**Monthly bill:**

bill(accountNo in number, bilDate in date, electricityUsage in number,  gasUsage in number, billGenerationDate in date)

*Algorithm:*

*Compute the total charges for the month.*

*Update ‘account’ table with account balance which is computed in previous stage.*

*Update the ‘bill’ table with usage information for the month.*

*Compute the due date for the bill and record it in ‘monthly\_bill’ table.*

*Case 1: Normal case: Customer does not end or start the service to the house in current month.*

*Value of total charge = previous balance + electricity usage \* unit charge for electricity + gas usage \* gas unit charge + flat fee*

*Case 2: There are two customers that have occupied the house in the month*

*Number\_of\_days\_old = days between account start date and account end date.*

*total\_days = Total number of days in the month*

*Number\_of\_days\_new = total\_days - Number\_of\_days\_old*

*Value of total charge for old customer =   previous balance +*

*(Number\_of\_days/ total\_days)  \* (electricity usage \* unit charge for electricity + gas usage \* gas unit charge) +*

*flat fee*

*Value of total charge for new customer =   previous balance +*

*(Number\_of\_days\_new / total\_days)  \* (electricity usage \* unit charge for electricity + gas usage \* gas unit charge) +*

*flat fee*

*Update account status to inactive for the old customer.*

*Display the bill details to the customer.*

Modified Algorithm:

Case 1: For the account, which is billed. If ((acc\_end\_date is null OR acc\_end\_date > bill end date) AND acc\_start\_date <= bill\_astart\_date)

Normal case: Customer does not end or start the service to the house in current month.

        Value of total charge = previous balance + electricity usage \* unit charge for electricity + gas usage \* gas unit charge + flat fee

Case 2: For the account, which is billed. If acc\_end\_date is NOT NULL AND acc\_end\_date < bill end date)

        If (acc\_start\_date > bill\_start\_date)

                    Billed\_days = days\_between(acc\_start\_date ,acc\_end\_date)

                    Else

                                Billed\_days = days\_between(bill\_start\_date ,acc\_end\_date)

        total\_days = Total number of days in the month

        Value of total\_charge =   previous balance +

(Billed\_days / total\_days)  \* (electricity usage \* unit charge for electricity + gas usage \* gas unit charge) +

flat fee

Update account status to inactive for the old customer

Display the bill details to the customer.

Update account balance as total\_charge.

Make previous  balance to 0.

*Input:* 1) accountNo

                          2) bilDate

                          3) electricityUsage

                          4) gasUsage

           5) billGenerationDate

*Output:*

Print the customer name, address, account number, previous balance, electricity usage and charges, gas usage and charges and total charges.

show errors;

CREATE OR REPLACE PROCEDURE billGeneration (accountNo in account.aid%type, billMonthYr IN date, electricityUsage in number, gasUsage in number, billGenerationDate in date) IS

total\_charge number;

Total\_Days Integer;

v\_count number := 0; /\*Used in display to identify the difference in billing days.\*/

v\_status account.status%type;

v\_end\_date account.end\_date%type;

v\_start\_date account.start\_date%type;

last\_day\_of\_month date;

--v\_month number;

first\_day\_of\_month date;

v\_previous\_balance monthly\_bill.previous\_balance%type;

billed\_days number;

v\_electricity\_rate rate.electricity\_rate%type;

v\_gas\_rate rate.gas\_rate%type;

v\_flat\_fee rate.flat\_fee%type;

v\_due\_date date;

v\_cname customer.cname%type;

v\_street house\_address.street%type;

v\_city house\_address.city%type;

v\_zip house\_address.zip%type;

v\_rid rate.rid%type;

v\_account\_balance account.account\_balance%type;

BEGIN

SELECT UPPER(status),end\_date,start\_date,account\_balance

INTO v\_status,v\_end\_date,v\_start\_date,v\_account\_balance

FROM account

WHERE aid = accountNo;

/\*Check if account exists and service is started for the account.\*/

IF (v\_status != 'ACTIVE') THEN

dbms\_output.put\_line('Service on house with Account No. ' ||accountNo|| ' is not active.');

GOTO end\_of\_procedure;

END IF;

/\*Get first and last day of the billed month\*/

SELECT ADD\_MONTHS((LAST\_DAY(billMonthYr)+1),-1), LAST\_DAY(billMonthYr)

INTO first\_day\_of\_month , last\_day\_of\_month

FROM DUAL;

SELECT trunc(last\_day\_of\_month - first\_day\_of\_month + 1)

INTO total\_days

FROM dual;

/\*Make previous\_balance as 0 if account has started service in the current month\*/

--SELECT EXTRACT(MONTH FROM billMonthYr) into v\_month FROM DUAL;

IF (v\_start\_date < first\_day\_of\_month) THEN

/\*Consider December month of previous year to compute previous\_balance.\*/

/\*IF (v\_month = 1) THEN

SELECT previous\_balance

INTO v\_previous\_balance

FROM monthly\_bill

WHERE aid = accountNo and EXTRACT(MONTH FROM bill\_date) = EXTRACT(MONTH FROM billMonthYr) - 1 AND

EXTRACT(YEAR FROM bill\_date) = EXTRACT(YEAR FROM billMonthYr)-1 ;

ELSE\*/

/\*Consider previous month of the same year to compute previous\_balance.\*/

SELECT previous\_balance

INTO v\_previous\_balance

FROM monthly\_bill

WHERE aid = accountNo and EXTRACT(MONTH FROM bill\_date) = EXTRACT(MONTH FROM billMonthYr) AND

EXTRACT(YEAR FROM bill\_date) = EXTRACT(YEAR FROM billMonthYr) ;

/\*END IF; \*/

ELSE

v\_previous\_balance := 0;

END IF;

SELECT electricity\_rate ,gas\_rate , flat\_fee, rid

INTO v\_electricity\_rate ,v\_gas\_rate , v\_flat\_fee,v\_rid

FROM rate

WHERE EXTRACT(MONTH FROM rate\_date) = EXTRACT(MONTH FROM billMonthYr) AND

EXTRACT(YEAR FROM rate\_date) = EXTRACT(YEAR FROM billMonthYr) ;

/\*Normal Case: When service is not started or ended in the current month. If(no end date specified OR end date is not in the current month) AND (start date is not more than first day of month)\*/

IF ((v\_end\_date is null OR v\_end\_date > last\_day\_of\_month) AND (v\_start\_date <= first\_day\_of\_month)) THEN

total\_charge := v\_previous\_balance + (electricityUsage \* v\_electricity\_rate) + (gasUsage \* v\_gas\_rate) + v\_flat\_fee;

v\_count := 1;

/\*When service is ending in the current month\*/

ELSIF((v\_end\_date IS NOT NULL) AND (v\_end\_date <= last\_day\_of\_month)) THEN

/\*update status of account ot inactive\*/

UPDATE account SET status = 'Inactive' WHERE aid = accountNo;

IF (v\_start\_date > first\_day\_of\_month) THEN

SELECT trunc(v\_end\_date - v\_start\_date)

INTO billed\_days

FROM account

WHERE aid = accountNo;

ELSIF (v\_start\_date <= first\_day\_of\_month) THEN

/\*\*/

SELECT trunc(v\_end\_date - first\_day\_of\_month)

INTO billed\_days

FROM dual;

END IF;

total\_charge := v\_previous\_balance + ((billed\_days+1) / total\_days) \* (electricityUsage \* v\_electricity\_rate + gasUsage \* v\_gas\_rate) + v\_flat\_fee;

ELSIF((v\_end\_date IS NOT NULL) AND (v\_end\_date > last\_day\_of\_month)) THEN

IF (v\_start\_date > first\_day\_of\_month) THEN

SELECT trunc(last\_day\_of\_month - v\_start\_date)

INTO billed\_days

FROM dual;

END IF;

total\_charge := v\_previous\_balance + ((billed\_days+1) / total\_days) \* (electricityUsage \* v\_electricity\_rate + gasUsage \* v\_gas\_rate) + v\_flat\_fee;

ELSIF((v\_end\_date IS NULL) AND (v\_start\_date > first\_day\_of\_month)) THEN

SELECT trunc(last\_day\_of\_month - v\_start\_date)

INTO billed\_days

FROM account

WHERE aid = accountNo;

total\_charge := v\_previous\_balance + ((billed\_days+1) / total\_days) \* (electricityUsage \* v\_electricity\_rate + gasUsage \* v\_gas\_rate) + v\_flat\_fee;

END IF;

/\*Calculating due date\*/

SELECT billGenerationDate + interval '15' day INTO v\_due\_date FROM DUAL;

/\*update account\_balance in account table as total\_charge + previous account balance.\*/

UPDATE account SET account\_balance = v\_account\_balance + total\_charge where aid = accountNo;

/\*Recording usage information and updating previous balance of account.\*/

INSERT INTO monthly\_bill values(seq\_monthly\_bill.nextVal,accountNo,v\_rid,billGenerationDate,(billed\_days+1) / total\_days \* electricityUsage,(billed\_days+1) / total\_days \* gasUsage,v\_account\_balance,v\_due\_date,total\_charge);

/\*Print out information for user\*/

SELECT cname INTO v\_cname FROM account a,customer c where a.cid = c.cid AND a.aid = accountNo;

SELECT street,city,zip INTO v\_street,v\_city,v\_zip FROM account a,house\_address h where a.aid = accountNo AND a.hid = h.hid;

dbms\_output.put\_line('Billing information:');

dbms\_output.put\_line('Customer Name: '|| v\_cname);

dbms\_output.put\_line('Address: '||v\_street ||', '||v\_city||', '||v\_zip);

dbms\_output.put\_line('Account No.: '||accountNo );

dbms\_output.put\_line('Previous Balance: '||'$'||v\_previous\_balance);

IF (v\_count = 1) THEN

dbms\_output.put\_line('Electricity charges for usage of '||electricityUsage ||'kWh is: $'|| (electricityUsage \* v\_electricity\_rate));

dbms\_output.put\_line('Gas charges for usage of '||gasUsage ||'kWh is: $'||

(gasUsage \* v\_gas\_rate));

ELSE

dbms\_output.put\_line('Electricity charges for usage of '||((billed\_days+1) / total\_days)\*electricityUsage ||'kWh is: $'||

((billed\_days+1) / total\_days)\*(electricityUsage \* v\_electricity\_rate));

dbms\_output.put\_line('Gas charges for usage of '||((billed\_days+1) / total\_days)\*gasUsage ||'kWh is: $'||

((billed\_days+1) / total\_days)\*(gasUsage \* v\_gas\_rate));

END IF;

dbms\_output.put\_line('Total charges for the month is: '|| total\_charge);

/\*Update message table\*/

INSERT INTO message VALUES(message\_id\_seq.nextval,accountNo,'Bill is ready.',systimestamp);

<<end\_of\_procedure>>

dbms\_output.put\_line('');

EXCEPTION

when no\_data\_found then dbms\_output.put\_line('Data for the speficied account doesnot exists');

when too\_many\_rows then dbms\_output.put\_line('Too many rows');

when others then dbms\_output.put\_line('Error in data.');

END;