DBMS Banking Project

Scope of Project:

Our team has designed a banking database management system which caters to many functions like keeping track of bank branches, accounts maintained by that branch, the customers it serves, the workers it employs etc. The banking database also keeps track of loans given to customers (a customer may have an account, may have taken a loan, or both), and the payments in installments made by the customers to repay the loan. Also, each bank branch has an employee as a manager, and maintains the details of the same. This is widely the scope in which this database operates.

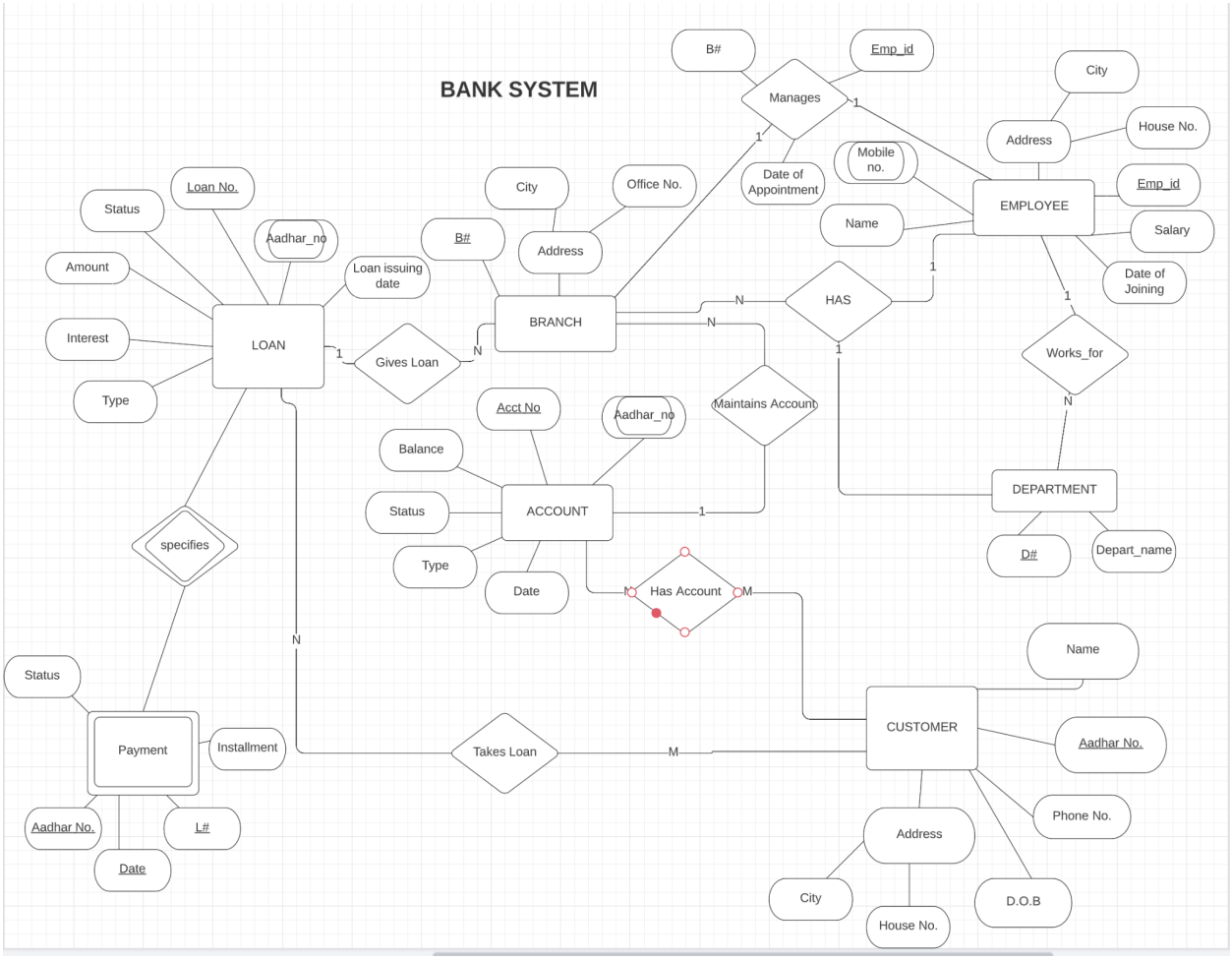
Stakeholders:

Some stakeholders of this project are:

1) Client (Customer) who can deposit money in various types of accounts, or can take loans for some purpose.
2) Banker (Employee) who works in the bank branch, and for a particular department.
3) Manager: Effectively the head of the branch, who is responsible for the various activities of the bank. He/she is also an employee, and will also belong to a department, but will be responsible for the overall functioning of that bank branch.

Do note that in a full real world scenario, this project would have lots of stakeholders involved, such as financial markets, governments, board of directors, etc. However, these entities all fall beyond the scope of this project, and have not been included in the above list.

ER DIAGRAM:



[DBMS_E-R_diagram: Lucidchart]

Relationships:

1) Has: A bank branch has Departments and Employees
2) Manages: Bank Branch has Manages relationship with employee, as 1 of the employees is the manager of the Branch.
3) Maintains Account: The Bank Branch maintains the account of customers.
4) Works For: An Employee has this relationship with the Department entity.

5) Gives Loan: The bank branch has this relationship with the Loan Entity.
6) Takes Loan: The Customer entity has this relationship with the Loan Entity.
7) Has Account: The Customer entity has this relationship with the Account Entity.
8) Specifies: The Payment is a weak entity, and its attributes are specified by the Loan Entity. The repayment is based on the Loan, if there is no loan, then no payment for that purpose is made.

Weak Entity: In this project, the Payment is a weak entity. Its existence is purely based on the existence of the Loan Entity. If no loan is taken, then there will be no payment to be made. Many attributes of the Payment entity are based on Loan Entity, like Loan Number, and Aadhar Number, and their values are set according to the values in the Loan Entity, so this is completely dependent on the Loan Entity.

Ternary Relationship: In this ER diagram, the Has relationship is a ternary relationship, as it links the entities Bank Branch, Employee and Department. There are multiple employees in a bank branch, who work in the various departments of the branch. So these entities are linked together by the Has relationship, and hence, this forms a ternary relationship.

Relational Schema

1) BRANCH(B# (UNIQUE,NOT NULL), Office No.(NOT NULL), City(NOT NULL))
2) DEPARTMENT(D# (UNIQUE,NOT NULL), Dept_Name(NOT NULL), B#)
3) CUSTOMER(Aadhar No.(UNIQUE,NOT NULL), Phone No.(NOT NULL), Date of Birth, Name(NOT NULL), House No.(NOT NULL), City(NOT NULL))
4) ACCOUNT(Account No (UNIQUE,NOT NULL), Balance(NOT NULL), Type(NOT NULL), Status(NOT NULL), {Aadhar No.},B#)
5) LOAN(Loan No. (UNIQUE, NOT NULL), Amount(NOT NULL), Interest(NOT NULL), Type(NOT NULL), {Aadhar No.}, B#, Status(NOT NULL))
6) EMPLOYEE(Emp_ID (UNIQUE,NOT NULL), Salary(NOT NULL), Date of Joining(NOT NULL), Name(NOT NULL), {Mobile No.}, City(NOT NULL), House No., B#, D#)
7) PAYMENTS(Aadhar_no (UNIQUE,NOT NULL),Loan_no (UNIQUE,NOT NULL),date(NOT NULL),Installment(NOT NULL),Status(NOT NULL))
8) MANAGES(B# (UNIQUE,NOT NULL),Emp_id,Date_of_appointment(NOT NULL))

DDL and Integrity Constraints:

1) Bank Branch: Primary Key - BranchNumber (not null, unique). For further entities, these not null, unique will be considered inherently in primary key.
2) Customer: Primary key - AadharNumber

3) Department: Primary Key - DepartmentNumber, Foreign Key - BranchNumber (from Bank Branch Table)
4) Employee: Primary Key - EmployeeID, Foreign Keys - BranchNumber, DepartmentNumber
5) Loan: Primary Key - LoanNumber, Foreign Keys - BranchNumber, AadharNumber
6) Account: Primary Key - AccountNumber, Foreign Keys - BranchNumber, AadharNumber
7) Payments: Primary Key - (AadharNumber, LoanNumber, Date), Foreign Keys - LoanNumber, AadharNumber
8) Manages: Primary Key - EmployeeID, Foreign Keys - BranchNumber, EmployeeID
   Note that in Manages table, both EmployeeID and BranchNumber can be taken as primary keys (means that they are both candidate keys), but we have taken EmployeeID as the primary key.

SQL Queries:

1. **List the average salary of all the departments.**
   Ans:
   SELECT DepartmentNumber,
   avg(Salary) OVER (PARTITION BY DepartmentNumber) AS AVG_Salary
   FROM EMPLOYEE;

DepartmentNum...	average_salary
1	20452783
2	20263639
3	11222417
4	14894666
5	19328745
6	21844680
7	15844903
8	17873782
9	13089546
10	22067205
11	11852046
12	13317945
13	12150287
14	10983118
15	10336834
16	14721631
17	22427522
18	15114190
19	18131907
20	16993549
21	13208167
22	15252807
23	18760981
24	24157408
25	18772124
26	22478405
27	10651999
28	21957195
29	20187228
30	21717017

2. **Write an SQL query to list the employees that are also managers**.
   Ans:
   SELECT E.Employeename from Employee as E,Manages as M
   WHERE(E.EmployeeID=M.EmployeeID) ;

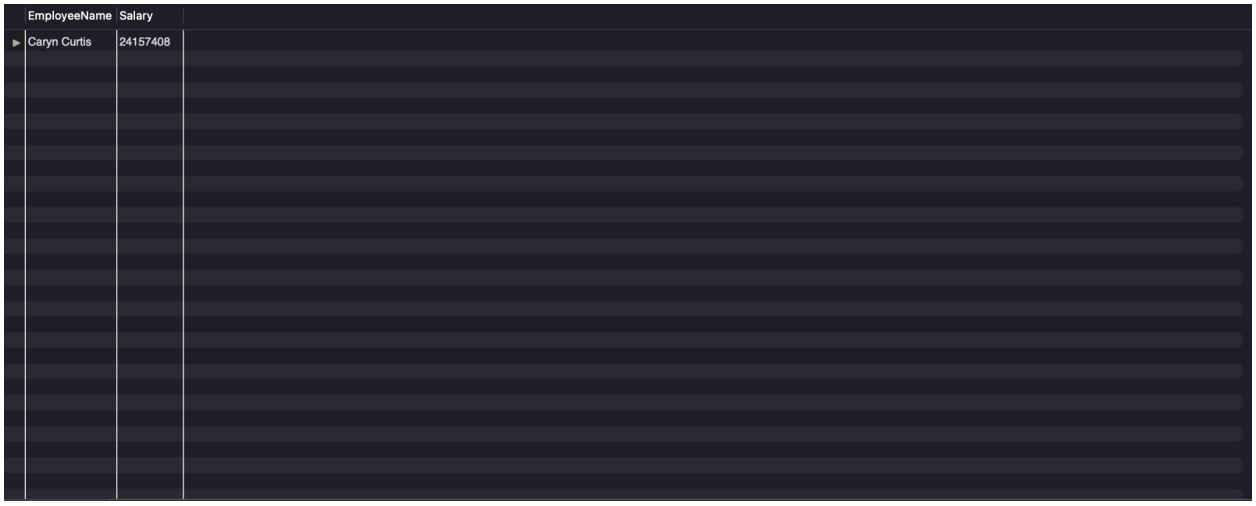
Employeename
Bibby Barwood
Baird Harlett
Oberon Jura
Patience Reavey
Noelyn Housego
Torry Donner
Noelle Little
Gracia Sharville
Demott Runnacles
Maritsa Nichols
Foss Volage
Beauregard Lun…
Myrtle Coughlin
Hardy Goulborne
Blake Cochrane
Averyl Gulliver
Cathee Frany
Dyna Overshott
Allister Kilcoyne
Kylynn Kasting

3. **Return details of the employee with the nth highest salary.**

Ans:

SELECT EmployeeName,Salary
From (SELECT EmployeeName,Salary,DENSE_RANK() over (ORDER by Salary desc) as rk
FROM EMPLOYEE) as a where rk=n

For N=5



EmployeeName	Salary
Caryn Curtis	24157408

4. **List all Active Loans by Price and Interest**

Ans.

SELECT C.Name,L.AadharNumber,L.Amount,L.Interest,L.Status,L.Type FROM LOAN AS L,CUSTOMER AS C WHERE C.AadharNumber=L.AadharNumber AND L.Status!="Fully Paid" AND L.Status!="Grace" ORDER BY Amount,Interest DESC;

Name	AadharNumber	Amount	Interest	Status	Type
Tina Weathey	6699-7334-2929	124199	10	Repayment Ongoing	Gold
Cole Chippindale	5589-9815-6742	379515	10	Repayment Ongoing	Personal
Cookie Floris	9810-3591-7194	717217	10	Repayment Ongoing	Vehicle
Catherine Riddel	5295-1205-7954	1147686	9	Repayment Ongoing	Home
Margalo Elldred	1983-2713-2408	1306185	9	Repayment Ongoing	Business
Margalo Elldred	1983-2713-2408	2246196	9	Repayment Ongoing	Vehicle
Wilbur Gritland	3827-9175-7019	2532743	8	Repayment Ongoing	Educational
Carrissa Ibeson	2916-6143-1155	2597927	9	Repayment Ongoing	Business
Carrissa Ibeson	2916-6143-1155	3125683	9	Repayment Ongoing	Home
Wilbur Gritland	3827-9175-7019	3176897	9	Repayment Ongoing	Vehicle
Davita Pladen	1361-2909-1956	3617155	10	Repayment Ongoing	Gold
Albie Cancott	3115-6511-4003	3796742	7	Repayment Ongoing	Personal
Farley Joddens	0394-0836-0938	4216250	7	Repayment Ongoing	Vehicle
Ezekiel Troillet	2267-6622-3253	4656968	9	Repayment Ongoing	Vehicle
Jock Matityahu	4597-3766-7385	7037634	8	Repayment Ongoing	Gold
Frank Heathfield	2788-4810-0143	7210879	8	Repayment Ongoing	Home
Julie Follett	4328-4494-4111	8723854	7	Repayment Ongoing	Business
Nickolai California	7548-6971-4573	8740464	9	Repayment Ongoing	Home
Kate Witsey	5174-0230-5988	8953327	9	Repayment Ongoing	Home
Stormi Temprell	6321-0225-5209	9582839	7	Repayment Ongoing	Home
Cookie Floris	9810-3591-7194	9942473	8	Repayment Ongoing	Personal
Karrah Glencorse	0060-9771-5565	105782...	6	Repayment Ongoing	Vehicle
Albie Cancott	3115-6511-4003	141949...	8	Repayment Ongoing	Home
Jock Matityahu	4597-3766-7385	144254...	6	Repayment Ongoing	Educational
Latrina Cluff	8482-6235-2555	184608...	7	Repayment Ongoing	Home
Wilbur Gritland	3827-9175-7019	203232...	8	Repayment Ongoing	Home
Juliette Butrimo...	7956-7811-0133	332664...	8	Repayment Ongoing	Vehicle
Latrina Cluff	8482-6235-2555	378743...	6	Repayment Ongoing	Educational
Wilbur Gritland	3827-9175-7019	409895...	9	Repayment Ongoing	Personal

5. **List Account Numbers,Aadhar Number and Loan amount of people who have taken a loan of amount 1cr and more and have 10cr or less in their accounts.**
   Ans:
   SELECT A.AadharNumber,A.AccountNumber,L.amount from ACCOUNT as A,LOAN as L where (A.AadharNumber=L.AadharNumber AND A.Balance<=100000000 AND L.amount>=10000000)

AadharNumber	AccountNumber	amount
8482-6235-2555	Ba-5993-ei3317	34972423
3115-6511-4003	Ea-6651-gl7276	14194948
7956-7811-0133	Mq-2090-ri7119	33266483
8482-6235-2555	Ba-5993-ei3317	18460816
3115-6511-4003	Ea-6651-gl7276	20353482
5295-1205-7954	Bh-7570-qj2286	18520576
4328-4494-4111	Ru-7251-vg8585	67631717
8482-6235-2555	Ba-5993-ei3317	37874378
7956-7811-0133	Mq-2090-ri7119	25669366

6.Create an Account view for all active accounts in bank

Ans:

CREATE VIEW all_accounts AS (SELECT * FROM ACCOUNT WHERE Status!="Closed");

AccountNumber	Balance	Type	Status	AadharNumber	BranchNumber
Af-0725-fv4207	135409526	Savings	Dormant	6680-4049-0096	19
Ah-3023-xf7446	315348377	Fixed Deposits	Running	9787-2919-4979	14
Ak-9564-jc3576	403998978	Fixed Deposits	Running	0060-9771-5565	12
Aq-0206-la3821	56045093	Checking	Running	0171-3393-2969	3
Aq-4574-ma1114	319965913	Checking	Active	9810-3591-7194	11
Aq-9654-rx9911	430062597	Checking	Dormant	1270-6875-8862	11
Ax-9642-kr1162	438561047	Fixed Deposits	Dormant	3458-5062-8839	3
Ba-5993-ei3317	51479945	Fixed Deposits	Dormant	8482-6235-2555	20
Bb-6276-po9394	173726548	Checking	Active	9232-1888-9502	11
Bb-6644-mv3052	420729423	Fixed Deposits	Active	7004-5925-5651	12
Be-5116-ns9671	269031359	Savings	Active	9869-8649-1676	7
Bh-3757-zo3874	869770557	Fixed Deposits	Dormant	9583-3760-4918	15
Bh-7570-qj2286	77408023	Savings	Dormant	5295-1205-7954	2
Bm-3692-wb5392	39657350	Savings	Active	7775-2948-3773	5
Br-0894-lf8406	7416969	Savings	Running	2822-9145-5943	15
Bt-6951-xj8461	122577810	Savings	Dormant	7553-9957-0620	9
By-8662-jz4680	876389217	Savings	Active	4421-6471-6019	19
Cn-2412-pl9853	460036781	Checking	Active	5513-3225-8629	5
Cn-8723-ex1963	64557456	Fixed Deposits	Dormant	7185-3698-0989	10
Cp-2760-ni6455	325944279	Savings	Active	3803-1905-8888	18
Cq-2676-ux9054	116317051	Checking	Active	0522-1052-0575	4
Ct-0633-lk6938	995145709	Checking	Running	6224-6827-6983	9
Ct-9928-fu5691	43089161	Fixed Deposits	Running	5270-0225-7750	15
Cv-6822-sg9224	24284216	Checking	Dormant	8572-1847-8935	20
Cv-9543-zs1925	462944289	Fixed Deposits	Dormant	0005-5989-5302	3
Cx-3839-sq8613	341584913	Checking	Active	3945-4356-3304	20
Cz-6450-qf7060	89266885	Checking	Dormant	5394-4913-7678	17
Dc-4055-hv5191	77483521	Checking	Active	3727-5626-9729	4

7. List all persons having more than 1 account

SELECT C.Name, C.AadharNumber,Count(C.AadharNumber) FROM customer as C INNER JOIN Account as A ON C.AadharNumber = A.AadharNumber GROUP BY C.AadharNumber HAVING COUNT(*) > 1 ORDER BY Count(C.AadharNumber) DESC;

Name	AadharNumber	Count(C.AadharNumber)
Michel Innott	6270-1096-4714	5
Ginelle Halworth	9232-1888-9502	5
Albie Cancott	3115-6511-4003	5
Conn Omand	2822-9145-5943	5
Hilary Demange	2227-9911-9278	5
Antonin Cotherill	4892-5262-4965	4
Rey Gallen	5270-0225-7750	4
Dalila Hulk	5598-1054-3576	4
Murry McGhee	5631-8425-4226	4
Emera Cutmore	0522-1052-0575	4
Karee Gehrts	6224-6827-6983	4
Eugenio McLean	9787-2919-4979	4
Cleo Sainter	0798-5908-8363	4
Mercie Edden	3632-1550-3415	4
Davita Katt	9314-0616-2902	4
Juliette Butrim…	7956-7811-0133	3
Ashlie Loveless	2313-7088-9573	3
Chrissy Dessei…	7525-1086-3101	3
Billi Enrico	2633-7814-9297	3
Jania Swigg	0005-5989-5302	3
Virgina Seeley	7004-5925-5651	3
Claudell Dunhill	8104-4564-8357	3
Bartram Tampli…	2823-8252-3607	3
Letitia Venes	6476-8429-3979	3
Carrissa Ibeson	2916-6143-1155	3
Joshua Gibberd	8773-1461-1289	3
Elsie Gagg	9583-3760-4918	3
Averil Merredy	3458-5062-8839	3
Persis Boshers	1239-9825-8586	3
Oralia Swyer	3700-4525-6311	3
Cookie Floris	9810-3591-7194	3
Adel Tabourin	5817-2008-1394	3
Kalie Machan	0176-6933-7158	3
Herb Broadbent	0171-3393-2969	3
Andreas Petru…	5572-2381-1317	3

8.List names for persons by which they pay the installment earliest

Ans:

SELECT C.Name,C.AadharNumber,L.LoanNumber,L.Amount,P.Installment,P.DATE FROM CUSTOMER AS C,LOAN AS L, PAYMENTS AS P WHERE C.AadharNumber=P.AadharNumber AND P.LoanNumber=L.LoanNumber AND L.LoanNumber=P.LoanNumber ORDER BY P.DATE,P.Installment ASC;

Name	AadharNumber	LoanNumber	Amount	Installment	DATE
Caleb Cossom	3909-8456-5240	Q5306	5017985	37483	1979-03-02
Juliette Butrimovich	7956-7811-0133	Y2147	25669366	12356	1990-05-06
Caleb Cossom	3909-8456-5240	G6039	5154174	14246	1992-08-14
Lawton Bromwich	3340-5080-0260	K2177	2559508	42866	1993-07-29
Lawton Bromwich	3340-5080-0260	M9298	1414721	28483	1994-03-25
Tina Weathey	6699-7334-2929	Q8345	5378910	87365	1996-05-08
Catherine Riddel	5295-1205-7954	G9557	1147686	23883	1997-01-06
Lawton Bromwich	3340-5080-0260	C2873	6487209	46884	2000-03-05
Tina Weathey	6699-7334-2929	Z0611	124199	87347	2000-06-10
Juliette Butrimovich	7956-7811-0133	Y2147	25669366	73269	2000-07-09
Frank Heathfield	2788-4810-0143	L8081	4087806	17890	2001-09-07
Tina Weathey	6699-7334-2929	Z4324	8005610	13266	2001-11-17
Frank Heathfield	2788-4810-0143	J5681	2182919	33820	2002-05-06
Frank Heathfield	2788-4810-0143	Z8774	7210879	17472	2003-09-15
Lawton Bromwich	3340-5080-0260	Q5627	8832111	324732	2004-01-02
Lawton Bromwich	3340-5080-0260	Z3848	106220	3286	2005-05-06
Catherine Riddel	5295-1205-7954	T6393	18520576	42573	2007-10-11
Hermia Fozzard	4338-7553-8941	Z6310	2929439	32383	2012-12-21
Wilie Cesaric	4741-3278-8495	W4691	78307858	58932	2021-03-09

9.Close all accounts owned by 2788-4810-0143

UPDATE ACCOUNT SET Status='Closed' WHERE AadharNumber='2788-4810-0143';

10.List the branch ids with their most dominant loan types

SELECT BranchNumber,MAX(Type) FROM LOAN GROUP BY BranchNumber;

BranchNumber	MAX(Type)
1	Vehicle
2	Vehicle
3	Vehicle
4	Vehicle
5	Vehicle
6	Personal
7	Vehicle
8	Vehicle
9	Home
10	Home
11	Vehicle
12	Vehicle
13	Personal
14	Vehicle
15	Vehicle
16	Vehicle
17	Personal
18	Vehicle
19	Vehicle
20	Vehicle

Indexing:

We have identified 8 attributes to create Index tables required for our SQL queries.

1. alter table Department
   add index ref1(DepartmentNumber);
   // Have made the DepartmentNumber(Primary Key) as indexing reference in the Department Table.

2. alter table ACCOUNT
   add index ref2(AccountNumber);
   // Have made the AccountNumber(Primary Key) as indexing reference in the Account Table.

3. alter table BANKBRANCH

add index ref3(City);
// Have made the City as an indexing reference in the BANKBRANCH Table.

4 . alter table CUSTOMER
add index ref4(Name);
// Have made the Name as an indexing reference in the CUSTOMER Table.

5 . alter table Manages
add index ref5(EmployeeID);
// Have made the EmployeeID(Primary Key) as indexing reference in the Manages Table.

6 . alter table Payments
add index ref6(AadharNumber);
// Have made the AadharNumber(Primary Key) as indexing reference in the Payments Table.

7 . alter table LOAN
add index ref7(LoanNumber);
// Have made the LoanNumber(Primary Key) as indexing reference in the LOAN Table.

8 . alter table Employee
add index ref8(EmployeeName);
// Have made the EmployeeName as indexing reference in the Employee Table

```
983 •   alter table Department
984     add index ref1(DepartmentNumber);
985
986     alter table ACCOUNT
987     add index ref2(AccountNumber);
988
989     alter table BANKBRANCH
990     add index ref3(City);
991
992     alter table CUSTOMER
993     add index ref4(Name);
```

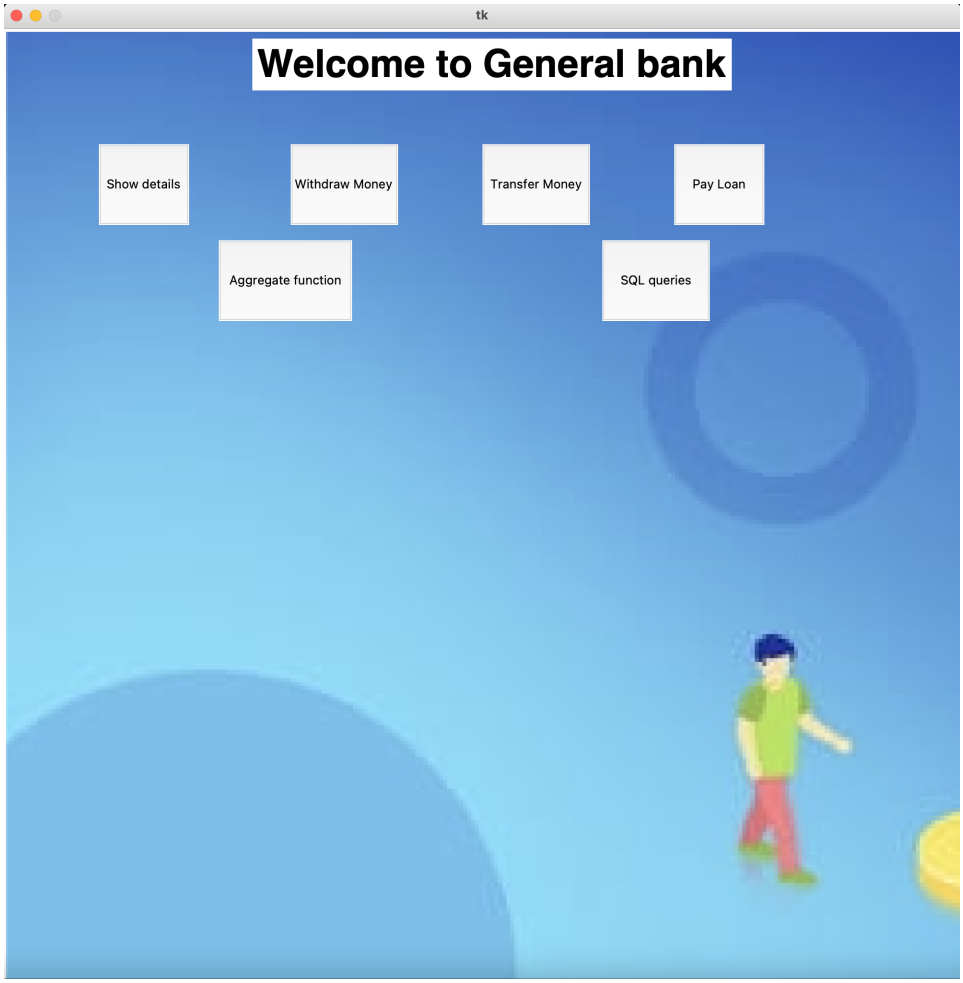
```
995     alter table Manages
996     add index ref5(EmployeeID);
997
998     alter table Payments
999     add index ref6(AadharNumber);
1000
1001    alter table LOAN
1002    add index ref7(LoanNumber);
1003
1004    alter table Employee
1005    add index ref8(EmployeeName);
```

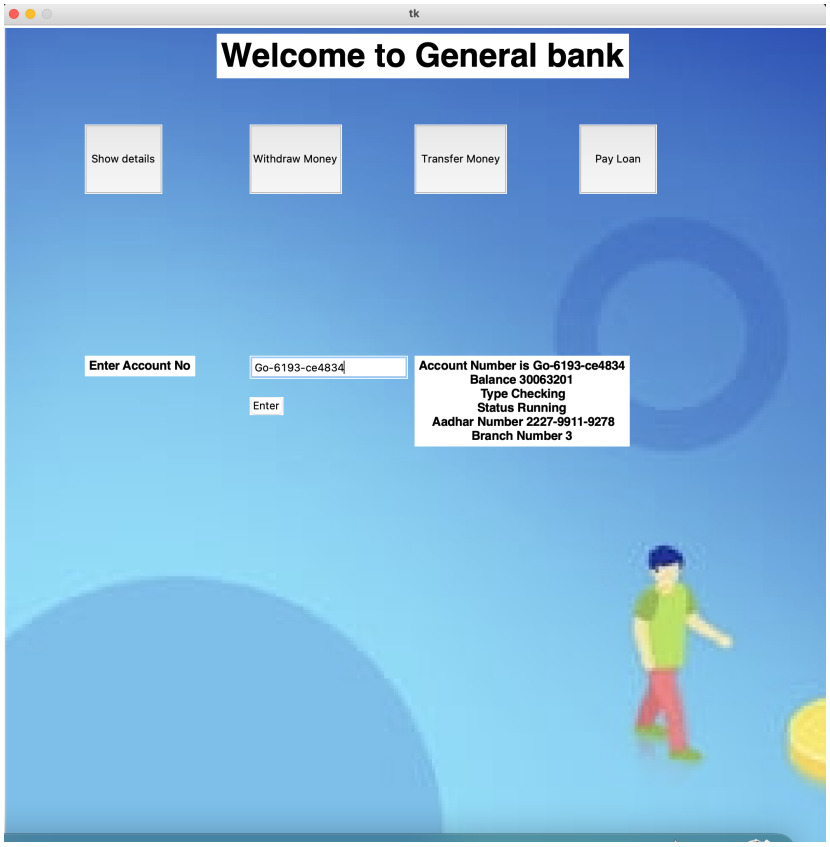
Embedded query:

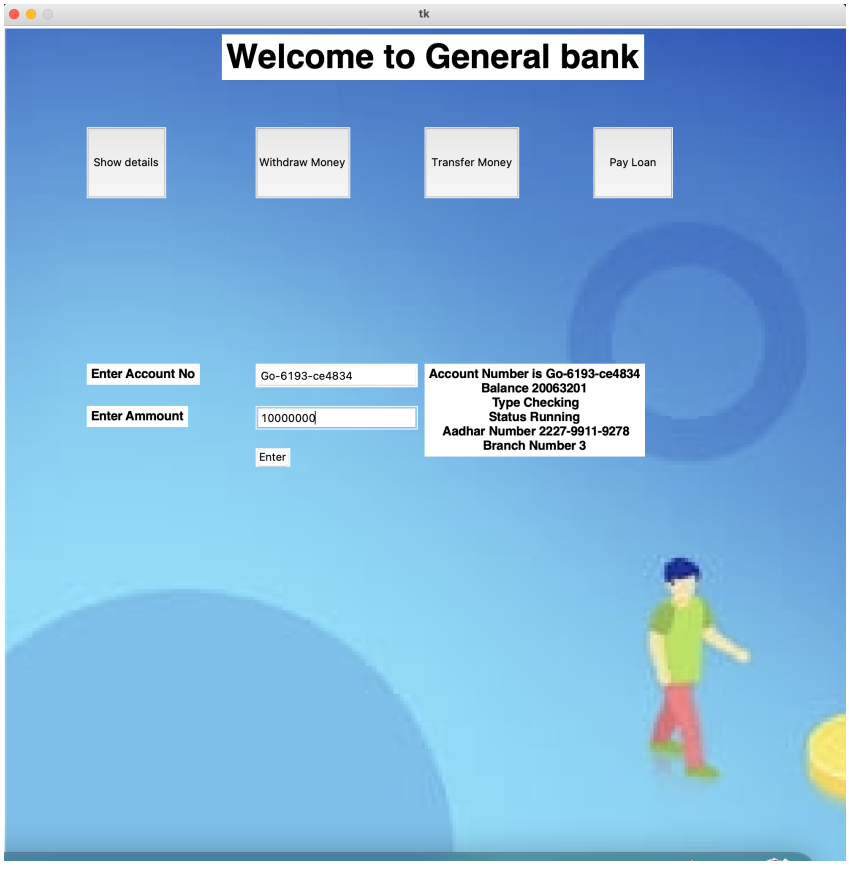
For the embedded query we embedded the sql queries in the python in which we included the basic bank functionalities like viewing basic account details,

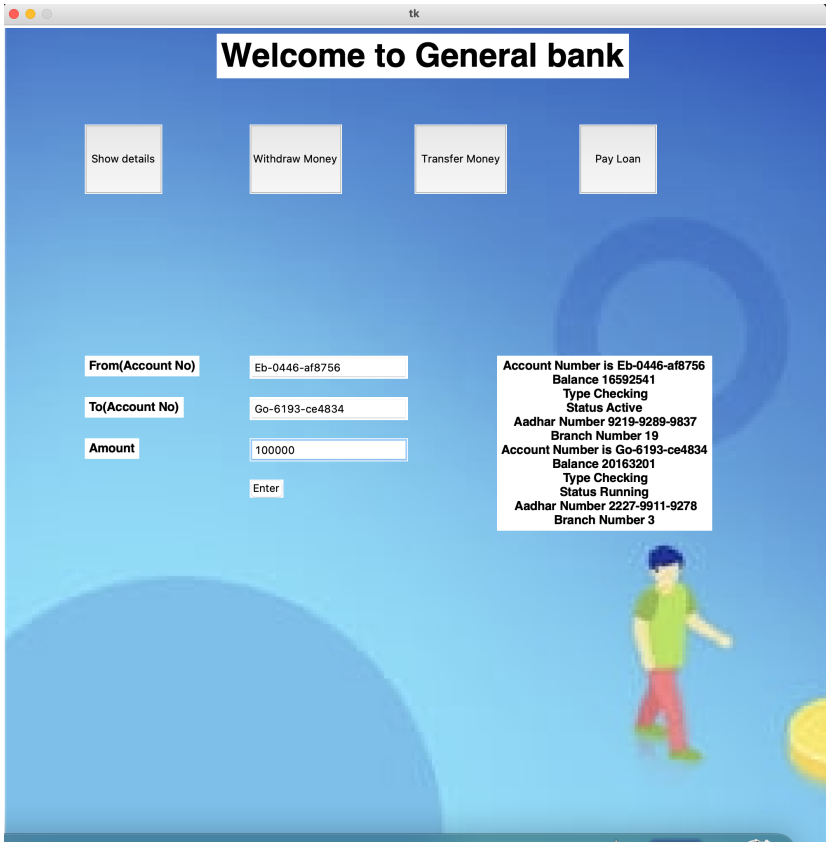
withdrawing money from bank and transferring money from one account to another account.

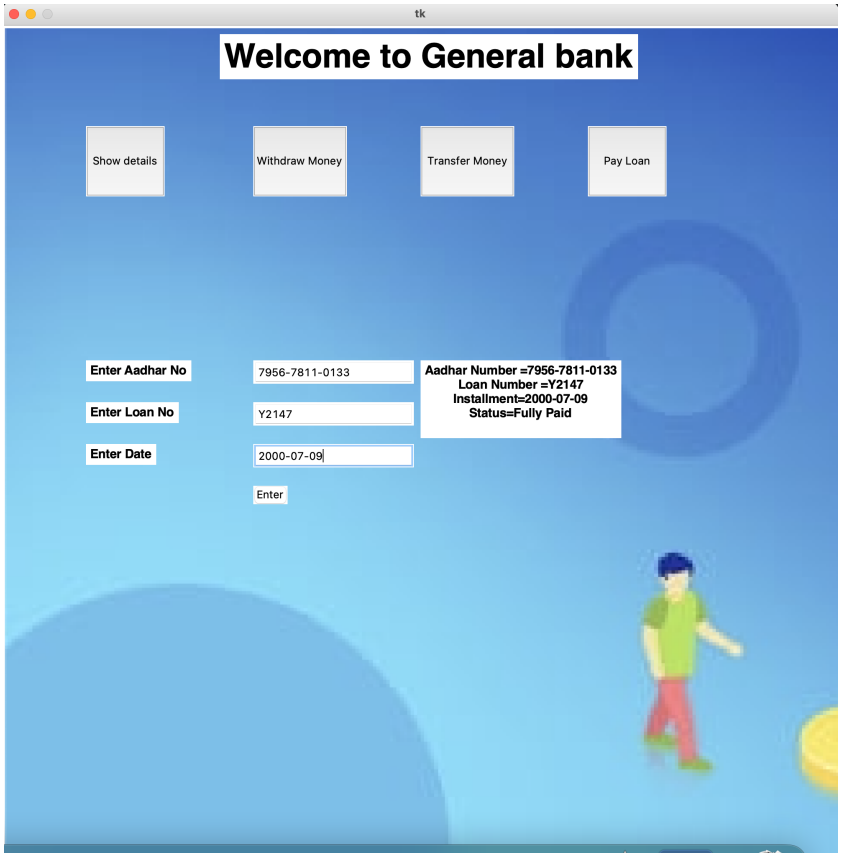
Screenshots:

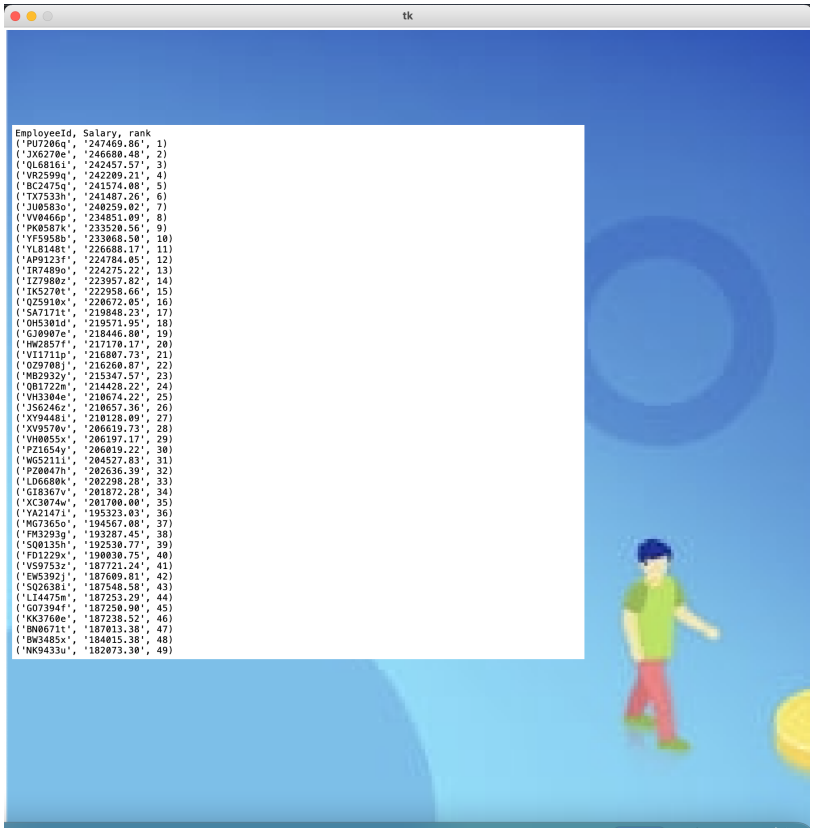


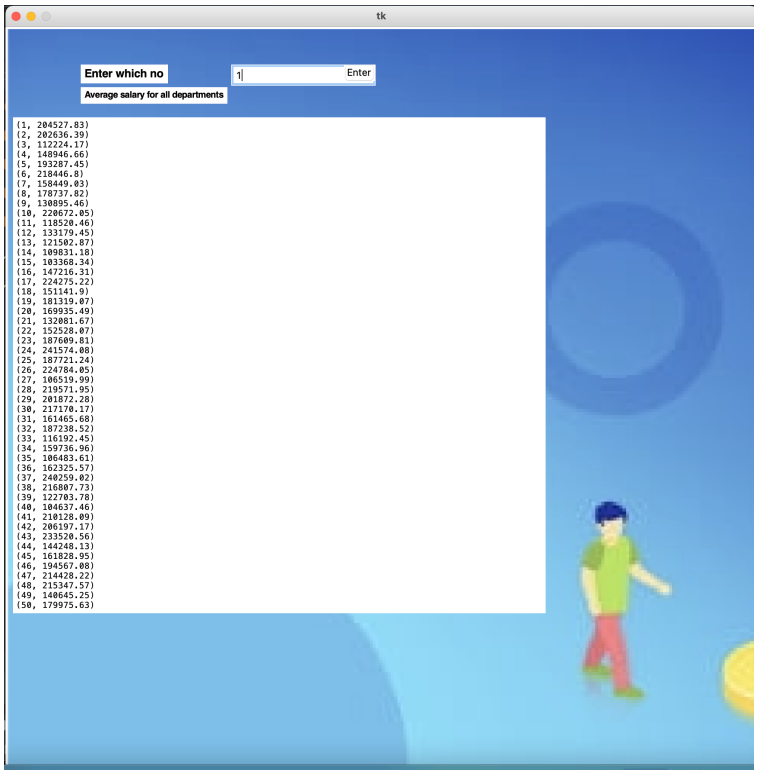

tk
Welcome to General bank
Show details
Withdraw Money
Transfer Money
Pay Loan
Enter Account No
Go-6193-ce4834
Enter
Account Number is Go-6193-ce4834
Balance 30063201
Type Checking
Status Running
Aadhar Number 2227-9911-9278
Branch Number 3

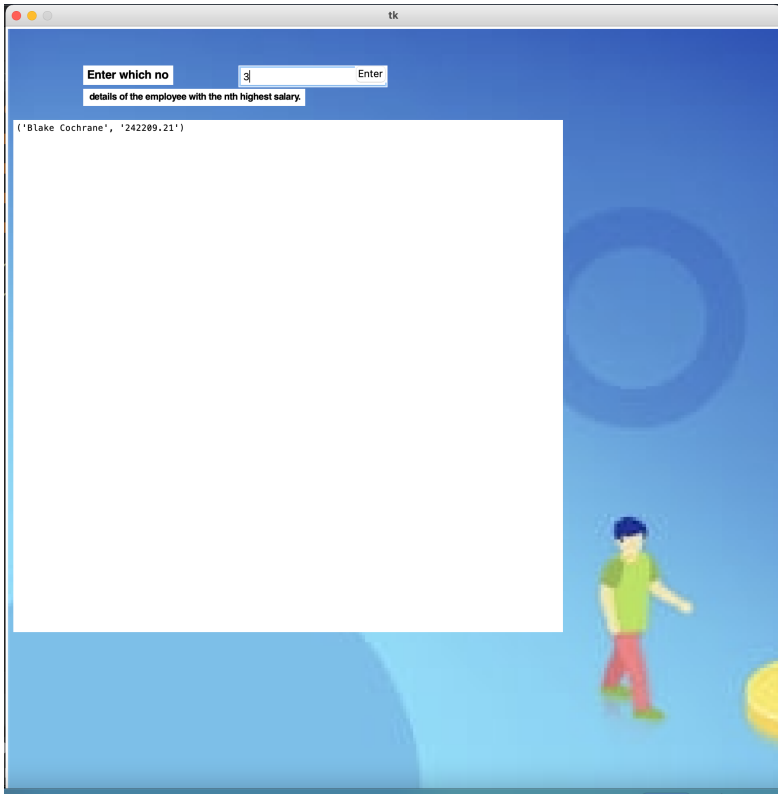

tk
Welcome to General bank
Show details
Withdraw Money
Transfer Money
Pay Loan
Enter Account No
Go-6193-ce4834
Enter Ammount
10000000
Enter
Account Number is Go-6193-ce4834
Balance 20063201
Type Checking
Status Running
Aadhar Number 2227-9911-9278
Branch Number 3

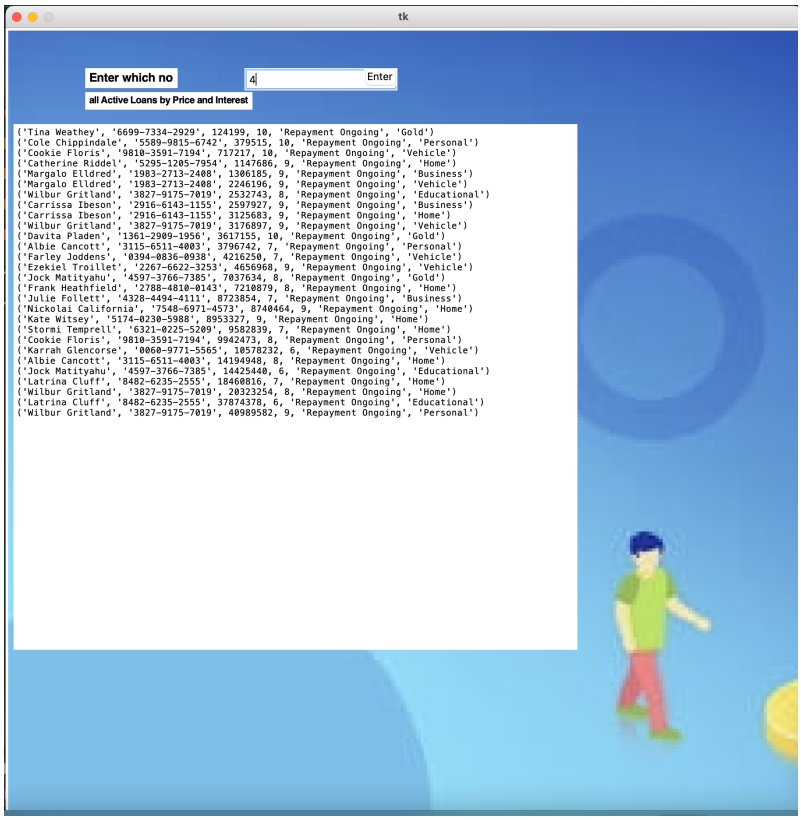

tk
Welcome to General bank
Show details
Withdraw Money
Transfer Money
Pay Loan
From(Account No)
Eb-0446-af8756
To(Account No)
Go-6193-ce4834
Amount
100000
Enter
Account Number is Eb-0446-af8756
Balance 16592541
Type Checking
Status Active
Aadhar Number 9219-9289-9837
Branch Number 19
Account Number is Go-6193-ce4834
Balance 20163201
Type Checking
Status Running
Aadhar Number 2227-9911-9278
Branch Number 3

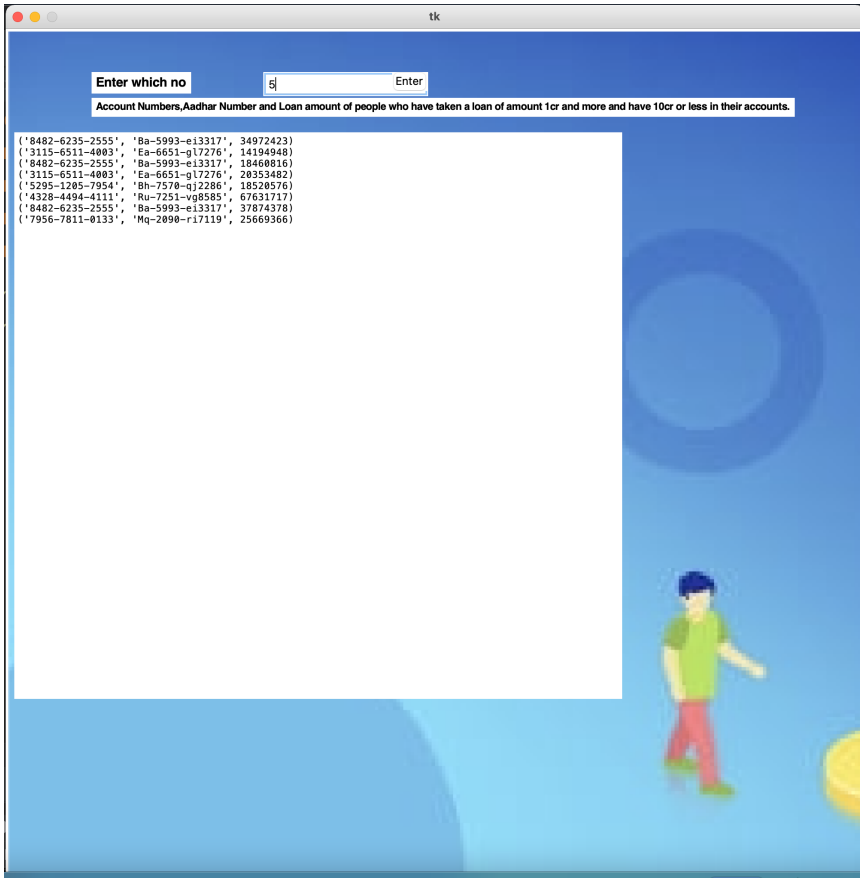

tk
Welcome to General bank
Show details
Withdraw Money
Transfer Money
Pay Loan
Enter Aadhar No
7956-7811-0133
Enter Loan No
Y2147
Enter Date
2000-07-09
Enter
Aadhar Number =7956-7811-0133
Loan Number =Y2147
Installment=2000-07-09
Status=Fully Paid

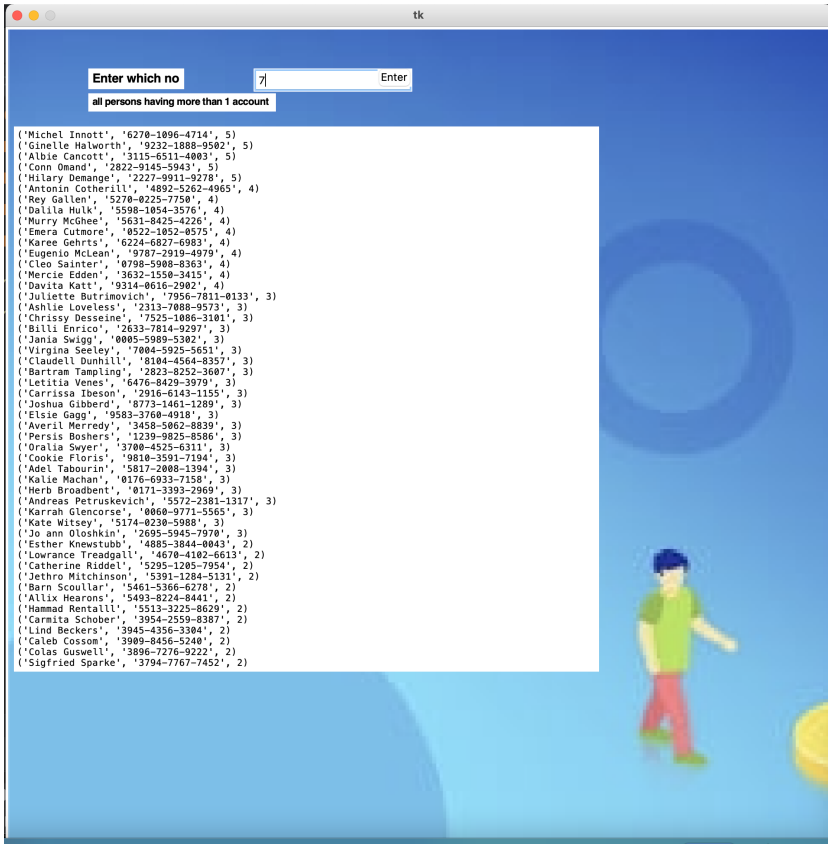

tk
EmployeeId, Salary, rank
('PU7206q', '247469.86', 1)
('JX6270e', '246680.48', 2)
('QL6816i', '242457.57', 3)
('VR2599q', '242209.21', 4)
('BC2475q', '241574.08', 5)
('TX7533h', '241487.26', 6)
('JU0583o', '240259.02', 7)
('VV0466p', '234851.09', 8)
('PK0587k', '233520.56', 9)
('YF595Bb', '233068.50', 10)
('YL8148t', '226688.17', 11)
('AP9123f', '224784.05', 12)
('IR7489o', '224275.22', 13)
('IZ7980z', '223957.82', 14)
('IK5270t', '222958.66', 15)
('QZ5910x', '220672.05', 16)
('SA7171t', '219848.23', 17)
('OH5301d', '219571.95', 18)
('GJ0907e', '218446.80', 19)
('HW2857f', '217170.17', 20)
('VI1711p', '216807.73', 21)
('OZ9708j', '216260.87', 22)
('MB2932y', '215347.57', 23)
('QB1722m', '214428.22', 24)
('VH3304e', '210674.22', 25)
('JS6246z', '210657.36', 26)
('XY9448i', '210128.09', 27)
('XV9570v', '206619.73', 28)
('VH0055x', '206197.17', 29)
('PZ1654y', '206019.22', 30)
('WG5211l', '204527.83', 31)
('PZ0047h', '202636.39', 32)
('LD6680k', '202298.28', 33)
('GI8367v', '201872.28', 34)
('XC3074w', '201700.00', 35)
('YA2147i', '195323.03', 36)
('MG7365o', '194567.08', 37)
('FM3293g', '193287.45', 38)
('SQ0135h', '192530.77', 39)
('FD1229x', '190030.75', 40)
('VS9753z', '187721.24', 41)
('EW5392j', '187609.81', 42)
('SQ2638i', '187548.58', 43)
('LI4475m', '187253.29', 44)
('GO7394f', '187250.90', 45)
('KK3760e', '187238.52', 46)
('BN0671t', '187013.38', 47)
('BW3485x', '184015.38', 48)
('NK9433u', '182073.30', 49)

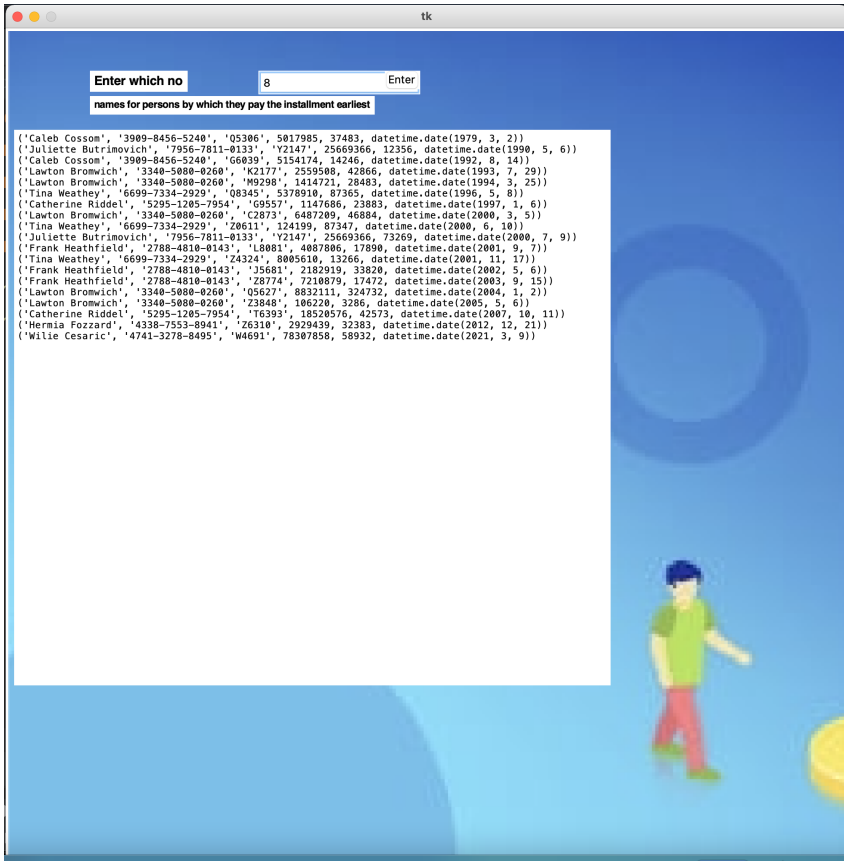

tk
Enter which no
1
Enter
Average salary for all departments
(1, 204527.83)
(2, 202636.39)
(3, 112224.17)
(4, 148946.66)
(5, 193287.45)
(6, 218446.8)
(7, 158449.83)
(8, 178737.82)
(9, 130895.46)
(10, 220672.05)
(11, 118520.46)
(12, 133179.45)
(13, 121502.87)
(14, 109831.18)
(15, 103368.34)
(16, 147216.31)
(17, 224275.22)
(18, 151141.9)
(19, 181319.07)
(20, 169935.49)
(21, 132081.67)
(22, 152528.07)
(23, 187609.81)
(24, 241574.08)
(25, 187721.24)
(26, 224784.05)
(27, 106519.99)
(28, 219571.95)
(29, 201872.28)
(30, 217170.17)
(31, 161465.68)
(32, 187238.52)
(33, 116192.45)
(34, 159736.96)
(35, 106483.61)
(36, 162325.57)
(37, 240259.02)
(38, 216807.73)
(39, 122703.78)
(40, 104637.46)
(41, 210128.09)
(42, 206197.17)
(43, 233520.56)
(44, 144248.13)
(45, 161828.95)
(46, 194567.08)
(47, 214428.22)
(48, 215347.57)
(49, 140645.25)
(50, 179975.63)

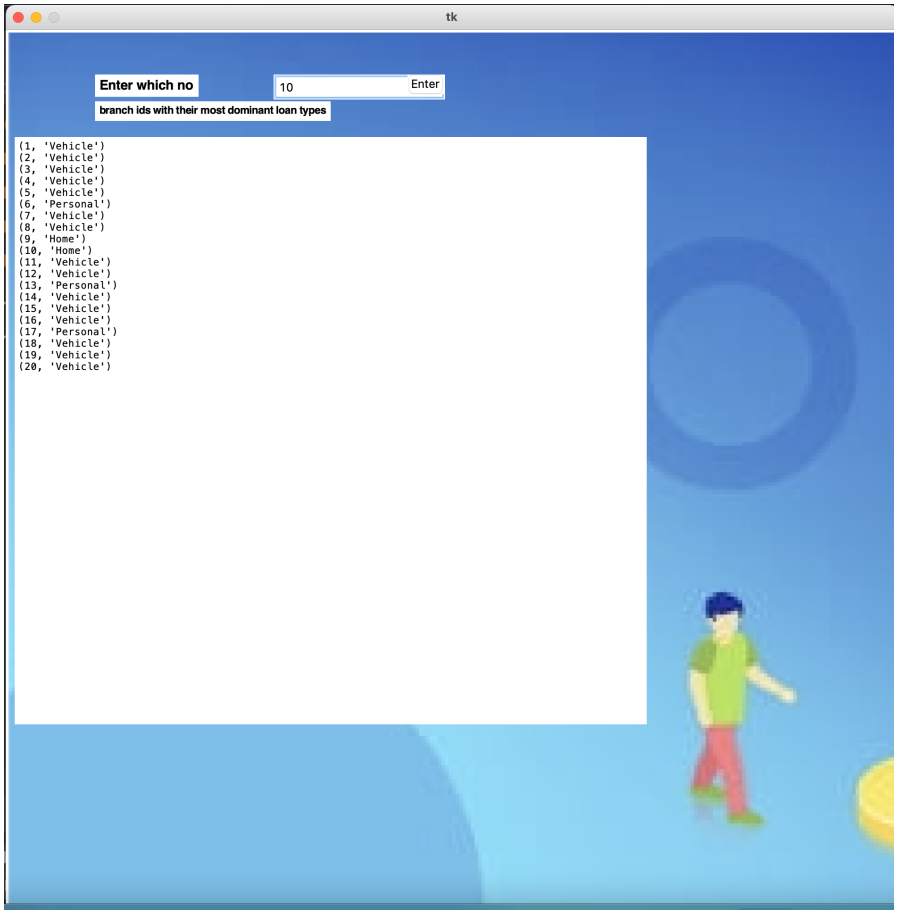

tk
Enter which no
3
Enter
details of the employee with the nth highest salary.
('Blake Cochrane', '242209.21')


tk
Enter which no
4
Enter
all Active Loans by Price and Interest
('Tina Weathey', '6699-7334-2929', 124199, 10, 'Repayment Ongoing', 'Gold')
('Cole Chippindale', '5589-9815-6742', 379515, 10, 'Repayment Ongoing', 'Personal')
('Cookie Floris', '9810-3591-7194', 717217, 10, 'Repayment Ongoing', 'Vehicle')
('Catherine Riddel', '5295-1205-7954', 1147686, 9, 'Repayment Ongoing', 'Home')
('Margalo Elldred', '1983-2713-2408', 1306185, 9, 'Repayment Ongoing', 'Business')
('Margalo Elldred', '1983-2713-2408', 2246196, 9, 'Repayment Ongoing', 'Vehicle')
('Wilbur Gritland', '3827-9175-7019', 2532743, 8, 'Repayment Ongoing', 'Educational')
('Carrissa Ibeson', '2916-6143-1155', 2597927, 9, 'Repayment Ongoing', 'Business')
('Carrissa Ibeson', '2916-6143-1155', 3125683, 9, 'Repayment Ongoing', 'Home')
('Wilbur Gritland', '3827-9175-7019', 3176897, 9, 'Repayment Ongoing', 'Vehicle')
('Davita Pladen', '1361-2909-1956', 3617155, 10, 'Repayment Ongoing', 'Gold')
('Albie Cancott', '3115-6511-4003', 3796742, 7, 'Repayment Ongoing', 'Personal')
('Farley Joddens', '0394-0836-0938', 4216250, 7, 'Repayment Ongoing', 'Vehicle')
('Ezekiel Troillet', '2267-6622-3253', 4656968, 9, 'Repayment Ongoing', 'Vehicle')
('Jock Matityahu', '4597-3766-7385', 7037634, 8, 'Repayment Ongoing', 'Gold')
('Frank Heathfield', '2788-4810-8143', 7210879, 8, 'Repayment Ongoing', 'Home')
('Julie Follett', '4328-4494-4111', 8723854, 7, 'Repayment Ongoing', 'Business')
('Nickolai California', '7548-6971-4573', 8740464, 9, 'Repayment Ongoing', 'Home')
('Kate Witsey', '5174-0230-5988', 8953327, 9, 'Repayment Ongoing', 'Home')
('Stormi Temprell', '6321-0225-5209', 9582839, 7, 'Repayment Ongoing', 'Home')
('Cookie Floris', '9810-3591-7194', 9942473, 8, 'Repayment Ongoing', 'Personal')
('Karrah Glencorse', '0060-9771-5565', 10578232, 6, 'Repayment Ongoing', 'Vehicle')
('Albie Cancott', '3115-6511-4003', 14194948, 8, 'Repayment Ongoing', 'Home')
('Jock Matityahu', '4597-3766-7385', 14425440, 6, 'Repayment Ongoing', 'Educational')
('Latrina Cluff', '8482-6235-2555', 18460816, 7, 'Repayment Ongoing', 'Home')
('Wilbur Gritland', '3827-9175-7019', 20323254, 8, 'Repayment Ongoing', 'Home')
('Latrina Cluff', '8482-6235-2555', 37874378, 6, 'Repayment Ongoing', 'Educational')
('Wilbur Gritland', '3827-9175-7019', 40989582, 9, 'Repayment Ongoing', 'Personal')


tk
Enter which no
5
Enter
Account Numbers,Aadhar Number and Loan amount of people who have taken a loan of amount 1cr and more and have 10cr or less in their accounts.
('8482-6235-2555', 'Ba-5993-ei3317', 34972423)
('3115-6511-4003', 'Ea-6651-gl7276', 14194948)
('8482-6235-2555', 'Ba-5993-ei3317', 18460816)
('3115-6511-4003', 'Ea-6651-gl7276', 20353482)
('5295-1205-7954', 'Bh-7570-qj2286', 18520576)
('4328-4494-4111', 'Ru-7251-vq8585', 67631717)
('8482-6235-2555', 'Ba-5993-ei3317', 37874378)
('7956-7811-0133', 'Mq-2090-ri7119', 25669366)


tk
Enter which no
7
Enter
all persons having more than 1 account
('Michel Innott', '6270-1096-4714', 5)
('Ginelle Halworth', '9232-1888-9502', 5)
('Albie Cancott', '3115-6511-4003', 5)
('Conn Omand', '2822-9145-5943', 5)
('Hilary Demange', '2227-9911-9278', 5)
('Antonin Cotherill', '4892-5262-4965', 4)
('Rey Gallen', '5270-0225-7750', 4)
('Dalila Hulk', '5598-1054-3576', 4)
('Murry McGhee', '5631-8425-4226', 4)
('Emera Cutmore', '0522-1052-0575', 4)
('Karee Gehrts', '6224-6827-6983', 4)
('Eugenio McLean', '9787-2919-4979', 4)
('Cleo Sainter', '0798-5908-8363', 4)
('Mercie Edden', '3632-1550-3415', 4)
('Davita Katt', '9314-0616-2902', 4)
('Juliette Butrimovich', '7956-7811-0133', 3)
('Ashlie Loveless', '2313-7088-9573', 3)
('Chrissy Desseine', '7525-1086-3101', 3)
('Billi Enrico', '2633-7814-9297', 3)
('Jania Swigg', '0005-5989-5302', 3)
('Virgina Seeley', '7004-5925-5651', 3)
('Claudell Dunhill', '8104-4564-8357', 3)
('Bartram Tampling', '2823-8252-3607', 3)
('Letitia Venes', '6476-8429-3979', 3)
('Carrissa Ibeson', '2916-6143-1155', 3)
('Joshua Gibberd', '8773-1461-1289', 3)
('Elsie Gagg', '9583-3760-4918', 3)
('Averil Merredy', '3458-5062-8839', 3)
('Persis Boshers', '1239-9825-8586', 3)
('Oralia Swyer', '3700-4525-6311', 3)
('Cookie Floris', '9810-3591-7194', 3)
('Adel Tabourin', '5817-2008-1394', 3)
('Kalie Machan', '0176-6933-7158', 3)
('Herb Broadbent', '0171-3393-2969', 3)
('Andreas Petruskevich', '5572-2381-1317', 3)
('Karrah Glencorse', '0060-9771-5565', 3)
('Kate Witsey', '5174-0230-5988', 3)
('Jo ann Oloshkin', '2695-5945-7970', 3)
('Esther Knewstubb', '4885-3844-0043', 2)
('Lowrance Treadgall', '4670-4102-6613', 2)
('Catherine Riddel', '5295-1205-7954', 2)
('Jethro Mitchinson', '5391-1284-5131', 2)
('Barn Scoullar', '5461-5366-6278', 2)
('Allix Hearons', '5493-8224-8441', 2)
('Hammad Rentalll', '5513-3225-8629', 2)
('Carmita Schober', '3954-2559-8387', 2)
('Lind Beckers', '3945-4356-3304', 2)
('Caleb Cossom', '3909-8456-5240', 2)
('Colas Guswell', '3896-7276-9222', 2)
('Sigfried Sparke', '3794-7767-7452', 2)


tk
Enter which no
8
Enter
names for persons by which they pay the installment earliest
('Caleb Cossom', '3909-8456-5240', 'Q5306', 5017985, 37483, datetime.date(1979, 3, 2))
('Juliette Butrimovich', '7956-7811-0133', 'Y2147', 25669366, 12356, datetime.date(1990, 5, 6))
('Caleb Cossom', '3909-8456-5240', 'G6039', 5154174, 14246, datetime.date(1992, 8, 14))
('Lawton Bromwich', '3340-5080-0260', 'K2177', 2559508, 42866, datetime.date(1993, 7, 29))
('Lawton Bromwich', '3340-5080-0260', 'M9298', 1414721, 28483, datetime.date(1994, 3, 25))
('Tina Weathey', '6699-7334-2929', 'Q8345', 5378910, 87365, datetime.date(1996, 5, 8))
('Catherine Riddel', '5295-1205-7954', 'G9557', 1147686, 23883, datetime.date(1997, 1, 6))
('Lawton Bromwich', '3340-5080-0260', 'C2873', 6487209, 46884, datetime.date(2000, 3, 5))
('Tina Weathey', '6699-7334-2929', 'Z0611', 124199, 87347, datetime.date(2000, 6, 10))
('Juliette Butrimovich', '7956-7811-0133', 'Y2147', 25669366, 73269, datetime.date(2000, 7, 9))
('Frank Heathfield', '2788-4810-0143', 'L8081', 4087806, 17890, datetime.date(2001, 9, 7))
('Tina Weathey', '6699-7334-2929', 'Z4324', 8005610, 13266, datetime.date(2001, 11, 17))
('Frank Heathfield', '2788-4810-0143', 'J5681', 2182919, 33820, datetime.date(2002, 5, 6))
('Frank Heathfield', '2788-4810-0143', 'Z8774', 7210879, 17472, datetime.date(2003, 9, 15))
('Lawton Bromwich', '3340-5080-0260', 'Q5627', 8832111, 324732, datetime.date(2004, 1, 2))
('Lawton Bromwich', '3340-5080-0260', 'Z3848', 106220, 3286, datetime.date(2005, 5, 6))
('Catherine Riddel', '5295-1205-7954', 'T6393', 18520576, 42573, datetime.date(2007, 10, 11))
('Hermia Fozzard', '4338-7553-8941', 'Z6310', 2929439, 32383, datetime.date(2012, 12, 21))
('Wilie Cesaric', '4741-3278-8495', 'W4691', 78307858, 58932, datetime.date(2021, 3, 9))



Users and Grants:

We have to give each table in the database access to a user to view and update. With this in mind, 3 users have been created.

1)

```
940
941 •   create user 'employee'@'localhost' identified by 'user_passwd';
942 •   GRANT select, insert, delete, update on CUSTOMER to 'employee'@'localhost';
943 •   GRANT select, insert, delete, update on LOAN to 'employee'@'localhost';
944 •   GRANT select, insert, delete, update on ACCOUNT to 'employee'@'localhost';
945 •   GRANT select, insert, delete, update on Payments to 'employee'@'localhost';
946
```

This is a user Employee who can access and update the Customer, Loan, Account and Payments tables.

2)

```
947 •   create user 'director'@'localhost' identified by 'user_passwd';
948 •   GRANT select, insert, delete, update on BANKBRANCH to 'director'@'localhost';
949 •   GRANT select, insert, delete, update on Department to 'director'@'localhost';
950 •   GRANT select, insert, delete, update on Manages to 'director'@'localhost';
```

This is a user Director, who has the privilege to access and update the Bank Branch, Department and Manages tables. Basically, the director can open/close a bank branch, decide to add/remove departments, and appoint/fire managers of the branches.

```
951
952 •   create user 'departmentHead'@'localhost' identified by 'user_passwd';
953 •   GRANT select, insert, delete, update on EMPLOYEE to 'departmentHead'@'localhost';
954
```

3)

This is the user credential for the head of department, who will maintain the Employee table. This user can access and update the Employee Table to add new employees/ remove previous employees and so on.

NOTE that these are tentative hierarchical users as generally observed in institutions. A general employee maintains the customer records, a HR/ Managerial level person handles the Employee details, and the general features like number of bank branches, departments, and managers are overseen by a higher authority (in this case the director).

Triggers:

1. Created a student_discount trigger(insert type) for LOAN Table that checks if the loan type is ‘Educational’ then the Interest on loan is reduced by 2%.

```
1009 •  create trigger student_discount BEFORE insert
1010    on LOAN for each row begin
1011    if (NEW.type='Educational') then
1012        set New.Interest=New.Interest-2;
1013    end if;
1014    end// -- trigger-1
1015    insert into LOAN (LoanNumber, Amount, Interest, Type, AadharNumber, BranchNumber, Status) values ('X2098', '98939085', 7, 'Educational', '4597-3766-7385', '7', 'Gra
1016    select * from Loan; -- testing the trigger-1
```

2. Created a close_account trigger(update type) for ACCOUNT Table that makes the balance=0 if the account status is ‘Closed’.

```
1018 •  create trigger close_account BEFORE update
1019    on ACCOUNT for each row begin
1020    if (NEW.Status='Closed') then
1021        set New.Balance=0;
1022    end if;
1023    end// -- trigger-2
1024    select * from account where AccountNumber='Ib-8754-rg0353';
1025    update account
1026    set status='Closed'
1027    where AccountNumber='Ib-8754-rg0353'; -- testing trigger-2
```

3. Created a salary_update trigger(update type) for Employee Table that increases the salary of an employee by 5% whenever he/she has a department change.

```
1030 •  create trigger update_salary before update
1031    on Employee for each row begin
1032    if (NEW.DepartmentNumber <> OLD.DepartmentNumber) then
1033        set NEW.salary = 1.05*Old.Salary;
1034    end if;
1035    end//
1036    update employee set DepartmentNumber = '5' where employeeID='VY0495h';
1037    select salary from employee where employeeID='VY0495h';
```

4. Created an open_account trigger(insert type) for ACCOUNT Table that is whenever a customer opens his bank account, he/she needs to deposit more than 1000 rupees then the account would not be inserted and will show ‘NOT ENOUGH BALANCE.

```
1040 •  create trigger open_account before insert
1041    on ACCOUNT for each row begin
1042    if (NEW.balance<1000) then
1043        SIGNAL SQLSTATE '45000'
1044        SET MESSAGE_TEXT = 'NOT ENOUGH BALANCE';
1045    end if;
1046    end//
1047 •  insert into ACCOUNT (AccountNumber, Balance, Type, Status, AadharNumber, BranchNumber) values ('Rk-5607-nk4027', '176', 'Savings', 'Closed', '3909-8456-5240', 5);
1048
```

Member Contribution:

Aayush Kapoor- ER Diagram, SQL Queries,Optimization,Embedded
Devansh Arora- ER Diagram,Relational Schema, SQL Queries,Embedded,UI,triggers
Prasun Pratik- ER Diagram, Data Population,Indexes, Users and Grants, Triggers, and Views.
Rohan Gupta- ER Diagram, Relational Schema, Data Population, Indexes, Users and Grants, Triggers, and Views.