## Question 1

You are designing a simple banking system that stores bank account information in an SQLite database. Implement a class ‘BankAccount’ representing a bank account with the following attributes and methods:

Attributes:

* ‘account\_number’: Account number (integer)
* ‘balance’: Current balance (float)

Create Methods:

* To initializes the account with the given account number, initial balance first and last name of the account holder. If the account number does not exist in the database, it creates a new entry with the provided initial balance.
* To deposit the specified amount into the account and updates the balance in the database.
* To withdraws the specified amount from the account if sufficient balance is available, and updates the balance in the database. If the balance is insufficient, it raises a custom exception ‘InsufficientFundsError’ and it must deduct a fee as penalty, even if the account becomes negative.
* To return the current balance of the account.
* To transfer the specified amount from the current account to the destination account, updating balances in both accounts. Handle cases where the source account has insufficient balance.

**NOTE**

* The database needs to keep a record of all transactions, including the type of transaction, deposit, withdraw or transfer.
* Ensure proper error handling and exception raising in case of database errors or insufficient funds during withdrawals.

**Tasks**

1. Create two instances of class Bank account, account1 and account2.
2. Transfer all the money from account1 to account2 (account1 should now have 0 balance)
3. Withdraw 200 from account 1
4. Store all the transactions in a dataframe and display

**Solution**

**Account details:**

Name

Last Name

Initial balance

Account number

**Transactions:**

Date

Transaction id

Account id

Type

Amount

Insufficient funds

**Transactions type:**

TID

Name

**Application**

**Check for insufficient funds**