

Esercizio sulle viste materializzate: aggiornamento mediante triggers

È dato il seguente schema relazionale di un data mart

```
INCASSO(IdSede, IdServizio, IdCategoriaAzienda, IdTempo, #Consulenze, Incasso)
SERVIZIO(IdServizio, Consulenza, Tipologia, Categoria)
TEMPO(IdTempo, Data, Mese, Bimestre, Trimestre, Quadrimestre, Semestre, Anno)
SEDE-CONSULENTI(IdSede, Sede, Città, Regione, AreaGeografica, #Consulenti)
AZIENDA(IdCategoriaAzienda, CategoriaAzienda, TipologiaAzienda, Nazionalità)
```

È data la vista materializzata VM1 definita nel modo seguente

VM1 contiene l'importo incassato e il numero di consulenze erogate, separatamente per "Tipologia di Servizio" erogato, "Area geografica" della sede dei consulenti e "Semestre".

La vista materializzata VM1 è caratterizzata dallo schema seguente

```
VM1 (TipologiaServizio, AreaGeograficaSedeC, Semestre, TotIncasso, NumConsulenzeTot)
```

La vista materializzata VM1 deve essere gestita **senza utilizzare** l'istruzione CREATE MATERIALIZED VIEW. La tabella derivata che realizza la vista materializzata è definita dalla seguente istruzione SQL

```
CREATE TABLE VM1 (TipologiaServizio VARCHAR(20),
    AreaGeograficaSedeC VARCHAR(20),
    Semestre,    VARCHAR(20),
    TotIncasso   INTEGER CHECK (TotIncasso IS NOT NULL and TotIncasso >0),
    NumConsulenzeTot   INTEGER
    CHECK (NumConsulenzeTot IS NOT NULL and NumConsulenzeTot >0),
PRIMARY KEY (TipologiaServizio, AreaGeograficaSedeC, Semestre)
)
```

Esistono le seguenti dipendenze tra gli attributi, non gestiti da vincoli d'integrità (non sono definiti vincoli di chiave esterna):

- TipologiaServizio ha come dominio SERVIZIO(Tipologia)
- AreaGeograficaSedeC ha come dominio SEDE-CONSULENTI(AreaGeografica)
- Semestre ha come dominio TEMPO(Semestre)

La gestione degli aggiornamenti sulla tabella derivata (vista materializzata) VM1 richiede la scrittura di trigger specifici.

Punto 1

Si scriva l'istruzione di INSERT per il caricamento iniziale dei dati nella tabella derivata VM1.

Nota: è possibile popolare una tabella relazionale mediante l'istruzione INSERT(SELECT...).

Punto 2

Si scriva il trigger per propagare l'aggiornamento (considerando la modalità di gestione del tempo di tipo 1) del valore dell'attributo Tipologia di Servizio della tabella Servizio alla vista materializzata VM1.

Punto 3

Si scriva il trigger per propagare le modifiche causate dall'inserimento di un nuovo record nella tabella dei fatti INCASSO alla vista materializzata VM1.

Esercizio sulle viste materializzate: aggiornamento mediante CREATE MATERIALIZED VIEW LOG e CREATE MATERIALIZED VIEW in Oracle

È dato il seguente schema relazionale di un data mart

```
INCASSO(IdSede, IdServizio, IdCategoriaAzienda, IdTempo, #Consulenze, Incasso)
SERVIZIO(IdServizio, Consulenza, Tipologia, Categoria)
TEMPO(IdTempo, Data, Mese, Bimestre, Trimestre, Quadrimestre, Semestre, Anno)
SEDE-CONSULENTI(IdSede, Sede, Città, Regione, AreaGeografica, #Consulenti)
AZIENDA(IdCategoriaAzienda, CategoriaAzienda, TipologiaAzienda, Nazionalità)
```

È data la vista materializzata VM1 definita nel modo seguente

VM1 contiene l'importo incassato e il numero di consulenze erogate, separatamente per "Tipologia di Servizio" erogato, "Area geografica" della sede dei consulenti e "Semestre".

La vista materializzata VM1 è caratterizzata dallo schema seguente

```
VM1 (TipologiaServizio, AreaGeograficaSedeC, Semestre, TotIncasso, NumConsulenzeTot)
```

La vista materializzata VM1 è creata con la seguente istruzione SQL

```
CREATE MATERIALIZED VIEW
BUILD IMMEDIATE
REFRESH FAST ON DEMAND
ENABLE QUERY REWRITE
AS    <Query>
```

Punto 1

Definire la query necessaria per completare la definizione della vista materializzata VM1

Punto 2

Scrivere le istruzioni che definiscono i MATERIALIZED VIEW LOG in Oracle necessari per l'aggiornamento automatico FAST della vista materializzata VM1. Si indichino **tutti e soli** i log necessari e all'interno di ogni definizione di log **tutti e soli** gli attributi necessari.