Kiddy Bank

**Objective:**

“Kiddy Bank” is an application through which you can automate certain processes in Kiddy Bank, for example, EMI calculation, member’s balance sheet, group’s balance sheet, etc. The application is divided into 3 major modules (1. Administration, 2. Deposit Management, 3. Loan Management) based on their functionality.

**System Technical Overview**

* Memberid + groupId will be used for identifying the loans or deposits of any member.
* createdOn & updatedOn fields will be added in all tables for storing metadata info.
* History table will be used to store any change in any table.
* Monthly balance sheet table will be used to store each month balance sheet of the member against the group. This will be used for quick overview of any account. The detailed balance sheet will be calculated based on DEPOSIT table always.

1. **Administration**
2. ***GROUP*** is used to store group’s metadata information.

|  |  |  |
| --- | --- | --- |
| ***GROUP*** | | |
| Id | Varchar(10) | Unique identifier of the group. |
| Name | Varchar(20) | Name of the group. |
| roi | Int | Rate of interest on the loans of any group member. |
| deposit | Int | Fixed deposit amount of the member per month. |
| organizerid | Varchar(10) | Unique identifier of organizer of the group |

1. ***MEMBER*** is used to store group’s metadata information.

|  |  |  |
| --- | --- | --- |
| ***MEMBER*** | | |
| Id | Varchar(10) | Unique identifier of the group which is PhoneNumber of the member. This will also be used for member login. |
| Name | Varchar(30) | Name of the member. |
| address | Varchar(50) | Residence address of the member. |
| Email | Varchar(30) | Email Address for sending the reports and reminders. |
| password | Varchar(50) | Password of the member login which is saved in the form of Hashcode. |
| Is\_organizer | boolean | If yes, the member can be an organizer of one/more group. |

1. ***M2M\_GRP\_MEM*** is used to maintain the relation between the members and the groups. The relation between member & group is many-to-many(M2M).

|  |  |
| --- | --- |
| ***M2M\_GRP\_MEM*** | |
| grpId | Varchar(10) |
| grpStatus | Varchar(10) |
| memId | Varchar(10) |
| memStatus | Varchar(10) |

1. ***HISTORY*** is used to track any change in any table.

|  |  |  |
| --- | --- | --- |
| ***HISTORY*** | | |
| id | Varchar(10) | Unique identifier of the history record. |
| entityId | Varchar(10) | Unique identifier of the entity. |
| entityName | Varchar(10) | Name of the entity(table name). |
| ­­oldValue | Varchar(100) | List of original values. Stringified value. |
| newValue | String[] | List of updated values. Stringified value. |

1. **Deposit Management**
2. ***DEPOSIT*** is used to store all the deposits done by the user.

|  |  |  |
| --- | --- | --- |
| **DEPOSIT** | | |
| Id (PK) | Varchar(10) | Unique identifier of the deposit. |
| MemberId | Varchar(10) | Unique identifier of the member who deposited. |
| GroupId | Varchar(10) | Unique identifier of the group in which member is part of. |
| Amount | int | Deposited amount. |
| Date | date | Deposited date. |
| Deposit\_month | Varchar(10) | Deposited month along with year. It will be useful for report generation. |
| Bank\_Tx\_id | Varchar(30) | Bank transaction id for the reference. |
| Loan\_id | Varchar(10) | Unique identifier of the loan against which the deposit has been done. It will be empty in case of regular monthly deposit. |
| Int\_amount | int | Interest amount against the principal/outstanding amount of the loan. It will be empty if the loan\_id is empty. |

1. ***BALANCE\_SHEET*** is used to store every month balance sheet for simplification and calculation.

|  |  |  |
| --- | --- | --- |
| ***BALANCE\_SHEET*** | | |
| Month | Varchar(10) | Month for which the balance sheet is prepared. |
| MemId | Varchar(10) | Unique identifier of the member who deposited. |
| GrpId | Varchar(10) | Unique identifier of the group in which member is part of. |
| total\_amount | int | Total amount till date by the member. |
| Balance\_amount | int | Total Balance amount against all the loans by the member. |

1. **Loan Management**
2. ***LOAN*** is used to store all the loans opted by the members.

|  |  |  |
| --- | --- | --- |
| ***LOAN*** | | |
| Id | Varchar(10) | Unique identifier of the loan. |
| MemId | Varchar(10) | Unique identifier of the member who opted for the loan. |
| GrpId | Varchar(10) | Unique identifier of the group in which member is part of. |
| Amount | int | Loan amount . |
| Issued\_date | date | Loan issued date. |
| Closing\_date | date | Closing date of the loan. |
| No\_of\_EMI | int | No. of EMIs in which the loan is going to be closed. |
| Bank\_transaction\_id | Varchar(30) | Bank transaction id for the reference. |
| Status | Varchar(10) | The status of the loan. OPEN, CLOSED. |
| parent\_loan\_id | Varchar(10) | Unique identifier of the parent loan. It will be empty if there is no balance transfer of current loan. |
| parent\_loan\_bal\_transfer | int | Amount transferred from parent loan. Outstanding amount which of parent loan which is transferred to current loan. |

**Useful Queries:**

1. CREATE TABLE table\_group (id varchar(10) NOT NULL, name varchar(30) NOT NULL, organizerid varchar(10) NOT NULL, deposit int, roi int, PRIMARY KEY (id) );
2. select \* from table\_group;
3. CREATE TABLE table\_member (id varchar(10) NOT NULL, name varchar(30) NOT NULL, address varchar(50), email varchar(30), password varchar(30), PRIMARY KEY (id) );
4. select \* from table\_member;
5. select \* from table\_m2m\_grp\_mem;
6. drop table table\_m2m\_grp\_mem;
7. CREATE TABLE table\_m2m\_grp\_mem (groupid varchar(255) NOT NULL, groupstatus varchar(255), memberid varchar(10) NOT NULL, memberstatus varchar(10), PRIMARY KEY (groupid, memberid) );
8. CREATE TABLE table\_deposit (id varchar(10) NOT NULL, memberid varchar(10) NOT NULL, groupid varchar(10), amount int NOT NULL, depositmonth varchar(10), banktxid varchar(30), loanid varchar(10), intamount int, PRIMARY KEY (id) );
9. drop table table\_deposit;
10. select \* from table\_deposit;

**To run the jar:** *java -jar target\piggy-bank-0.0.1-SNAPSHOT.jar*

