# Project Title: Data Exploration with Azure SQL Database – Customer, Account, and Loan Feeds

## Objective:

Trainees will explore and manipulate multiple related datasets using Azure SQL Database. The focus will be on organizing the datasets, identifying data types, and exploring their relationships and contents.

## Tools Required:

- Azure SQL Database

- SQL Management tools (Azure Data Studio or SQL Server Management Studio)

- Dataset - <https://kaggle.com/datasets/9234c6c4d25b6eb7c3dbb15a0e33d65ae68a405d42acba8db1248defee7aff9c>

- GitHub

## Project Tasks:

### 1. Setting Up Azure SQL Database

- Step 1.1: Create an Azure SQL Database in the Azure portal.

- Define a new database and server.

- Step 1.2: Name the database `CustomerAccountLoanDB`.

### 2. Data Organization

- Step 2.1: Create tables for the provided feeds:

- Customer Feed:

| CREATE TABLE customers (  customer\_id INT PRIMARY KEY,  first\_name VARCHAR(50),  last\_name VARCHAR(50),  address VARCHAR(100),  city VARCHAR(50),  state VARCHAR(50),  zip VARCHAR(20)  ); |
| --- |

- Account Feed:

| CREATE TABLE accounts (  account\_id INT PRIMARY KEY,  customer\_id INT,  account\_type VARCHAR(50),  balance DECIMAL(10, 2),  FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)  ); |
| --- |

- Transaction Feed:

| CREATE TABLE transactions (  transaction\_id INT PRIMARY KEY,  account\_id INT,  transaction\_date DATE,  transaction\_amount DECIMAL(10, 2),  transaction\_type VARCHAR(50),  FOREIGN KEY (account\_id) REFERENCES accounts(account\_id)  ); |
| --- |

- Loan Feed:

| CREATE TABLE loans (  loan\_id INT PRIMARY KEY,  customer\_id INT,  loan\_amount DECIMAL(10, 2),  interest\_rate DECIMAL(5, 2),  loan\_term INT,  FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)  ); |
| --- |

- Loan Payment Feed:

| CREATE TABLE loan\_payments (  payment\_id INT PRIMARY KEY,  loan\_id INT,  payment\_date DATE,  payment\_amount DECIMAL(10, 2),  FOREIGN KEY (loan\_id) REFERENCES loans(loan\_id)  ); |
| --- |

### 3. Data Insertion

- Step 3.1: Populate tables with sample data using `INSERT INTO` statements for each table.

- Step 3.2: Ensure data consistency and relationships, ensuring each foreign key points to valid primary keys.

Data available @ <https://kaggle.com/datasets/9234c6c4d25b6eb7c3dbb15a0e33d65ae68a405d42acba8db1248defee7aff9c>

### 4. Data Exploration

- Step 4.1: Write query to retrieve all customer information:

- Step 4.2: Query accounts for a specific customer:

- Step 4.3: Find the customer name and account balance for each account

- Step 4.4: Analyze customer loan balances:

- Step 4.5: List all customers who have made a transaction in the 2024-03

### 5. Aggregation and Insights

- Step 5.1: Calculate the total balance across all accounts for each customer:

- Step 5.2: Calculate the average loan amount for each loan term:

- Step 5.3: Find the total loan amount and interest across all loans:

- Step 5.4: Find the most frequent transaction type

- Step 5.5: Analyze transactions by account and transaction type:

### 6. Advanced Analysis

- Step 6.1: Create a view of active loans with payments greater than $1000:

- Step 6.2: Create an index on `transaction\_date` in the `transactions` table for performance optimization:

## Deliverables:

- A SQL script with the table creation and queries for data exploration.

- Screenshots of the queries and their results.

- Upload the SQL file (query.sql) and the document to GitHub.