

Draft solution

Practise 3 – Materialized views

Creation and updating of the materialized view with the use CREATE MATERIALIZED VIEW in ORACLE

CREATE MATERIALIZED VIEW

1.

```
create materialized view GROUPBYMonthYear
```

```
build immediate
```

```
as
```

```
SELECT DateMonth, DateYear, SUM(NumberOfCalls) as NumCalls, SUM(Price) as TotPrice
```

```
FROM FACTS F, TIMEDIM T
```

```
WHERE F.ID_time = T.ID_time
```

```
GROUP BY DateMonth, DateYear;
```

2.

```
create materialized view GROUPBYMonthPhoneRateType
```

```
build immediate
```

```
as
```

```
select DateMonth, f. id_phonerate, p.phoneratetype, SUM(NumberOfCalls) as NumCalls, SUM(Price)  
as TotPrice
```

```
from PHONERATE P, TIMEDIM T, FACTS F
```

```
where F.ID_time = T.ID_time and f.id_phonerate = p.id_phonerate and t.dateyear=2003
```

```
group by DateMonth, f.id_phonerate, p.phoneratetype
```

Update and management of materialized view with CREATE MATERIALIZED VIEW LOG – for students who practice with their own PC)

CREATE MATERIALIZED VIEW LOG

CREATE MATERIALIZED VIEW LOG ON FACTS

WITH SEQUENCE, ROWID (ID_time, id_phonerate, Id_location_receiver, Id_Location_Caller, NumberOfCalls, Price)

INCLUDING NEW VALUES;

CREATE MATERIALIZED VIEW LOG ON TIMEDIM

WITH SEQUENCE, ROWID (ID_time, DateMonth, DateYear)

INCLUDING NEW VALUES;

CREATE MATERIALIZED VIEW LOG ON phonerate

WITH SEQUENCE, ROWID (id_phonerate, phoneratetype)

INCLUDING NEW VALUES;

Update of MATERIALIZE VIEW.

STEP 2.3 Try to modify the FACTS table as follows:

insert into FACTS(id_time, id_phonerate, ID_location_Caller, ID_location_receiver, price, numberofcalls) values(8,1,558,752,40000,150)

Before

	DATEMONTH	DATEYEAR	NUMCALLS	TOTPRICE
1	20-JUL-03	2003	73216	272838200
2	20-OCT-04	2004	64800	241677000
3	20-OCT-03	2003	64800	241677000
4	20-JUL-04	2004	73200	272836600
5	20-AUG-03	2003	73200	272836600
6	20-AUG-04	2004	73200	272836600

After

	DATEMONTH	DATEYEAR	NUMCALLS	TOTPRICE
1	20-JUL-03	2003	73216	272838200
2	20-OCT-04	2004	64800	241677000
3	20-OCT-03	2003	64800	241677000
4	20-JUL-04	2004	73200	272836600
5	20-AUG-03	2003	73350	272876600
6	20-AUG-04	2004	73200	272836600

2.

Before

	DATEMONTH	ID_PHONERATE	PHONERATETYPE
1	20-JUL-03	5	CartaNatale
2	20-JUL-03	1	Notte
3	20-AUG-03	1	Notte
4	20-AUG-03	3	Mattino
5	20-OCT-03	3	Mattino
6	20-OCT-03	1	Notte
7	20-JUL-03	6	Festivi
8	20-AUG-03	2	Giorno
9	20-AUG-03	4	24 ore su 24
10	20-AUG-03	5	CartaNatale
11	20-JUL-03	3	Mattino
12	20-OCT-03	2	Giorno
13	20-OCT-03	4	24 ore su 24
14	20-OCT-03	5	CartaNatale
15	20-JUL-03	2	Giorno
16	20-JUL-03	4	24 ore su 24

Insert into FACTS(Id_time, ID_phoneRate, ID_location Caller, ID_location_Receiver, Price, NumberOfCalls) values(1,6,558,752,100,100)

After

	DATEMONTH	ID_PHONERATE	PHONERATETYPE
1	20-JUL-03	5	CartaNatale
2	20-JUL-03	1	Notte
3	20-AUG-03	1	Notte
4	20-AUG-03	3	Mattino
5	20-OCT-03	3	Mattino
6	20-OCT-03	1	Notte
7	20-AUG-03	2	Giorno
8	20-AUG-03	4	24 ore su 24
9	20-AUG-03	5	CartaNatale
10	20-JUL-03	3	Mattino
11	20-OCT-03	2	Giorno
12	20-OCT-03	4	24 ore su 24
13	20-OCT-03	5	CartaNatale
14	20-JUL-03	2	Giorno
15	20-JUL-03	4	24 ore su 24
16	20-JUL-03	6	Festivi

Update and management of views via Trigger

VM Tables

1. VM1

```
CREATE TABLE VM1 (  
DateMonth DATE CHECK (DateMonth IS NOT NULL),  
DateYear INTEGER CHECK (DateYear IS NOT NULL),  
TOT_NumberOfCalls INTEGER,  
TOT_Price INTEGER);  
  
INSERT INTO VM1 (DateMonth, DateYear, TOT_NumberOfCalls, TOT_Price)  
(SELECT DateMonth, DateYear, SUM(NumberOfCalls), SUM(Price)  
FROM FACTS F, TIMEDIM T  
WHERE F.ID_time = T.ID_time  
GROUP BY DateMonth, DateYear);
```

2. VM2

```
CREATE TABLE VM2 (  
DateMonth DATE CHECK (DateMonth IS NOT NULL),  
phoneRate VARCHAR(20) CHECK (phoneRate IS NOT NULL),  
TOT_NumberOfCalls INTEGER,  
TOT_Price INTEGER);  
  
INSERT INTO VM2 (DateMonth, phoneRate, TOT_NumberOfCalls, TOT_Price)  
(select DateMonth, phoneRateType, sum(NumberOfCalls), sum(Price)  
from phoneRate t, TIMEDIM te, FACTS f  
where f.ID_time = te.ID_time and f.id_phoneRate = t.id_phoneRate and te.DateYear=2003  
group by DateMonth, phoneRateType);
```

Triggers

1. TRIGGER 1

```
create TRIGGER Trigger1
AFTER INSERT ON FACTS
FOR EACH ROW
DECLARE
N NUMBER;
VAR_DateMonth DATE;
VAR_DateYear NUMBER;
BEGIN

SELECT DateMonth INTO VAR_DateMonth
FROM TIMEDIM
WHERE ID_time = :NEW.ID_time;

SELECT DateYear INTO VAR_DateYear
FROM TIMEDIM
WHERE ID_time = :NEW.ID_time;

SELECT Count(*) INTO N
FROM VM1
WHERE DateMonth=Var_DateMonth AND DateYear = Var_DateYear;

if (N > 0) then
    update VM1
    set TOT_NumberOfCalls = TOT_NumberOfCalls + :NEW.NumberOfCalls, TOT_Price =
    TOT_Price + :NEW.Price
    where DateMonth= Var_DateMonth AND DateYear = Var_DateYear;
else
    insert into VM1 (DateMonth, DateYear, Tot_NumberOfCalls, Tot_Price)
    values (Var_DateMonth, Var_DateYear, :NEW.NumberOfCalls, :NEW.Price);
end if;
```

END;

2. TRIGGER 2

create or replace NONEDITIONABLE TRIGGER Trigger2

AFTER INSERT ON FACTS

FOR EACH ROW

DECLARE

N NUMBER;

VAR_DateMonth DATE;

VAR_phoneRate VARCHAR(20);

BEGIN

SELECT DateMonth INTO VAR_DateMonth

FROM TIMEDIM

WHERE ID_time = :NEW.ID_time;

SELECT phoneRateType INTO VAR_phoneRate

FROM phoneRate

WHERE Id_phonerate = :NEW.Id_phonerate;

SELECT Count(*) INTO N

FROM VM2

WHERE DateMonth=Var_DateMonth AND phoneRate = Var_phoneRate;

if (N > 0) then

 update VM2

 set TOT_NumberOfCalls = TOT_NumberOfCalls + :NEW.NumberOfCalls, TOT_Price =
 TOT_Price + :NEW.Price

 where DateMonth= Var_DateMonth AND phoneRate = Var_phoneRate;

else

```

insert into VM2 (DateMonth, phoneRate, Tot_NumberOfCalls, Tot_Price)
values (Var_DateMonth, Var_phoneRate, :NEW.NumberOfCalls, :NEW.Price);

```

end if;

END;

Output VM1 after update via trigger.

MESE	ANNO	SUM(CHIAMATE)	SUM(PREZZO)
1 20-LUG-03	2003	73416	272838400
2 20-AGO-03	2003	153200	272836900
3 20-OTT-03	2003	64800	241677000
4 20-LUG-04	2004	73200	272836600
5 20-AGO-04	2004	73200	272836600
6 20-OTT-04	2004	64800	241677000

Output VM2 after update via trigger.

MESE	TIPO_TARIFFA	SUM(CHIAMATE)	SUM(PREZZO)
1 20-LUG-03	Notte	10216	3362600
2 20-LUG-03	Giorno	10200	11462500
3 20-LUG-03	Mattino	22200	233769000
4 20-LUG-03	24 ore su 24	10200	10071100
5 20-LUG-03	CartaNatale	20400	14173000
6 20-AGO-03	Notte	50200	3361150
7 20-AGO-03	Giorno	50200	11462650
8 20-AGO-03	Mattino	22200	233769000
9 20-AGO-03	24 ore su 24	10200	10071100
10 20-AGO-03	CartaNatale	20400	14173000
11 20-OTT-03	Notte	9000	3060000
12 20-OTT-03	Giorno	9000	10350000
13 20-OTT-03	Mattino	19800	206658000
14 20-OTT-03	24 ore su 24	9000	9009000
15 20-OTT-03	CartaNatale	18000	12600000
16 20-LUG-03	Festivi	100	100
17 20-LUG-03	Business	100	100