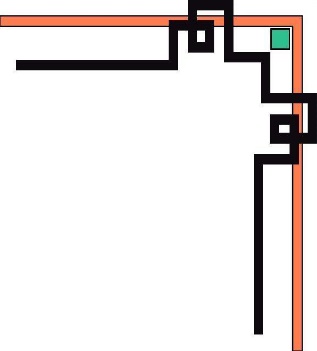
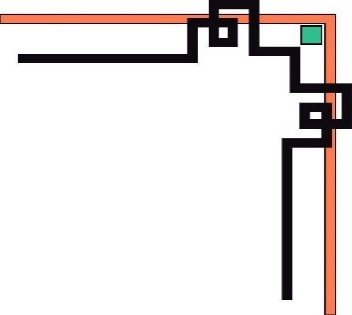
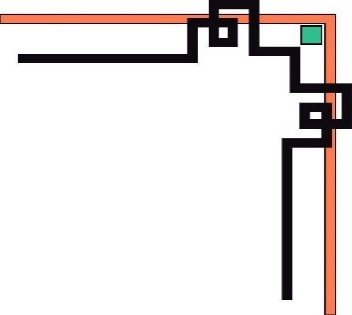
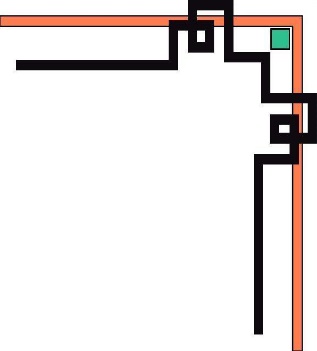
**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**



**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO ASSIGNMENT 2**

GVHD

**Phan Trọng Nhân**

Nguyễn Ngọc Tân 1813942

Nguyễn Duy Thanh 1813967

Nguyễn Quang Tùng 1814701

**TP.HCM, 19/12/2020**

[**PHẦN I: THIẾT KẾ CƠ SỞ DỮ LIỆU VẬT LÝ** 3](#_Toc59237514)

[**A- Triển khai cơ sở dữ liệu** 3](#_Toc59237515)

[**Lược đồ cơ sở dữ liệu quan hệ của ngân hàng** 3](#_Toc59237516)

[**Tạo bảng trên oracle live sql** 4](#_Toc59237517)

[**B. CHÈN DỮ LIỆU** 10](#_Toc59237518)

[**PART 2: STORE PROCEDURE / FUNCTION / SQL** 14](#_Toc59237519)

[a/ 14](#_Toc59237520)

[b/ 15](#_Toc59237521)

[c/ 16](#_Toc59237522)

[d/ 17](#_Toc59237523)

[**PART 3: BUILDING APPLICATIONS** 17](#_Toc59237524)

# **PHẦN I: THIẾT KẾ CƠ SỞ DỮ LIỆU VẬT LÝ**

## **A- Triển khai cơ sở dữ liệu**

**Yêu cầu:**

* Bạn phải triển khai cơ sở dữ liệu của mình, dựa trên chủ đề được chỉ định, vào

cơ sở dữ liệu.

* Đưa ra lời giải thích đầy đủ về các lựa chọn của bạn về kiểu dữ liệu, độ dài dữ liệu và ràng buộc trong cơ sở dữ liệu của bạn.

### **Lược đồ cơ sở dữ liệu quan hệ của ngân hàng**

Branch(BraName,Email,No,Street,District,City,Region,Empcode)

Bra\_Phone(BraName,BraPhone)

Branch\_Fax(BraName,Fax)

Employee(EmpCode,no,street,district,city,dob,email,Lname,Fname,BraName)

Emp\_Phone(EmpCode,EmpPhone)

Customer(CusCode,Fname,Lname,HomeAddr,Email,OffAddr,EmpCode)

Cus\_Phone(CusCode,CusPhone)

Account(Number,CusCode,AccCode)

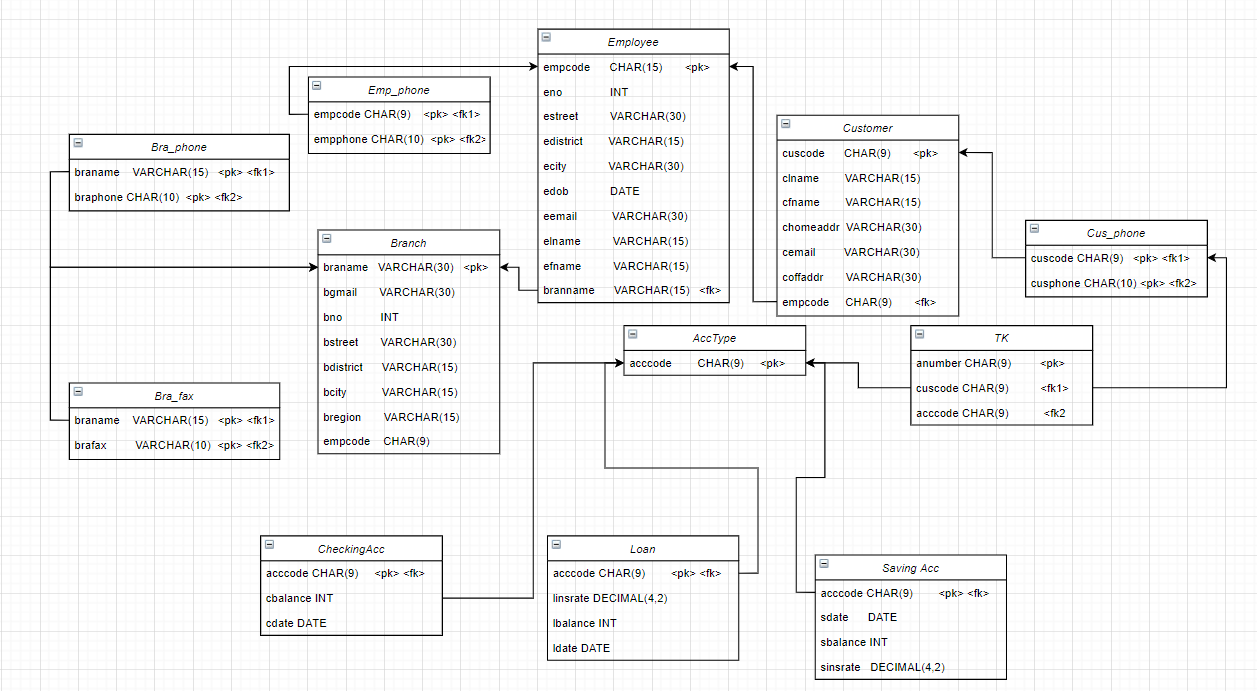
AccType(AccCode)

SavingAcc(AccCode,Date,Balance,InsRate)

CheckingAcc(AccCode,Balance,Date)

Loan(AccCode,InsRate,Balance,Date)

### **Tạo bảng trên oracle live sql**



/\*

Dưới đây là 4 kiểu dữ liệu nhóm em sử dụng trong việc tạo bảng:

1/ Cú pháp kiểu dữ liệu VARCHAR(kich\_thuoc) trong đó:

- kich\_thuoc là số kí tư lưu trữ

- Có độ dài tùy biến

- Và không chứa kí tự unicode

Ở đây em xét trường hợp dữ liệu nhập vào không chứa unicode, nếu muốn nhập dữ liệu có chứa unicode thì chúng ta sẽ sử dụng NVARCHAR(kich\_thuoc)

2/ Cú pháp kiểu dữ liệu INT để lưu số nguyên có kích thước từ -2,147,483,648 đến 2,147,483,647

3/ Cú pháp kiểu dữ liệu CHAR(kich\_thuoc) trong đó:

- kich\_thuoc là số kí tự lưu trữ.

- Độ dài cố định.

- Thêm dấu cách về bên phải để bù phần trống cho đủ số kí tự.

- Không chứa kí tự Unicode.

Tương tự thì ở đây em xét trường hợp dữ liệu nhập vào không chứa unicode, nếu muốn nhập dữ liệu có chứa unicode thì chúng ta sẽ sử dụng NCHAR(kich\_thuoc)

4/ Cú pháp kiểu dữ liệu DATE có giá trị từ '01-01-0001' đến '31-12-9999' được định nghĩa dưới dạng 'DD-MM-YY'

Dựa trên các tính chất của các kiểu dữ liệu mà nhóm em lựa chọn những kiểu phù hợp với hệ cơ sở dữ liệu:

- Đối với kích thước trong VARCHAR nhóm em lựa chọn VARCHAR(15) và VARCHAR(30) trong đó VARCHAR(15) sử dụng trong để đặt tên và VARCHAR(30) sử dụng trong gmail và địa chỉ vì chúng có độ dài không có định và cần nhiều kí tự hơn

- Kiểu dữ liệu DATE sử dụng liên quan đến ngày

- Kiểu dữ liệu INT sử dụng để lưu số nhà trong phần địa chỉ

- Đối với kích thước trong CHAR nhóm em lựa chọn CHAR(9) và CHAR(10) có độ dài không đổi và bằng với yêu cầu để không lãng phí dữ liệu trong đó CHAR(9) sử dụng lưu mã số của chi nhánh, nhân viên, khách hàng và tài khoản có 9 kí tự. CHAR(10) sử dụng lưu số điện thoại, fax gồm 10 kí tự

5/ DECIMAL(4,2) để lưu insrate có giá trị từ 00.00 – 99.99

\*/

CREATE TABLE Branch

( braname VARCHAR(15) not null ,

bgmail VARCHAR(30) ,

bno INT not null ,

bstreet VARCHAR(30) not null ,

bdistrict VARCHAR(15) ,

bcity VARCHAR(15) not null ,

bregion VARCHAR(15) not null ,

empcode CHAR(9) not null ,

PRIMARY KEY (braname)

-- chi nhánh cần có địa chỉ cụ thể nên không được bỏ trống

-- mỗi chi nhánh đều có nhân viên quản lí nên giá trị empcode là not null

-- mỗi chi nhánh có tên duy nhất và không null nên được chọn làm khóa chính

);

CREATE TABLE Employee

( empcode CHAR(9) not null ,

eno INT not null ,

estreet VARCHAR(30) not null ,

edistrict VARCHAR(15) ,

ecity VARCHAR(15) not null,

edob DATE not null ,

eemail VARCHAR(30) ,

elname VARCHAR(15) not null,

efname VARCHAR(15) not null,

braname VARCHAR(15) not null,

PRIMARY KEY (empcode),

FOREIGN KEY (braname) REFERENCES Branch(braname)

-- nhân viên bắt buộc phải lưu họ tên, địa chỉ và chi nhánh làm việc

-- mỗi nhân viên có một mã duy nhất và không null nên chọn empcode làm khóa chính

);

CREATE TABLE Customer

( cuscode CHAR(9) not null ,

clname VARCHAR(15) not null ,

cfname VARCHAR(15) not null ,

chomeaddr VARCHAR(30) not null ,

cemail VARCHAR(30) ,

coffaddr VARCHAR(30) ,

empcode CHAR(9) not null ,

PRIMARY KEY (cuscode),

FOREIGN KEY (empcode) REFERENCES Employee(empcode)

-- khách hàng bắt buộc phải lưu họ tên, địa chỉ nhà

-- mỗi khách hàng có một mã duy nhất và không null nên chọn empcode làm khóa chính

);

CREATE TABLE AccType

( acccode CHAR(9) not null,

PRIMARY KEY (acccode)

);

CREATE TABLE TK

( anumber CHAR(9) not null ,

cuscode CHAR(9) not null ,

acccode CHAR(9) not null ,

PRIMARY KEY (anumber),

FOREIGN KEY (cuscode) REFERENCES Customer(cuscode),

FOREIGN KEY (acccode) REFERENCES AccType(acccode)

-- mỗi tài khoản có một số tài khoản duy nhất và không null nên được chọn làm khóa chính

-- cuscode và acccode là 2 khóa ngoại

);

CREATE TABLE SavingAcc

( acccode CHAR(9) not null,

sdate DATE not null ,

sbalance INT not null ,

sinsrate DECIMAL(4,2) not null ,

PRIMARY KEY (acccode),

FOREIGN KEY (acccode) REFERENCES AccType(acccode)

-- mọi thông tin trong SavingAcc đều là thông tin bắt buộc phải có

);

CREATE TABLE CheckingAcc

( acccode CHAR(9) not null ,

cbalance INT not null ,

cdate DATE not null ,

PRIMARY KEY (acccode),

FOREIGN KEY (acccode) REFERENCES AccType(acccode)

-- mọi thông tin trong CheckingAcc đều là thông tin bắt buộc phải có

);

CREATE TABLE Loan

( acccode CHAR(9) not null,

linsrate DECIMAL(4,2) not null ,

lbalance INT not null ,

ldate DATE not null ,

PRIMARY KEY (acccode),

FOREIGN KEY (acccode) REFERENCES AccType(acccode)

-- mọi thông tin trong Loan đều là thông tin bắt buộc phải có

);

-- Dưới đây là thông tin cách thức liên lạc nên tất cả không được null

CREATE TABLE Bra\_Phone

( braname VARCHAR(15) not null,

braphone CHAR(10) not null,

PRIMARY KEY (braname,braphone),

FOREIGN KEY (braname) REFERENCES Branch(braname)

);

CREATE TABLE Bra\_Fax

( braname VARCHAR(15) not null,

brafax CHAR(10) not null,

PRIMARY KEY (braname,brafax),

FOREIGN KEY (braname) REFERENCES Branch(braname)

);

CREATE TABLE Emp\_Phone

( empcode CHAR(9) not null,

empphone CHAR(10) not null,

PRIMARY KEY (empcode,empphone),

FOREIGN KEY (empcode) REFERENCES Employee(empcode)

);

CREATE TABLE Cus\_Phone

( cuscode CHAR(9) not null,

cusphone CHAR(10) not null ,

PRIMARY KEY (cuscode,cusphone),

FOREIGN KEY (cuscode) REFERENCES Customer(cuscode)

);

## **B. CHÈN DỮ LIỆU**

ALTER SESSION SET NLS\_DATE\_FORMAT = 'DD-MM-YYYY';

INSERT INTO Branch VALUES ('OCB', 'ocb@gmail.com' , '11', 'Hai Ba Trung', 'Quan 1' , 'Ho Chi Minh' , 'Mien nam' , '000000001');

INSERT INTO Branch VALUES ('BIDV', 'bidv@gmail.com' , '22', 'Luong Thanh', 'Quan 2' , 'Ho Chi Minh' , 'Mien nam' , '000000002');

INSERT INTO Branch VALUES ('AGRIBANK', 'agribank@gmail.com' , '33', 'Mac Dinh Chi', 'Quan 3' , 'Ho Chi Minh' , 'Mien nam' , '000000003');

INSERT INTO Branch VALUES ('ACB', 'acb@gmail.com' , '44', 'Pham Ngu Lao', 'Quan 4' , 'Ho Chi Minh' , 'Mien nam' , '000000004');

INSERT INTO Employee VALUES ('100000000', '10', 'Bui Vien', 'Quan 1', 'Ho Chi Minh' , '09-01-1965', 'vantan@gmail.com', 'Nguyen Van','Tan','OCB');

INSERT INTO Employee VALUES ('100000001', '20', 'Ly Thuong Kiet', 'Quan 2', 'Ho Chi Minh' , '10-02-1964', 'ngoctan@gmail.com', 'Nguyen Ngoc','Tan','BIDV');

INSERT INTO Employee VALUES ('100000002', '30', 'Hung Vuong', 'Quan 3', 'Ho Chi Minh' , '11-03-1963', 'quangtung@gmail.com', 'Nguyen Quang','Tung','AGRIBANK');

INSERT INTO Employee VALUES ('100000003', '40', 'Cao Ba Quat', 'Quan 4', 'Ho Chi Minh' , '12-04-1962', 'duythanh@gmail.com', 'Nguyen Duy','Thanh','ACB');

INSERT INTO Customer VALUES ('200000000', 'Nguyen Van' , 'Thang' , '20 Cao Bang' , 'vanthang@gmail.com' , '02 Vinh Xuan' , '100000000');

INSERT INTO Customer VALUES ('200000001', 'Tran Hoang' , 'Viet' , '20 Cao Bang' , 'hoangviet@gmail.com' , '03 Dong Du' , '100000001');

INSERT INTO Customer VALUES ('200000002', 'Hoang Quynh' , 'Anh' , '20 Cao Bang' , 'quynhanh@gmail.com' , '04 Dong Khoi' , '100000002');

INSERT INTO Customer VALUES ('200000003', 'Tran Yen' , 'Nhu' , '20 Cao Bang' , 'yennhu@gmail.com' , '05 Le Dung' , '100000003');

INSERT INTO AccType VALUES ('300000000');

INSERT INTO AccType VALUES ('300000001');

INSERT INTO AccType VALUES ('300000002');

INSERT INTO AccType VALUES ('300000003');

INSERT INTO AccType VALUES ('300000004');

INSERT INTO AccType VALUES ('300000005');

INSERT INTO AccType VALUES ('300000006');

INSERT INTO AccType VALUES ('300000007');

INSERT INTO AccType VALUES ('300000008');

INSERT INTO AccType VALUES ('300000009');

INSERT INTO AccType VALUES ('300000010');

INSERT INTO AccType VALUES ('300000011');

INSERT INTO TK VALUES ('400000000','200000000','300000000');

INSERT INTO TK VALUES ('400000001','200000001','300000001');

INSERT INTO TK VALUES ('400000002','200000002','300000002');

INSERT INTO TK VALUES ('400000003','200000003','300000003');

INSERT INTO TK VALUES ('400000004','200000000','300000004');

INSERT INTO TK VALUES ('400000005','200000001','300000005');

INSERT INTO TK VALUES ('400000006','200000002','300000006');

INSERT INTO TK VALUES ('400000007','200000003','300000007');

INSERT INTO TK VALUES ('400000008','200000000','300000008');

INSERT INTO TK VALUES ('400000009','200000001','300000009');

INSERT INTO TK VALUES ('400000010','200000002','300000010');

INSERT INTO TK VALUES ('400000011','200000003','300000011');

INSERT INTO SavingAcc VALUES ('300000000', '05-12-2019' ,'25000000','06.50');

INSERT INTO SavingAcc VALUES ('300000001', '06-11-2019' ,'45000000','06.50');

INSERT INTO SavingAcc VALUES ('300000002', '07-10-2020' ,'125000000','06.50');

INSERT INTO SavingAcc VALUES ('300000003', '08-09-2020' ,'205000000','06.50');

INSERT INTO CheckingAcc VALUES ('300000004','100000000', '05-12-2019');

INSERT INTO CheckingAcc VALUES ('300000005','200000000', '06-11-2019');

INSERT INTO CheckingAcc VALUES ('300000006','300000000', '07-10-2020');

INSERT INTO CheckingAcc VALUES ('300000007','400000000', '08-09-2020');

INSERT INTO Loan VALUES ('300000008', '05.00' ,'12000000', '05-12-2019');

INSERT INTO Loan VALUES ('300000009', '05.00' ,'24000000', '06-11-2019');

INSERT INTO Loan VALUES ('300000010', '05.00' ,'36000000', '07-10-2020');

INSERT INTO Loan VALUES ('300000011', '05.00' ,'48000000', '08-09-2020');

INSERT INTO Bra\_Phone VALUES ('OCB', '0848807872');

INSERT INTO Bra\_Phone VALUES ('OCB', '0848807873');

INSERT INTO Bra\_Phone VALUES ('BIDV', '0397755014');

INSERT INTO Bra\_Phone VALUES ('BIDV', '0397755016');

INSERT INTO Bra\_Phone VALUES ('AGRIBANK', '0395126195');

INSERT INTO Bra\_Phone VALUES ('AGRIBANK', '0395126196');

INSERT INTO Bra\_Phone VALUES ('ACB', '0384054105');

INSERT INTO Bra\_Phone VALUES ('ACB', '0384054106');

INSERT INTO Bra\_Fax VALUES ('OCB', '1111111111');

INSERT INTO Bra\_Fax VALUES ('OCB', '2222222222');

INSERT INTO Bra\_Fax VALUES ('BIDV', '3333333333');

INSERT INTO Bra\_Fax VALUES ('BIDV', '4444444444');

INSERT INTO Bra\_Fax VALUES ('AGRIBANK', '5555555555');

INSERT INTO Bra\_Fax VALUES ('AGRIBANK', '6666666666');

INSERT INTO Bra\_Fax VALUES ('ACB', '7777777777');

INSERT INTO Bra\_Fax VALUES ('ACB', '8888888888');

INSERT INTO Emp\_Phone VALUES ('100000000', '0123456789');

INSERT INTO Emp\_Phone VALUES ('100000001', '0123456788');

INSERT INTO Emp\_Phone VALUES ('100000002', '0123456787');

INSERT INTO Emp\_Phone VALUES ('100000003', '0123456786');

INSERT INTO Emp\_Phone VALUES ('100000000', '0123456785');

INSERT INTO Emp\_Phone VALUES ('100000001', '0123456784');

INSERT INTO Emp\_Phone VALUES ('100000002', '0123456783');

INSERT INTO Emp\_Phone VALUES ('100000003', '0123456782');

INSERT INTO Cus\_Phone VALUES ('200000000', '9876543210');

INSERT INTO Cus\_Phone VALUES ('200000001', '9876543211');

INSERT INTO Cus\_Phone VALUES ('200000002', '9876543212');

INSERT INTO Cus\_Phone VALUES ('200000003', '9876543213');

INSERT INTO Cus\_Phone VALUES ('200000000', '9876543220');

INSERT INTO Cus\_Phone VALUES ('200000001', '9876543221');

INSERT INTO Cus\_Phone VALUES ('200000002', '9876543222');

INSERT INTO Cus\_Phone VALUES ('200000003', '9876543223');

# **PART 2: STORE PROCEDURE / FUNCTION / SQL**

Bank Database

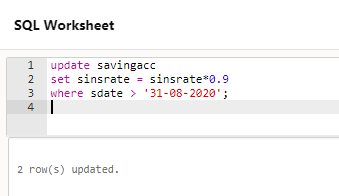
## a/

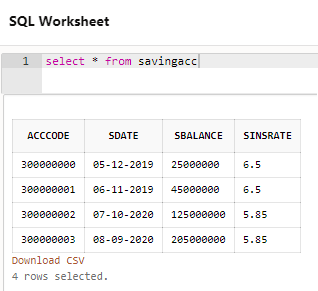
**Decrease interest rate to 10% for all saving accounts whose opening date is from 01/09/2020.**

update savingacc

set sinsrate = sinsrate\*0.9

where sdate > '31-08-2020';





## b/

**Select all the account information of the customer named ‘Nguyen Van Thang’.**

SELECT cuscode, clname, cfname, chomeaddr, cemail, coffaddr

from customer

where clname = 'Nguyen Van' and cfname = 'Thang';

select Customer.cuscode, Customer.clname, Customer.cfname, Cus\_Phone.cusphone

from (Customer

INNER JOIN Cus\_Phone on Customer.cuscode = Cus\_Phone.cuscode )

where Customer.clname = 'Nguyen Van' and Customer.cfname = 'Thang';

select Customer.cuscode, Customer.clname, Customer.cfname, TK.anumber, SavingAcc.sdate, SavingAcc.sbalance, SavingAcc.sinsrate

from ((Customer INNER JOIN TK on Customer.cuscode = TK.cuscode ) INNER JOIN SavingAcc on TK.acccode = SavingAcc.acccode)

where Customer.clname = 'Nguyen Van' and Customer.cfname = 'Thang';

select Customer.cuscode, Customer.clname, Customer.cfname, TK.anumber, CheckingAcc.cbalance, CheckingAcc.cdate

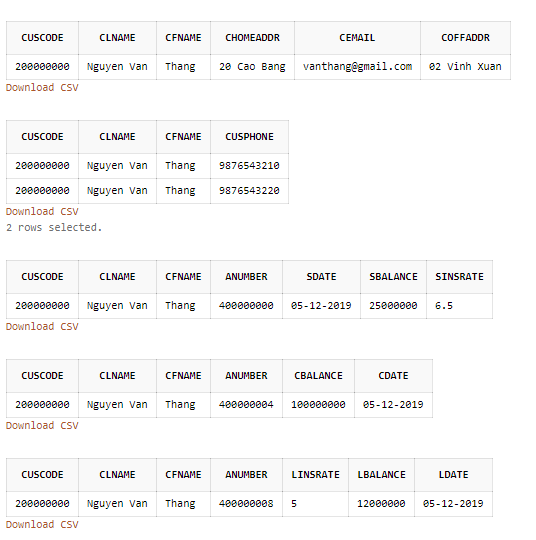
from ((Customer INNER JOIN TK on Customer.cuscode = TK.cuscode ) INNER JOIN CheckingAcc on TK.acccode = CheckingAcc.acccode)

where Customer.clname = 'Nguyen Van' and Customer.cfname = 'Thang';

select Customer.cuscode, Customer.clname, Customer.cfname, TK.anumber, Loan.linsrate, Loan.lbalance, Loan.ldate

from ((Customer INNER JOIN TK on Customer.cuscode = TK.cuscode ) INNER JOIN Loan on TK.acccode = Loan.acccode)

where Customer.clname = 'Nguyen Van' and Customer.cfname = 'Thang';



## c/

**Write a function to calculate the total balance for each account type of a customer. Input: Customer ID Output: A list of balance per account type**

**Tạo type TABLE\_BALANCE\_TYPE:**

create or replace TYPE table\_balance\_type IS TABLE OF NUMBER(38,0);



**Tạo Hàm:**

create or replace FUNCTION Total\_Balance (custom\_id IN CHAR)

RETURN TABLE\_BALANCE\_TYPE

AS result\_fun TABLE\_BALANCE\_TYPE := TABLE\_BALANCE\_TYPE();

BEGIN

result\_fun.extend(3);

SELECT SUM(A.SBALANCE) INTO result\_fun(1)

FROM SAVINGACC A, TK D

WHERE D.CUSCODE = custom\_id AND D.ACCCODE = A.ACCCODE;

SELECT SUM(B.CBALANCE) INTO result\_fun(2)

FROM CHECKINGACC B, TK D

WHERE D.CUSCODE = custom\_id AND D.ACCCODE = B.ACCCODE;

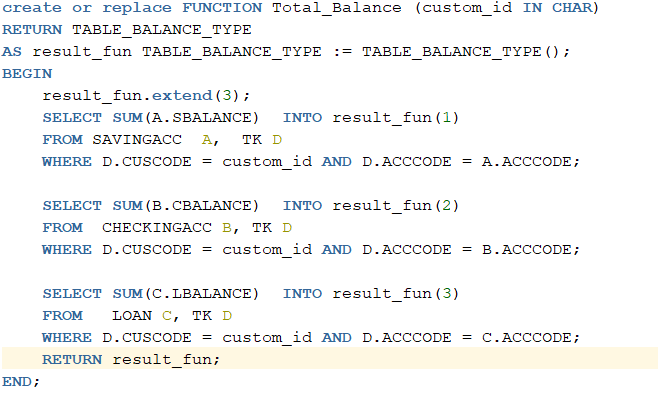
SELECT SUM(C.LBALANCE) INTO result\_fun(3)

FROM LOAN C, TK D

WHERE D.CUSCODE = custom\_id AND D.ACCCODE = C.ACCCODE;

RETURN result\_fun;

END;



Kết quả:

## 

Giải thích: ở đây khi truyền giá trị mã của khách hàng, hàm sẽ trả về 1 bảng với hàng thứ nhất là tổng balance của tài khoản loại saving, hàng thứ hai là tổng balance của tài khoản loại checking, và cuối cùng là của loans.

## d/

**Write a procedure to sort the employee in decreasing number of customers he/she takes care in a period of time .Input: Start date, End date Output: A list of sorting customers.**

CREATE OR REPLACE PROCEDURE sort\_employee(bdate DATE,edate DATE)

IS

BEGIN

DBMS\_OUTPUT.put\_line ('Start date = ' || bdate ||' '||'End date = '|| edate);

FOR rec IN(

SELECT E.EMPCODE as G, COUNT(C.CUSCODE) AS H

FROM EMPLOYEE E,CUSTOMER C

WHERE (E.EMPCODE = C.EMPCODE) AND C.CUSCODE IN

(

SELECT DISTINCT D.CUSCODE

FROM CUSTOMER D,TK T,SAVINGACC S,CHECKINGACC F,LOAN L

WHERE (D.CUSCODE = T.CUSCODE AND (T.ACCCODE = S.ACCCODE) AND

((S.SDATE > bdate) AND (S.SDATE < edate)))

OR (D.CUSCODE = T.CUSCODE AND (T.ACCCODE = F.ACCCODE) AND

((F.CDATE > bdate) AND (F.CDATE < edate)))

OR (D.CUSCODE = T.CUSCODE AND (T.ACCCODE = L.ACCCODE) AND

((L.LDATE > bdate) AND (L.LDATE < edate)))

)

GROUP BY E.EMPCODE

ORDER BY COUNT(C.CUSCODE))

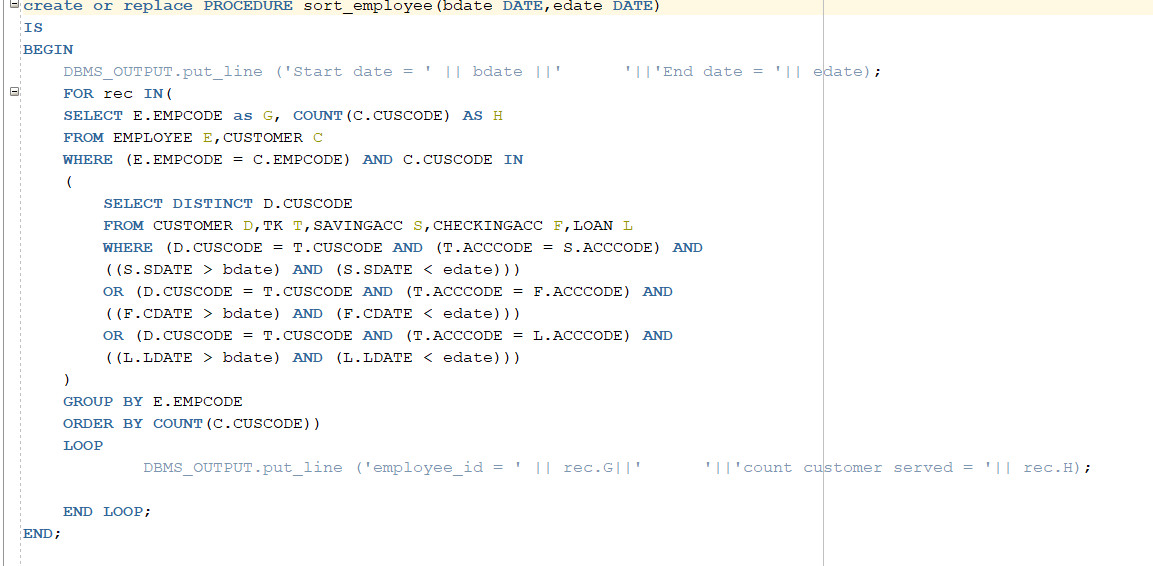
LOOP

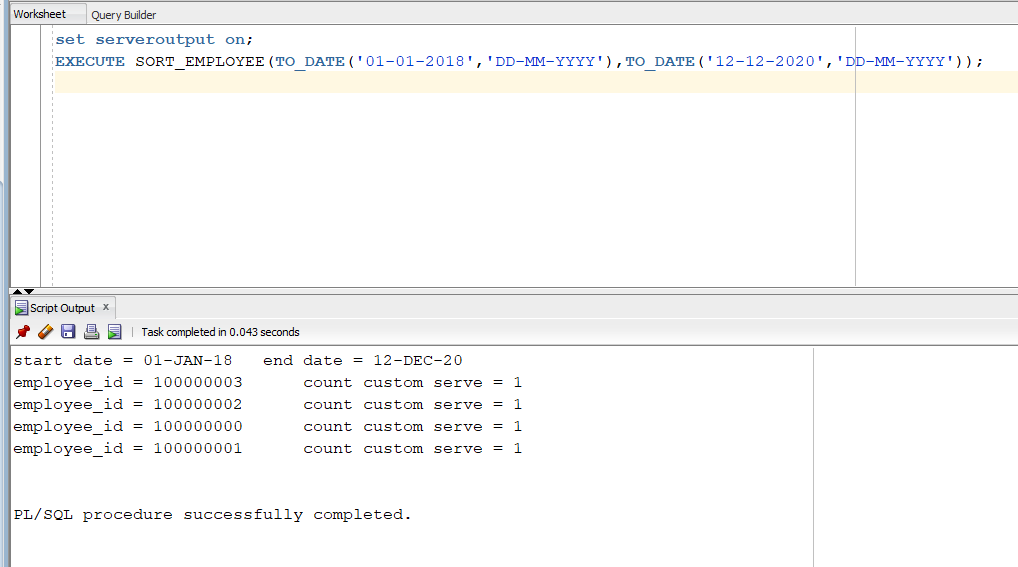
DBMS\_OUTPUT.put\_line ('employee\_id = ' || rec.G||' '||'count customer served = '|| rec.H);

END LOOP;

END;

Kết quả:





ở đây trong thân của procedure, nhóm em chỉ in list các id của nhân viên cùng với số lượng khách hàng phục vụ giảm dần. Nếu cần thiết trong procedure tụi em vẫn có thể lưu trữ bảng giá trị này vào Curson.

# **PART 3: BUILDING APPLICATIONS**

Tụi em vẫn đang trong giai đoạn hoàn thành các chức năng theo yêu cầu của bài tập lớn và sẽ cố gắng hoàn thành và demo trong buổi báo cáo. Mong thầy châm chước cho nhóm em.