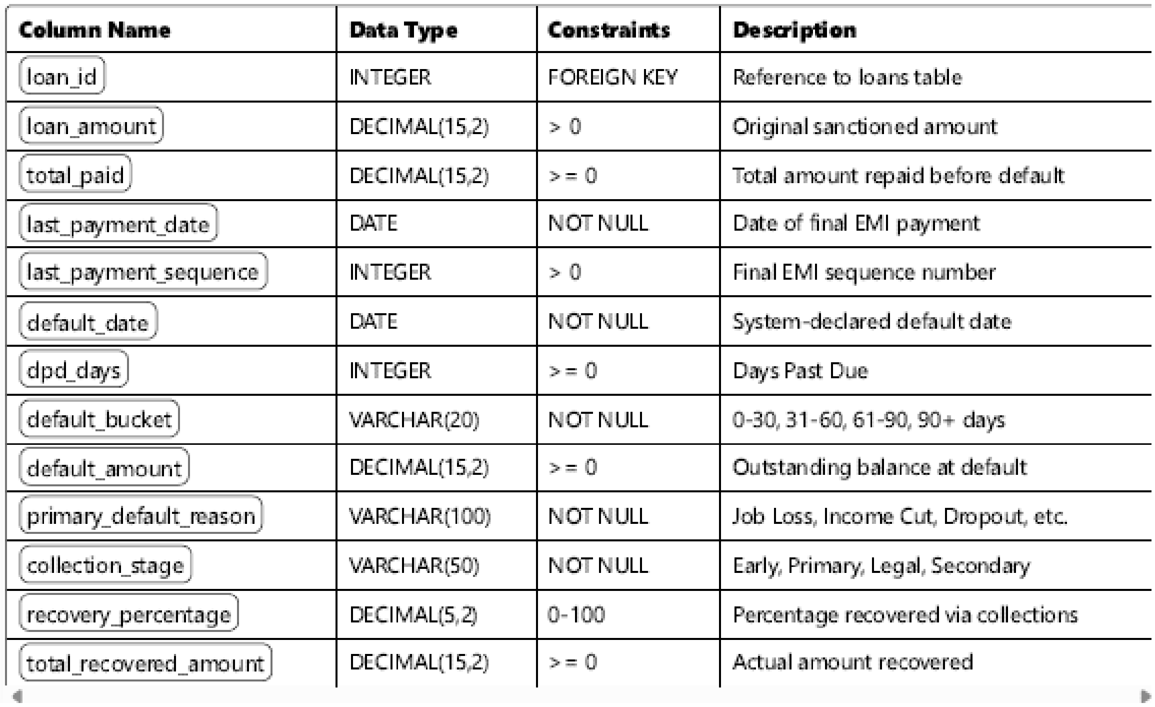
EMI payment transaction history



Column Name	Data Type	Constraints	Description
payment_id	INTEGER	PRIMARY KEY	Unique payment identifier
loan_id	INTEGER	FOREIGN KEY	Reference to loans table
payment_date	DATE	NOT NULL	Actual payment received date
due_date	DATE	NOT NULL	Scheduled EMI due date
payment_amount	DECIMAL(15,2)	> = 0	Total amount received
principal_amount	DECIMAL(15,2)	> = 0	Principal component of EMI
interest_amount	DECIMAL(15,2)	> = 0	Interest component of EMI
penalty_amount	DECIMAL(15,2)	> = 0	Late payment penalty
days_early_late	INTEGER	ANY	Days before/after due date
payment_sequence_number	INTEGER	> 0	EMI number (1st, 2nd, etc.)
total_outstanding_principal	DECIMAL(15,2)	> = 0	Remaining loan balance
payment_method	VARCHAR(50)	NOT NULL	UPI, NetBanking, Cash, Cheque, Auto
payment_status	VARCHAR(20)	NOT NULL	Successful, Failed, Pending
transaction_reference	VARCHAR(100)	UNIQUE	Unique transaction ID

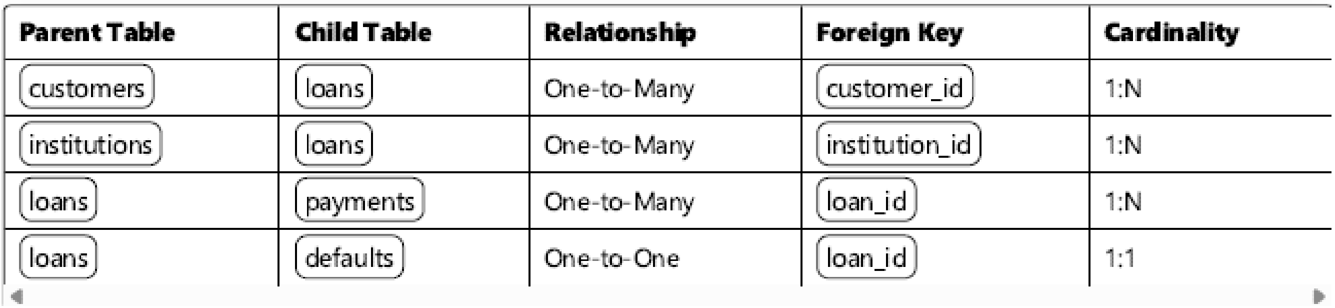
3.5 Defaults Table

Default classification and recovery management

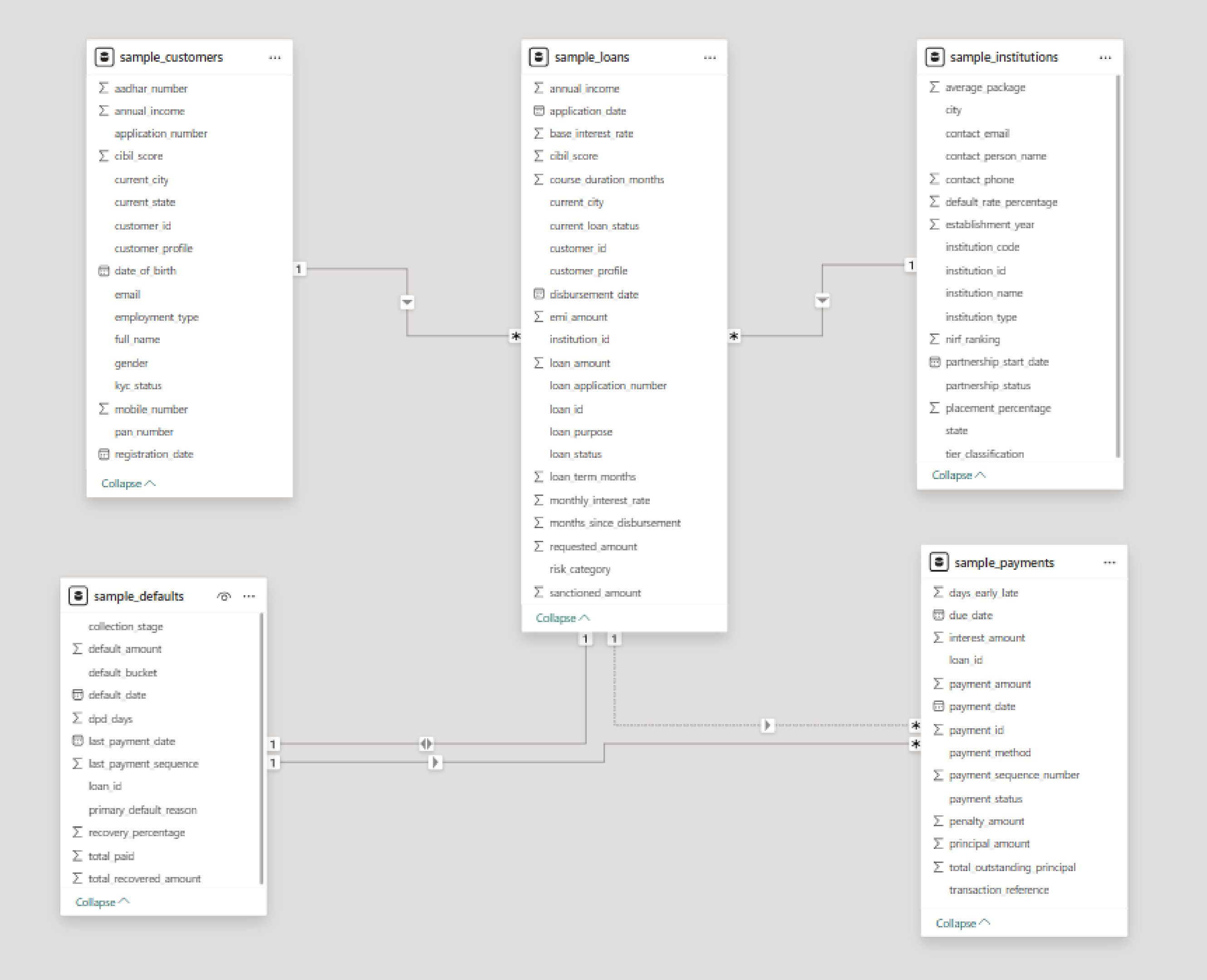


Column Name	Data Type	Constraints	Description
loan_id	INTEGER	FOREIGN KEY	Reference to loans table
loan_amount	DECIMAL(15,2)	> 0	Original sanctioned amount
total_paid	DECIMAL(15,2)	> = 0	Total amount repaid before default
last_payment_date	DATE	NOT NULL	Date of final EMI payment
last_payment_sequence	INTEGER	> 0	Final EMI sequence number
default_date	DATE	NOT NULL	System-declared default date
dpd_days	INTEGER	> = 0	Days Past Due
default_bucket	VARCHAR(20)	NOT NULL	0-30, 31-60, 61-90, 90+ days
default_amount	DECIMAL(15,2)	> = 0	Outstanding balance at default
primary_default_reason	VARCHAR(100)	NOT NULL	Job Loss, Income Cut, Dropout, etc.
collection_stage	VARCHAR(50)	NOT NULL	Early, Primary, Legal, Secondary
recovery_percentage	DECIMAL(5,2)	0-100	Percentage recovered via collections
total_recovered_amount	DECIMAL(15,2)	> = 0	Actual amount recovered

4.2 Relationship Details



Parent Table	Child Table	Relationship	Foreign Key	Cardinality
customers	loans	One-to-Many	customer_id	1:N
institutions	loans	One-to-Many	institution_id	1:N
loans	payments	One-to-Many	loan_id	1:N
loans	defaults	One-to-One	loan_id	1:1



Entity-Relationship Diagram: Theoretical Explanation

The ER diagram represents a relational model for a student loan ecosystem, capturing the key entities and their relationships involved in the loan lifecycle — from application to repayment and default. The system is designed for data analytics, reporting, and machine learning use cases in the educational finance domain.

---

1. Entities and Attributes

a. Institutions

This table holds master data for partner educational institutions.

- **Primary Key: institution_id**

- **Attributes: institution_name, institution_type, tier_classification, partnership_status, default_rate_%**
- **Purpose: Links each loan to the corresponding institution, allowing institution-level risk and performance analysis.**

b. Customers

Represents loan applicants and students.

- **Primary Key: id**
- **Attributes: application_no, full_name, cibil_score, annual_income, kyc_status**
- **Purpose: Captures applicant details including creditworthiness (CIBIL score) and income for underwriting purposes.**

c. Loans

Stores transactional information about each loan.

- **Primary Key: loan_id**
- **Foreign Keys: customer_id (→ Customers), institution_id (→ Institutions)**
- **Attributes: loan_amount, loan_status, emi_amount**
- **Purpose: Core table that connects customers to institutions and tracks the disbursement and status of loans.**

d. Payments

Records EMI transactions made against active loans.

- **Primary Key: payment_id**
- **Foreign Key: loan_id (→ Loans)**
- **Attributes: payment_date, payment_amount, payment_status**
- **Purpose: Enables repayment tracking, payment behavior analysis, and early warning for delinquencies.**

e. Defaults

Captures default events and recovery performance.

- **Foreign Key: loan_id (→ Loans)**
- **Attributes: default_date, dpd_days (days past due), recovery_%, collection_stage**

- **Purpose: Used for modeling recovery processes, risk analytics, and regulatory reporting on loan defaults.**

---

2. Relationships Between Entities

- **Institutions ↔ Loans: A *one-to-many* relationship exists, where one institution can be associated with many loans through the institution_id.**
- **Customers ↔ Loans: A *one-to-many* relationship, where a single customer may apply for one or more loans.**
- **Loans ↔ Payments: A *one-to-many* relationship, where each loan can have multiple associated EMI payment records.**
- **Loans ↔ Defaults: A *one-to-one (optional)* relationship, indicating that not all loans will default, but each default record must be linked to a specific loan.**

---

3. Integrity Constraints

- **Primary Keys (PK): Ensure uniqueness within each table.**
- **Foreign Keys (FK): Maintain referential integrity by linking related records across tables.**
- **Cascading Policies: Carefully handled (e.g., CASCADE DELETE is avoided to preserve audit history).**

4.3 Referential Integrity Constraints

- **CASCADE DELETE** – Not recommended (preserves audit trails)
- **CASCADE UPDATE** – Limited for privacy law changes
- **FOREIGN KEY CHECKS** – Enforced across all relationships
- **NOT NULL ENFORCEMENT** – Required for all keys
- **NULL HANDLING** – Foreign keys are NULL only when specified

---

5. Suggested Usage Scenarios

5.1 Credit Risk Analytics

- **Credit Risk Modeling** – Build ML models to predict loan defaults
- **Risk Scoring Assessment** – Develop customer risk assessment algorithms
- **Portfolio Risk Assessment** – Analyze institution-wise and geography-wise risk
- **Early Warning Systems** – Identify anomalies through missed or delayed EMI signals

5.2 Business Intelligence & Reporting

- **Loan Performance Dashboards** – Track disbursement, collections, and defaults
- **Institution Partnership Analysis** – Evaluate risk by institution
- **Customer Segment Behavior** – Group customers by demographic and behavior
- **Channel Trend Analysis** – Compare traditional vs. digital onboarding

5.3 Operational Analytics

- **EMI Collection Optimization** – Improve payment collection strategies
- **Delinquency Monitoring** – Identify customers falling behind on payments
- **Customer Segmentation** – Analyze customer interaction patterns
- **Fraud Detection** – Identify suspicious application or payment patterns

5.4 Machine Learning & AI Applications

- **Automated Underwriting** – ML-powered application decision engines
- **Dynamic Pricing Models** – Optimize interest rates based on risk
- **Collection Strategy Optimization** – Improve recovery rates through ML
- **Customer Lifetime Value** – Predict long-term customer profitability

5.5 Training & Education

- **SQL Query Practice** – Complex joins, aggregations, and window functions
- **Data Visualization Training** – Create meaningful BI charts and reports
- **Statistical Analysis** – Practice correlation, regression, and hypothesis testing
- **Business Case Studies** – Real-world scenarios for analytical decision making

6. Technology Stack

6.1 Data Generation

- **Apache Spark** – Distributed data generation framework
- **Pyspark** – Python API for Spark
- **Faker** – Synthetic data creation library
- **dbldatagen** – Table-based data generation
- **Pandas** – In-memory manipulation
- **Numpy** – Numerical and array support

6.2 Data Storage & Access

- **CSV Format** – For SQL import portability
- **Parquet Support** – For efficient columnar storage
- **Google Drive + GitHub** – Dataset download and sync
- **Spark DataFrames** – In-memory processing format

6.3 Analytics & Visualization

- **Python** – Primary scripting language
- **Jupyter Notebooks** – IDE for training workflows
- **Power BI** – Dashboard creation tool
- **Tableau** – Visual data exploration
- **Jupyter Notebook** – Interactive analysis environment