## Task 01: Create Materialized Views - ON DEMAND

CREATE MATERIALIZED VIEW mv\_monthly\_rep\_demand

BUILD IMMEDIATE

AS

SELECT tariff\_type

, tariff\_name

, tariff\_code

, operation\_method\_type

, am\_sum

, count\_tr

, ROUND ( 100 \* am\_sum / ( SUM ( am\_sum ) OVER (PARTITION BY tariff\_type) )

, 3 ) prg

, profit

FROM ( SELECT tariff\_type

, tariff\_name

, tariff\_code

, operation\_method\_type

, ROUND ( SUM ( payment\_sum / currency\_to\_dollar )

, 2 )

am\_sum

, COUNT ( payment\_sum / currency\_to\_dollar ) count\_tr

, ROUND ( SUM ( ( payment\_sum / currency\_to\_dollar ) \* tariff\_payment\_sum / 100 )

, 2 )

profit

FROM u\_sa\_data.tmp\_transactions\_info

WHERE tariff\_type IN ('Local Transfer', 'International Transfer')

AND event\_dt >= TRUNC ( TO\_DATE ( '01-APR-2007'

, 'dd-mon-yyyy' )

, 'month' )

AND event\_dt <= (TRUNC ( TO\_DATE ( ADD\_MONTHS ( TO\_DATE ( '01-APR-2007'

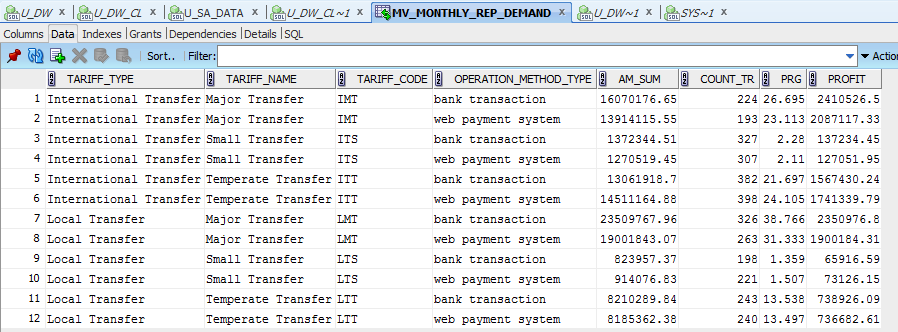
, 'dd-mon-yyyy' )

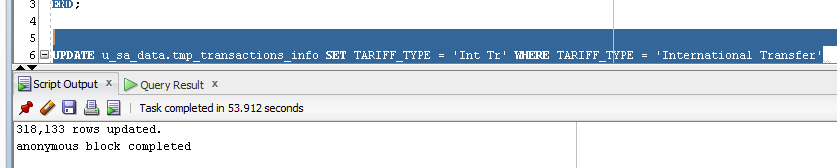
, 1 ) )

, 'month' )

- 1)

GROUP BY tariff\_type, tariff\_name, tariff\_code, operation\_method\_type );



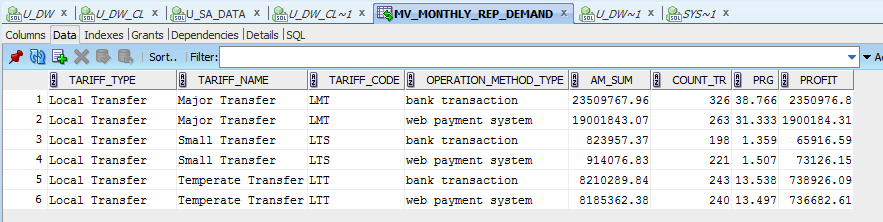


BEGIN

***dbms\_mview.refresh*** ( 'MV\_MONTHLY\_REP\_DEMAND'

, 'c' );

END;

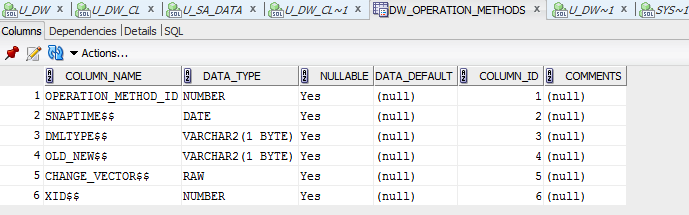


## Task 02: Create Materialized Views - ON COMMIT

CREATE MATERIALIZED VIEW LOG ON u\_sa\_data.tmp\_transactions\_info

WITH PRIMARY KEY

INCLUDING NEW VALUES;



CREATE MATERIALIZED VIEW mv\_daily\_rep\_commit

BUILD IMMEDIATE

REFRESH ON COMMIT

AS SELECT

tariff\_type

, tariff\_name

, tariff\_code

, operation\_method\_type

, SUM ( payment\_sum / currency\_to\_dollar )

am\_sum

, COUNT ( payment\_sum / currency\_to\_dollar ) count\_tr

, SUM ( ( payment\_sum / currency\_to\_dollar ) \* tariff\_payment\_sum / 100 )

profit

, COUNT ( ( payment\_sum / currency\_to\_dollar ) \* tariff\_payment\_sum / 100 ) cnt1

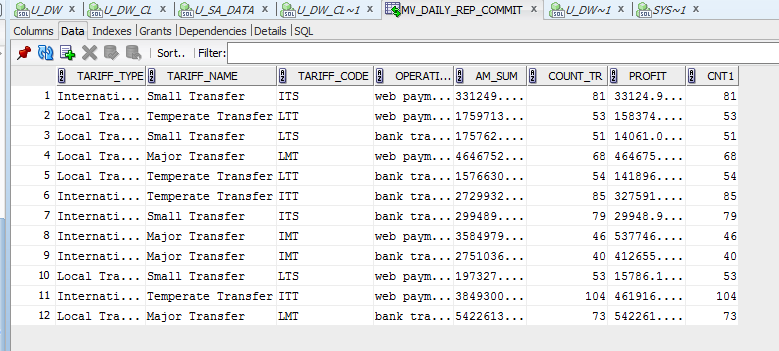
FROM u\_sa\_data.tmp\_transactions\_info

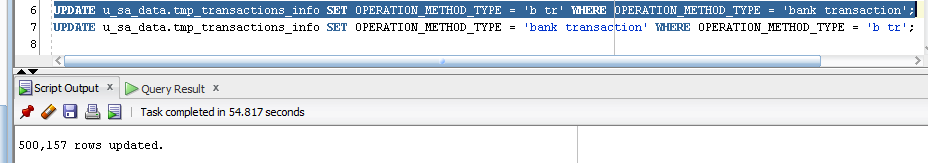
WHERE tariff\_type IN ('Local Transfer', 'International Transfer')

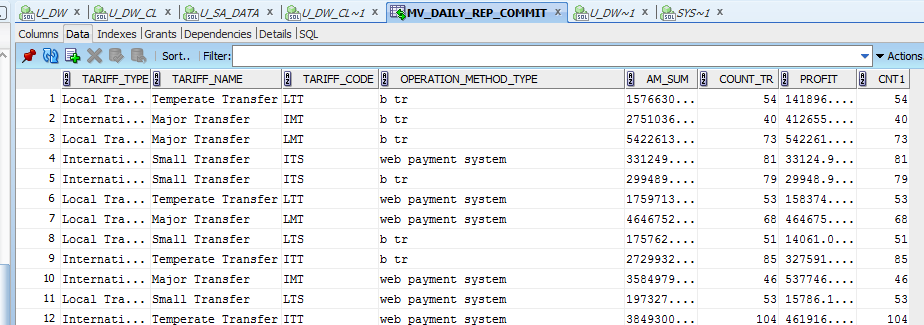
AND TRUNC ( event\_dt

, 'DAY' ) = TO\_DATE ( '15-APR-07' )

GROUP BY tariff\_type,tariff\_code, tariff\_name, operation\_method\_type;







## 

## Task 03: Create Materialized Views - Refreshing at definitive Time moment

CREATE MATERIALIZED VIEW monthly\_rep\_model

REFRESH START WITH SYSDATE NEXT SYSDATE + 1/1440

AS

SELECT event\_dt

, tariff\_type

, tariff\_name

, am\_sum

, count\_tr

, profit

, prg

FROM ( SELECT TRUNC ( TO\_DATE ( event\_dt

, 'dd-mon-yy' )

, 'month' )

event\_dt

, tariff\_type

, tariff\_name

, tariff\_payment\_sum

, SUM ( SUM ( payment\_sum / currency\_to\_dollar ) ) OVER (PARTITION BY tariff\_type) par\_sum

, SUM ( payment\_sum / currency\_to\_dollar )

am\_sum

, COUNT ( payment\_sum / currency\_to\_dollar ) count\_tr

FROM u\_sa\_data.tmp\_transactions\_info

WHERE tariff\_type IN ('Local Transfer', 'International Transfer')

AND tariff\_name IN ('Small Transfer', 'Temperate Transfer')

AND TO\_CHAR ( TRUNC ( event\_dt

, 'YYYY' )

, 'YYYY' ) IN ('2012')

GROUP BY (tariff\_type, tariff\_name, TRUNC ( TO\_DATE ( event\_dt

, 'dd-mon-yy' )

, 'month' ), tariff\_payment\_sum))

MODEL RETURN UPDATED ROWS

PARTITION BY ( tariff\_type, tariff\_name )

DIMENSION BY ( event\_dt )

MEASURES ( 0 prg, 0 profit, am\_sum, par\_sum, count\_tr, tariff\_payment\_sum )

RULES AUTOMATIC ORDER

( prg [event\_dt] = ROUND ( ( 100 \* am\_sum[CV ( event\_dt )] / par\_sum[CV ( event\_dt )] )

, 5 ),

profit [event\_dt] = ROUND ( ( count\_tr[CV ( event\_dt )] \* tariff\_payment\_sum[CV ( event\_dt )] / 100 )

, 2 ) );

