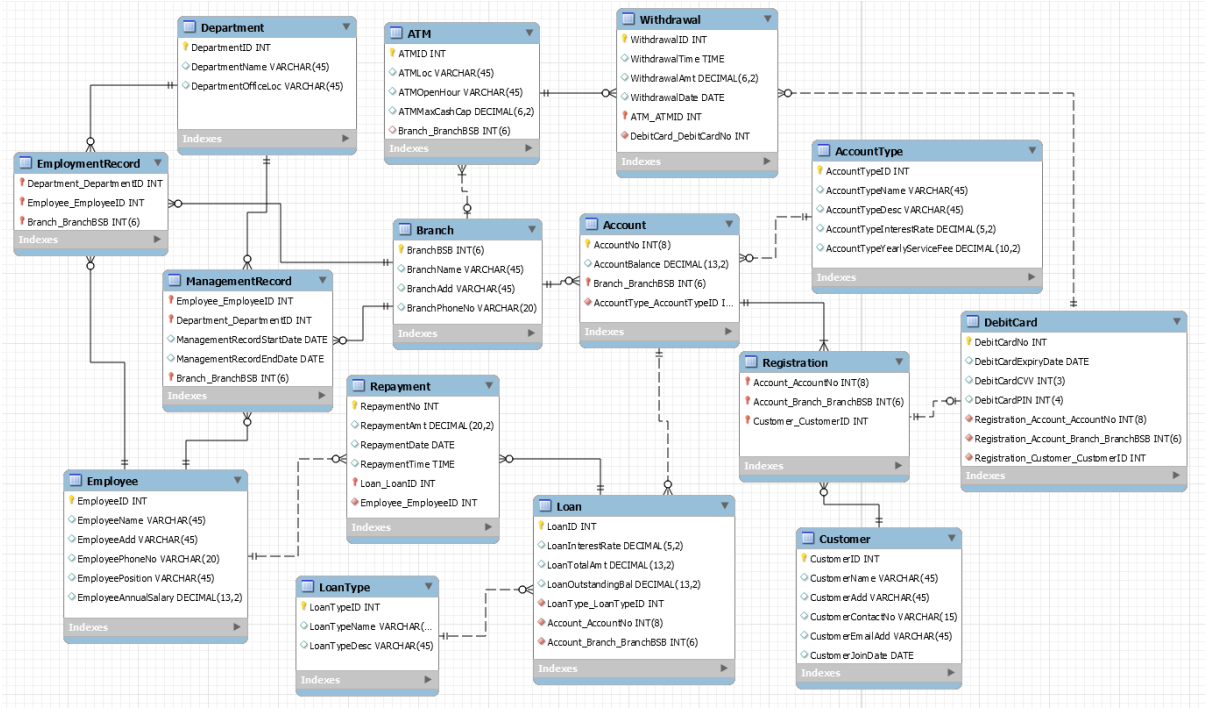
ISYS224 2018 – ASSIGNMENT ONE

StudentID: 45282188
StudentName: Chun Hei Matthew Lee
Workshop Number: week 7
Tutor Name: Nardin Hanna(Tutorial)**,** Jandson Santos(Practical)
Workshop Day and Time: Wednesday 1-2(Tutorial)**,** Thursday 2-3(Practical)

Task 1: Data model

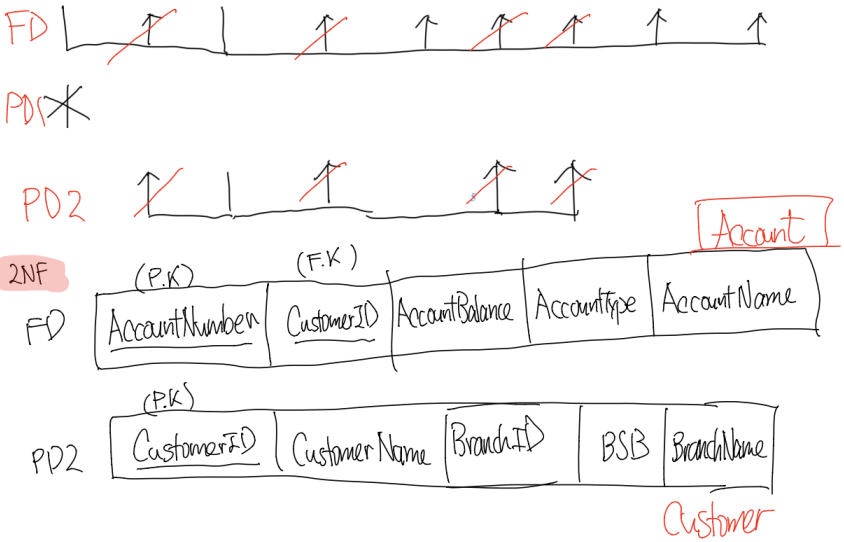


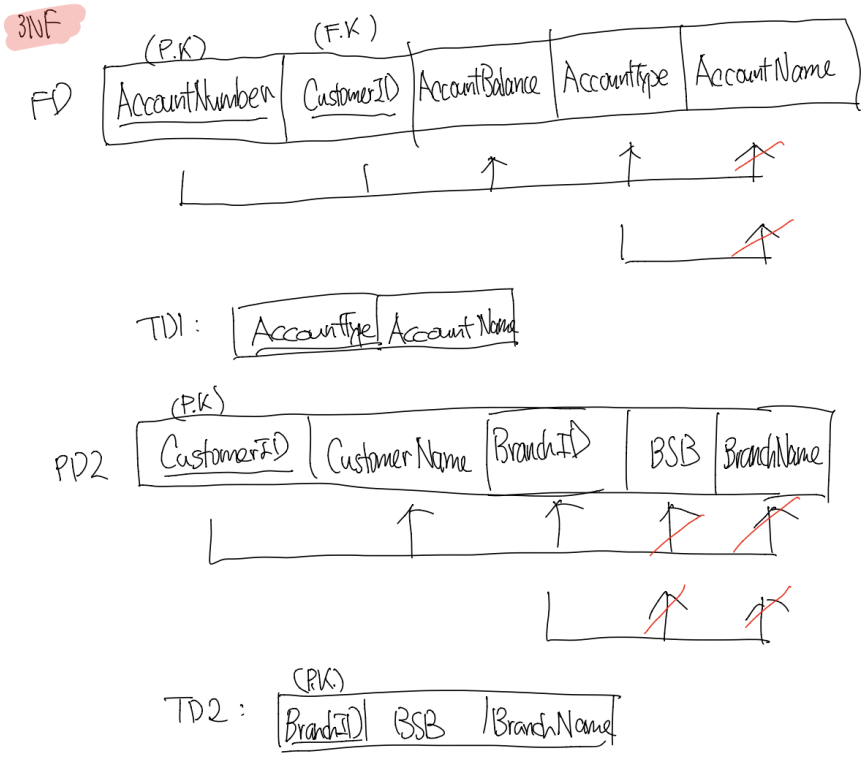
Task 2: Normalisation

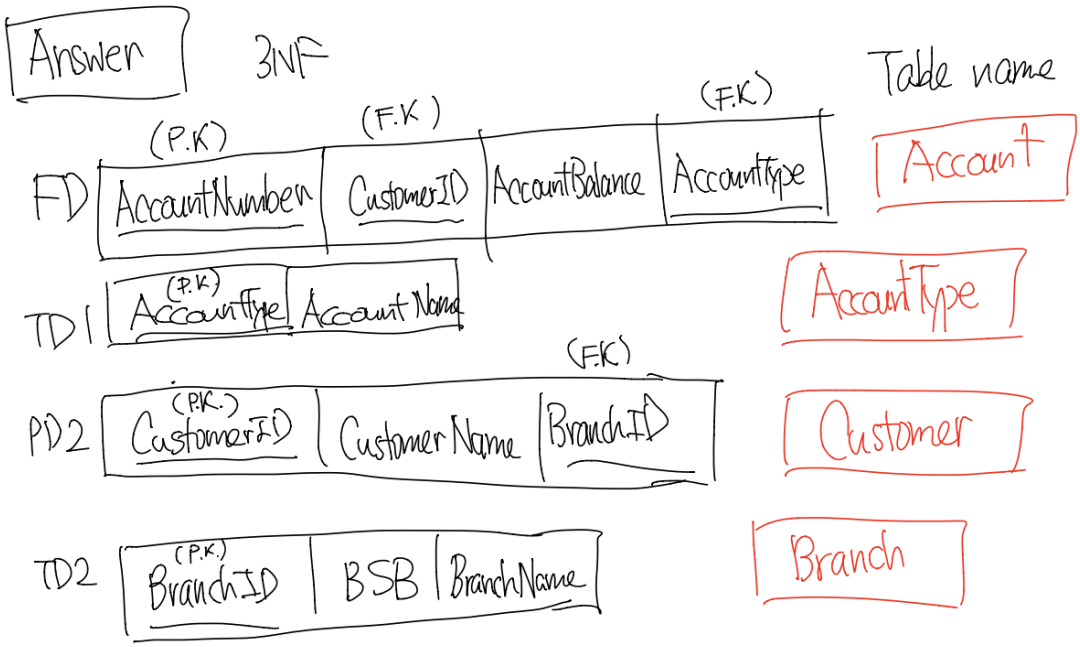
The given table is in this normal form: 1

Decomposition into normal form: 3

AccountNumber	BSB	CustomerID	CustomerName	AccountBalance	BranchID	BranchName	Account Type	Account Name
9047 1234	062 345	C001	Loki	500	Branch1	Burwood	SAV	Savings
9047 1234	062 345	C002	Iron Man	500	Branch1	Burwood	SAV	Savings
9123 2313	062 345	C001	Loki	3000	Branch1	Burwood	CHECK	Checking
9132 3412	024 213	C003	Thor	5400	Branch2	Ashfield	GOAL	Goal Saver
9546 2134	024 213	C003	Thor	2300	Branch2	Ashfield	CHECK	Checking
9546 2134	024 213	C004	Black Widow	2300	Branch2	Ashfield	CHECK	Checking
9021 3241	034 234	C005	Wanda. M	4000	Branch3	Hornsby	IRA	Retirement
9047 1234	021 123	C006	Hulk	800	Branch4	Epping	IRA	Retirement







Task 3 -DDL

It is in the script uploaded.

Task 4: DML

```
insert  into `AccountType`(`AccountTypeID`,`AccountTypeName`,`AccountTypeDesc`,`AccountTypeInterestRate`,`AccountTypeYearlyServiceFee`)
values
(001,'SAV','Saving',5.00,50.00),
(002,'CHECK','Checking',6.00,80.00),
(003,'GOAL','Goal Saver',5.50,70.00),
(004,'IRA','Retirement',8.00,85.00),
(005,'LOAN','Loan',5.00,120.00),
(006,'JOINT','Joint Account',7.00,100.00);


insert  into `Customer`(`CustomerID`,`CustomerName`,`CustomerAdd`,`CustomerContactNo`,`CustomerEmailAdd`,`CustomerJoinDate`)
values
(0001,'Matthew Lee','11/22 CYBER ST STRATHFIELD','0433322112','matthewlee123@gmail.com','2007-03-15'),

(0002,'Mary Lam','21/12 HELLOWEEN ST BLACKTOWN','0435635678','marylam332@gmail.com','2006-09-12'),

(0003,'Jay Chou','12/22 TAIWAN ST BURWOOD','0563956745','jaychou213@gmail.com','2013-05-07'),

(0004,'Selina Tusi','01/09 BABY ST MACQUARIE PARK','0876567853','selinatusi3322@gmail.com','2017-11-12'),

(0005,'Ella Tam','11/09 WOOLWORTH ST MACQUARIE UNIVERSITY','0876567333','ellatam1444@gmail.com','2018-02-12'),

(0006,'Yvonne Lam','11/29 BABY ST HURSTVILLE','0876537853','yvonnelam2@gmail.com','2015-07-12');

insert  into `LoanType`(`LoanTypeID`,`LoanTypeName`,`LoanTypeDesc`)
values
(1,'STU','student loan'),
(2,'MORT','mortgage loan'),
(3,'PL','personal loan'),
(4,'BUS','business loan'),
(5,'CAR','car loan');

insert into `Branch`(`BranchBSB`,`BranchName`,`BranchAdd`,`BranchPhoneNo`)
values
(123450,'Burwood','11/20 BURWOOD ST BURWOOD','24759655'),
(123451,'Strathfield','11/11 STRATHFIELD ST BURWOOD','29987775'),
(123452,'Rhodes','11/20 PARRAMATTA ST RHODES','29988775'),
(123453,'Parramatta','02/02 PARRAMATTA ST PARRAMATTA','21102201'),
(123454,'Epping','11/20 EPPING ST EPPING','29975555'),
(123455,'Eastwood','11/20 EASTWOOD ST EASTWOOD','24434433');
```

```
insert into `Employee`(`EmployeeID`,`EmployeeName`,`EmployeeAdd`,`EmployeePhoneNo`,`EmployeePosition`,`EmployeeAnnualSalary`)
values
(001,'Lawerence Chung','11/22 STRATHFIELD ST STRATHFIELD','0556443454','SERVICE MANAGER',100000.00),
(002,'TsiTsi Tam','01/01 BURWOOD ST BURWOOD','0456456456','TELLER',55000.00),
(003,'Matthew Vi','10/02 EPPING ST EPPING','0444444322','TELLER',60000.00),
(004,'Michael Lee','10/02 EASTWOOD ST EASTWOOD','0111222334','LOAN ASSISTANT',59000.00),
(005,'Yvonne Ti','10/02 DENISTONE ST DENISTONE','01111112222','LOAN MANAGER',130000.00),
(006,'Fi Lee','07/02 WEST ST WESTLY','0333442253','TECHNICIAN MANAGER',80000.00),
(007,'Lilly Lee','10/12 ANGEL ST ANGEL','0321123321','Human Resource Manager',70000.00),
(008,'Rick Chung','11/02 TIME ST TIME','0890789789','TELLER',60000.00),
(009,'Chris Delgado','21/02 DREAM ST DREAM','0297986875','TELLER',65000.00),
(010,'Michael Yip','11/22 MALAYSIAN ST MA','0556443434','SERVICE MANAGER',100000.00);

insert into `Department`(`DepartmentID`,`DepartmentName`,`DepartmentOfficeLoc`) values
(1,'Loan','1st floor'),
(2,'Customer Service','Ground floor'),
(3,'Information Technology','2rd floor'),
(4,'Human Resource','2rd floor');

insert into `ATM`(`ATMID`,`ATMLoc`,`ATMOpenHour`,`ATMMaxCashCap`,`Branch_BranchBSB`)
values
(1,'Burwood','6:00',2000.00,123450),
(2,'Strathfield','6:00',2000.00,123451),
(3,'Rhodes','6:00',2000.00,123452),
(4,'Parramatta','6:00',2000.00,123453),
(5,'Epping','6:00',2000.00,123454),
(6,'Eastwood','6:00',2000.00,123455);

insert into `Account`(`AccountNo`,`AccountBalance`,`Branch_BranchBSB`,`AccountType_AccountTypeID`) values

(12345678,5000.00,123450,001),
(23456789,2000.00,123450,002),
(63456789,20000.00,123450,006),
(53456789,100050.20,123451,004),
(08080808,50000.00,123451,005),
(09980808,40000.00,123452,005),
(10080808,70000.00,123452,005);
```

```
insert into `Registration`(`Account_Branch_BranchBSB`,`Account_AccountNo`,`Customer_CustomerID`)
values
(123450,12345678,001),
(123450,23456789,002),
(123451,53456789,005),
(123450,63456789,005),
(123450,63456789,006),
(123451,08080808,001),
(123452,09980808,002),
(123452,10080808,003);


insert into `DebitCard`(`DebitCardNo`,`DebitCardExpiryDate`,`DebitCardCVV`,`DebitCardPIN`,`Registration_Account_Branch_BranchBSB`,`Registration_Account_AccountNo`,`Registration_Customer_CustomerID`)
values
(001,'2020-01-20',123,1234,123450,12345678,001),
(002,'2021-12-01',234,2341,123450,12345678,001),
(003,'2020-04-04',345,3451,123450,63456789,005),
(004,'2020-05-11',456,4561,123450,63456789,006);

insert into `Withdrawal`(`WithdrawalID`,`WithdrawalTime`,`WithdrawalAmt`,`WithdrawalDate`,`DebitCard_DebitCardNo`,`ATM_ATMID`)
values
(0001,'13:30:12',200.00,'2018-03-05',001,005),
(0002,'16:24:53',150.00,'2018-04-15',002,001),
(0003,'09:11:23',150.00,'2018-05-15',003,002),
(0004,'10:41:01',120.00,'2018-08-15',004,003),
(0005,'10:45:01',120.00,'2018-08-15',004,003);

insert into `Loan`(`LoanID`,`LoanInterestRate`,`LoanTotalAmt`,`LoanOutstandingBal`,`Account_Branch_BranchBSB`,`Account_AccountNo`,`LoanType_LoanTypeID`) values

(001,5.00,100000.00,5000.00,123451,08080808,1),
(002,5.00,150000.00,7500.00,123452,09980808,2),
(003,5.00,100000.00,5000.00,123452,10080808,3);
```

```
insert into `Repayment`(`RepaymentNo`,`RepaymentAmt`,`RepaymentDate`,`RepaymentTime`,`Employee_EmployeeID`,`Loan_LoanID`) values
(001,2000.00,'2018-03-03','13:13:12',004,001),
(002,3000.00,'2018-04-03','15:13:12',004,002),
(003,4000.00,'2018-05-03','13:15:12',004,003);


Insert into `EmploymentRecord`(`Department_DepartmentID`,`Employee_EmployeeID`,`Branch_BranchBSB`)values
(2,002,123450),
(2,003,123450),
(1,004,123450),
(2,008,123450),
(2,009,123450);

Insert into `ManagementRecord` (`Employee_EmployeeID`,`Department_DepartmentID`,`ManagementRecordStartDate`,`ManagementRecordEndDate`,`Branch_BranchBSB`)
values
(001,2,'2018-01-01','2019-01-01',123450),
(005,1,'2018-01-01','2019-01-01',123450),
(006,3,'2018-01-01','2019-01-01',123450),
(007,4,'2018-01-01','2019-01-01',123450),
(010,2,'2018-01-01','2019-01-01',123451);
```

AccountType

AccountTypeID	AccountTypeName	AccountTypeDesc	AccountTypeInterestRate	AccountTypeYearlyServiceFee
1	SAV	Saving	5.00	50.00
2	CHECK	Checking	6.00	80.00
3	GOAL	Goal Saver	5.50	70.00
4	IRA	Retirement	8.00	85.00
5	LOAN	Loan	5.00	120.00
6	JOINT	Joint Account	7.00	100.00

Customer

CustomerID	CustomerName	CustomerAdd	CustomerContactNo	CustomerEmailAdd	CustomerJoinDate
1	Matthew Lee	11/22 CYBER ST STRATHFIELD	0433322112	matthewlee123@gmail.com	2007-03-15
2	Mary Lam	21/12 HELLOWEEN ST BLACKTOWN	0435635678	marylam332@gmail.com	2006-09-12
3	Jay Chou	12/22 TAIWAN ST BURWOOD	0563956745	jaychou213@gmail.com	2013-05-07
4	Selina Tusi	01/09 BABY ST MACQUARIE PARK	0876567853	selinatusi3322@gmail.com	2017-11-12
5	Ella Tam	11/09 WOOLWORTH ST MACQUARIE UNIVE…	0876567333	ellatam1444@gmail.com	2018-02-12
6	Yvonne Lam	11/29 BABY ST HURSTVILLE	0876537853	yvonnelam2@gmail.com	2015-07-12

LoanType

LoanTypeID	LoanTypeName	LoanTypeDesc
1	STU	student loan
2	MORT	mortgage loan
3	PL	personal loan
4	BUS	business loan
5	CAR	car loan

Branch

BranchBSB	BranchName	BranchAdd	BranchPhoneNo
123450	Burwood	11/20 BURWOOD ST BURWOOD	24759655
123451	Strathfield	11/11 STRATHFIELD ST BURWOOD	29987775
123452	Rhodes	11/20 PARRAMATTA ST RHODES	29988775
123453	Parramatta	02/02 PARRAMATTA ST PARRAMATTA	21102201
123454	Epping	11/20 EPPING ST EPPING	29975555
123455	Eastwood	11/20 EASTWOOD ST EASTWOOD	24434433

Employee

EmployeeID	EmployeeName	EmployeeAdd	EmployeePhoneNo	EmployeePosition	EmployeeAnnualSalary
1	Lawerence Chung	11/22 STRATHFIELD ST STRATHFIELD	0556443454	SERVICE MANAGER	100000.00
2	TsiTsi Tam	01/01 BURWOOD ST BURWOOD	0456456456	TELLER	55000.00
3	Matthew Vi	10/02 EPPING ST EPPING	0444444322	TELLER	60000.00
4	Michael Lee	10/02 EASTWOOD ST EASTWOOD	0111222334	LOAN ASSISTANT	59000.00
5	Yvonne Ti	10/02 DENISTONE ST DENISTONE	01111112222	LOAN MANAGER	130000.00
6	Fi Lee	07/02 WEST ST WESTLY	0333442253	TECHNICIAN MANAGER	80000.00
7	Lilly Lee	10/12 ANGEL ST ANGEL	0321123321	Human Resource Manager	70000.00
8	Rick Chung	11/02 TIME ST TIME	0890789789	TELLER	60000.00
9	Chris Delgado	21/02 DREAM ST DREAM	0297986875	TELLER	65000.00
10	Michael Yip	11/22 MALAYSIAN ST MA	0556443434	SERVICE MANAGER	100000.00

Department

DepartmentID	DepartmentName	DepartmentOfficeLoc
1	Loan	1st floor
2	Customer Service	Ground floor
3	Information Technology	2rd floor
4	Human Resource	2rd floor

ATM

ATMID	ATMLoc	ATMOpenHour	ATMMaxCashCap	Branch_BranchBSB
1	Burwood	6:00	2000.00	123450
2	Strathfield	6:00	2000.00	123451
3	Rhodes	6:00	2000.00	123452
4	Parramatta	6:00	2000.00	123453
5	Epping	6:00	2000.00	123454
6	Eastwood	6:00	2000.00	123455

Account

AccountNo	AccountBalance	Branch_BranchBSB	AccountType_AccountTypeID
8080808	50000.00	123451	5
9980808	40000.00	123452	5
10080808	70000.00	123452	5
12345678	5000.00	123450	1
23456789	2000.00	123450	2
53456789	100050.20	123451	4
63456789	20000.00	123450	6

Registration

Account_AccountNo	Account_Branch_BranchBSB	Customer_CustomerID
8080808	123451	1
12345678	123450	1
9980808	123452	2
23456789	123450	2
10080808	123452	3
53456789	123451	5
63456789	123450	5
63456789	123450	6

DebitCard

DebitCardNo	DebitCardExpiryDate	DebitCardCVV	DebitCardPIN	Registration_Account_AccountNo	Registration_Account_Branch_BranchBSB	Registration_Customer_CustomerID
1	2020-01-20	123	1234	12345678	123450	1
2	2021-12-01	234	2341	12345678	123450	1
3	2020-04-04	345	3451	63456789	123450	5
4	2020-05-11	456	4561	63456789	123450	6

Withdrawal

WithdrawalID	WithdrawalTime	WithdrawalAmt	WithdrawalDate	ATM_ATMID	DebitCard_DebitCardNo
1	13:30:12	200.00	2018-03-05	5	1
2	16:24:53	150.00	2018-04-15	1	2
3	09:11:23	150.00	2018-05-15	2	3
4	10:41:01	120.00	2018-08-15	3	4
5	10:45:01	120.00	2018-08-15	3	4

Loan

LoanID	LoanInterestRate	LoanTotalAmt	LoanOutstandingBal	LoanType_LoanTypeID	Account_AccountNo	Account_Branch_BranchBSB
1	5.00	100000.00	5000.00	1	8080808	123451
2	5.00	150000.00	7500.00	2	9980808	123452
3	5.00	100000.00	5000.00	3	10080808	123452

Repayment

RepaymentNo	RepaymentAmt	RepaymentDate	RepaymentTime	Loan_LoanID	Employee_EmployeeID
1	2000.00	2018-03-03	13:13:12	1	4
2	3000.00	2018-04-03	15:13:12	2	4
3	4000.00	2018-05-03	13:15:12	3	4

EmploymentRecord

Department_DepartmentID	Employee_EmployeeID	Branch_BranchBSB
2	2	123450
2	3	123450
1	4	123450
2	8	123450
2	9	123450

ManagementRecord

Employee_EmployeeID	Department_DepartmentID	ManagementRecordStartDate	ManagementRecordEndDate	Branch_BranchBSB
1	2	2018-01-01	2019-01-01	123450
5	1	2018-01-01	2019-01-01	123450
6	3	2018-01-01	2019-01-01	123450
7	4	2018-01-01	2019-01-01	123450
10	2	2018-01-01	2019-01-01	123451

Task 5: DML

--add a screenshot of your query outputs

1.

```
select count(accountNo), AccountTypeName
from Account a, AccountType at
where a.AccountType_AccountTypeID=at.AccountTypeID
and AccountTypeName='JOINT'
group by AccountTypeName;
```

Result Grid | Filter Rows:

count(accountNo)	AccountTypeName
1	JOINT

2.

```
select a.AccountNo , a.Branch_BranchBSB
from Account a, Loan l
where a.AccountNo=l.Account_AccountNo and
a.Branch_BranchBSB = l.Account_Branch_BranchBSB and
l.LoanType_LoanTypeID in (select LoanTypeID from LoanType where LoanTypeName ='PL');
```

AccountNo	Branch_BranchBSB
10080808	123452

3.

```
select e.EmployeeName, d.DepartmentName, b.BranchName
from Employee e, Department d, ManagementRecord mr, Branch b
where e.EmployeeID=mr.Employee_EmployeeID and
d.DepartmentID= mr.Department_DepartmentID and
b.BranchBSB= mr.Branch_BranchBSB;
```

EmployeeName	DepartmentName	BranchName
Yvonne Ti	Loan	Burwood
Lawerence Chung	Customer Service	Burwood
Michael Yip	Customer Service	Strathfield
Fi Lee	Information Technology	Burwood
Lilly Lee	Human Resource	Burwood

4.

```
select e.EmployeeName
from Employee e, ManagementRecord mr
where e.EmployeeID= mr.Employee_EmployeeID and
e.EmployeeAnnualSalary >( select avg(EmployeeAnnualSalary)
from Employee e
where e.EmployeeID in (select Employee_EmployeeID from ManagementRecord));
```

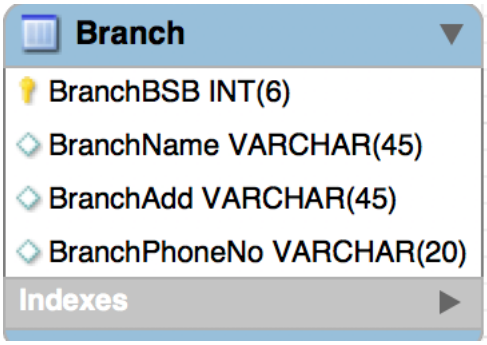
EmployeeName
Lawerence Chung
Yvonne Ti
Michael Yip

ASSUMPTION

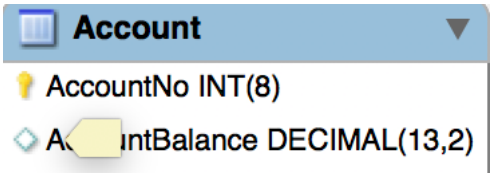
Task 1:

Assumption for creating relations

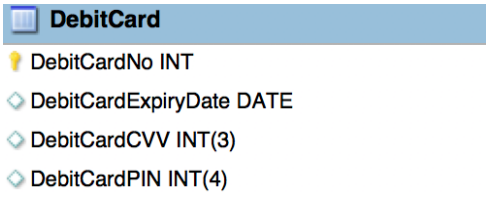
1. In branch, I use **BSB** as unique identifier instead of putting BSB in my 'Account' relation, as I assume that banks can be distinguished using BSB irrespective to their location, for example, BSB for Burwood branch NSW is different with BSB for Burwood VIC. Assume 6 digits BSB can provide large unique combination -(assume the BSB doesn't start with a zero). **BranchPhoneNo** is set to VARCHAR(20) as I assume phone number can include symbols.



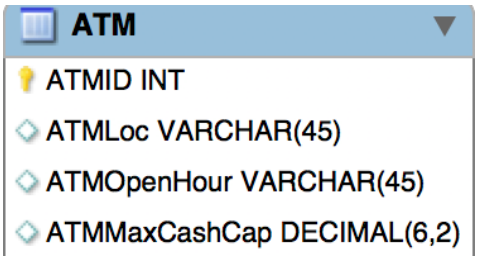
2. I chose not to have **BSB** in 'Account' relation as the assumption is made in 1. AccountNo is set to 8 integer digits as assumption. AccountBalance is assumed to be less than 13 digits including 2 decimal places.



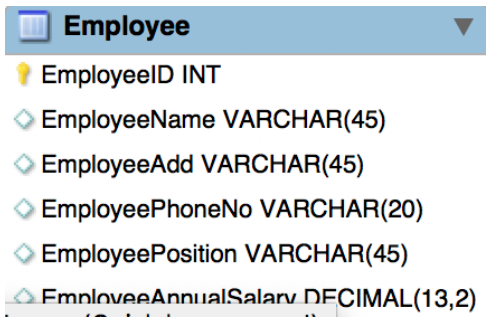
3. In DebitCard, CVV is assumed to be 3 integer digits; DebitCardPIN is assumed to be 4 digit numbers.



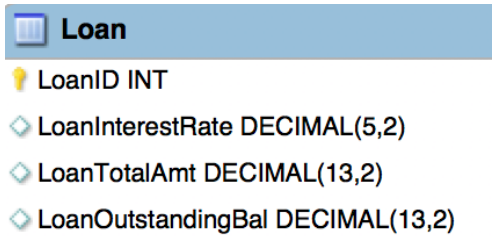
4. In ATM, ATMMaxCashCap is assumed to be 6 digits number including 2 decimal places.



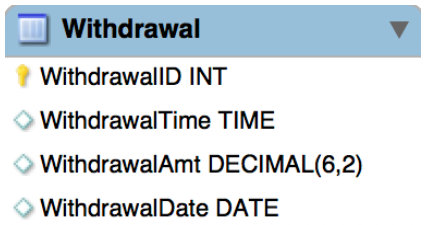
5. In Employee relation, EmployeeAnnualSalary is assumed to be 13 digit numbers including 2 decimal places. Phone number is assumed to be not just numerical values.



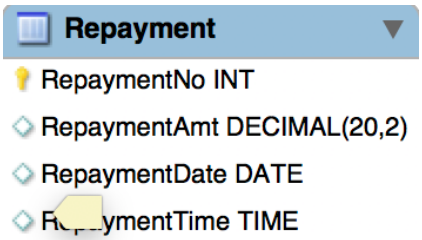
6. In loan relation, 'LoaninterestRate', 'LoanTotalAmt', and 'LoanOutstandingBal' are assumed be to 5 digit number including 2 decimal places, 13 digit number including 2 decimal places and 13 digit number including 2 decimal places respectively.



7. In withdrawal relation, 'WithdrawalAmt' is assumed to be 6 digit number including 2 decimal places.



8. In repayment relation, 'RepaymentAmt' is assumed to be 20 digit number including 2 decimal places as repayment can be one-time payment or split repayment.



Assumption for relationship

1. Branch vs ATM

I use non-identifying 1 to many relationship because ATM, as a strong entity, uses a unique identifier-ATMID as I assume the ATM is created with an unique ID by manufacturer but by branch. Therefore, it is assumed there is no duplicate ATMID irrespective to branch location. Assumption for the relationship is that one branch can have multiple ATM or at least one ATM , and each ATM may or may not associated with branch as ATM can be existed not in bank.

2. AccountType vs Account
I use many to one relationship here because one AccountType can have minimum no accounts (when you first create the accountType)or many accounts; one Account must at least belong to one accountType and at most one accountType. Non-identifying relationship used here because account with a composite key of BSB and AccountNo provides uniqueness, so only need foreign key from AccountType.

3. Customer vs Account
Many to many relationship is used here as I assume that each customer can have no accounts (assume it happens when they first join the bank or terminate the registration but they are still customers) or many accounts, and each account belongs to at least one customer or many customers as joint account.

4. Customer(Register) vs DebitCard
Since I assume that only customers associated with accounts can apply debit cards, so I created a relationship between Register and DebitCard relations. 1 to 1 relationship is used here as each account's customer can choose to get no debit cards or at most 1 debit card for security issue, and each card belongs to at least 1 customer and at most 1 customer. Non- identifying feature is used because Debit card number is assumed to be unique as it is created for individual associated with an account based on unique algorithm (card validation method).

5. ATM vs Withdrawal
1 to many relationship is used here as each ATM can have no withdrawals for maintenance and many withdrawals, each withdrawal will only happen at least at one ATM and at most one ATM for concurrency control. WithdrawalID is assumed to be relied on ATM as WithdrawalNo can be repeated using different ATMs so I choose identifying 1 to many relationship.

6. DebitCard vs Withdrawal
1 to many relationship is used here as each debit card can have no withdrawal or many withdrawal, as if I have enough cash , or I need to withdrawal several time for spending more money. Each withdrawal must belong to at least one card and at most one card. Non-identifying feature is used as withdrawalID with ATMID is assumed to be unique and also for security issue.

7. LoanType vs Loan
Each loan has at least 1 type and at most 1 type and each loanType can have no loans at the moment in the bank or many loan owing by customers at the same. LoanID is assumed to be unique so it uses non-identifying 1 to many relationship.

8. Loan vs Account
Each loan must belong to at least 1 account and at most 1 account. Each account can have no loans or many loans (student loan, car loan….) at the same time. Non- identifying 1 to many relationship is used as loanID is unique.

9. Repayment vs Loan
Each loan can have no repayments at the moment due to financial hardship, or many repayments. Each repayment must be linked to at least 1 loan and at most 1 loan. Identifying 1 to many relationship is used as RepaymentNo is based on Loan (different loan can have same repaymentNo).

10. Repayment vs Employee
1 to many relationship as each repayment is overlooked by at least 1 employee and at most 1 employee, whereas each employee can monitor no repayments or many repayments. Non- identifying 1 to many relationship is used as RepaymentNo(with Loan_LoanID) is unique.

11. **Employee vs Branch vs Department (for both managers and employee)**
Many to many relationship as each employee is assumed to work at least one branch and many branches because employee can be causal workers, whereas each branch can at have no employees(plan for new branch), or many employees. I assume managers register through managementRecord and normal employees register through employmentRecord.

- **Employee vs EmploymentRecord**
  Can have many employees (teller, loan assistant….) but also can be optional because some employees are managers, or new employees who haven't assigned a department at a particular branch.
- **Department vs EmploymentRecord**
  EmploymentRecord are assumed to match at at least 1 department and at most 1 department (employees might not be able to work at different branch at the same time, so if employee wants to change branch he might require to register another employment record), whereas each departments can have no employmentRecord, for example new department created.
- **Branch vs EmploymentRecord**
  Employment record must at least match a branch and at most a branch . Each branch can have no employmentRecord (new branch)or many employmentRecord.
- **Employee vs ManagementRecord**
  Not all employees are managers so mandatory is not used here. A manager can have many managementRecord due to branch transfer, and each record is for at least 1 and at most 1 manager. So 1 to many.
- **Department vs manages**
  Each managementRecord must manage exactly one department at particular branch and each department can have no records (new department) or many records.
- **Branch vs ManagementRecord**
  Each managementRecord must match at least one branch and at most 1 branch, as I assume manager who registered a record can only manage a place(1 department and 1 branch). One branch can have no record or many records (new branch).

12. **Branch vs Account**
Each branch can have no accounts (new branch)or many accounts, where as each account must be created exactly at one Branch. Account can be repeated at different branches(small chances) therefore account require the primary key of branch so identifying 1 to many relationship is needed for uniqueness.

Task 4:

JOINT ACCOUNT is assumed to be an account type.

Task 5:

1. I assume that we just need the number of JOINT accounts (not customer registration) so I make a join between the two entities 'Account' and 'AccountType'. Group the 'JOINT' account to show the number of accounts that are 'JOINT'.

2.I assume we need to find the account number and corresponding branchBSB for the Personal loan, since I assume I don't know what the loanTypeID for personal loan, so I need to use subquery to find the loanTypeID and pass it the where statement in my outer query. Join the Account and Loan and filter for personal loan based on the ID passed from subquery.

3. Assume I need the name, departmentName, branch name for each manager, so I join employee (to get the name of the manager), managementRecord (to get the ID for management), department (to get the department name),and Branch(to get the name of the branch).

4.the innermost subquery gives the ID of all managers, and pass them to first inner query to calculate the average and use it as an indicator to determine which managers have salary greater than the average. And print out the name.