Δημιουργία Αποθήκης Δεδομένων (Data Warehouse) με χρήση του DBMS SQL SERVER

Άσκηση

Μία εμπορική εταιρεία δραστηριοποιείται στην πώληση φαρμάκων και προϊόντων προσωπικής περιποίησης και υγιεινής. Η εταιρεία διαθέτει 15 υποκαταστήματα (φαρμακεία) σε 5 διαφορετικές περιοχές της Αθήνας.

Τα υποκαταστήματα είναι συνδεμένα σε ένα κεντρικό πληροφοριακό σύστημα το οποίο υποστηρίζει την απαιτούμενη λειτουργικότητα των πωλήσεων. Πρόκειται για ένα σύστημα OLTP στο οποίο είναι συνδεδεμένα όλα τα τερματικά (υπολογιστές και ταμειακές μηχανές) των σημείων πώλησης της εταιρείας.

Το τμήμα πωλήσεων (marketing) ενδιαφέρεται να αναπτύξει μία αποθήκη δεδομένων, ώστε να είναι σε θέση να αναλύει τα δεδομένα και να παράγει στατιστικές αναφορές σχετικά με την πορεία των πωλήσεων και την συμπεριφορά των πελατών της.

Τα δεδομένα που θα τροφοδοτούν την αποθήκη είναι διαθέσιμα υπό την μορφή ενός γραμμογραφημένου αρχείου κειμένου, το οποίο παράγεται από το πληροφοριακό σύστημα της εταιρείας. Κάθε εγγραφή του αρχείου αποτελείται από 17 πεδία.

Καλείστε να σχεδιάσετε και να υλοποιήσετε την παραπάνω αποθήκη δεδομένων προκειμένου να αυξήσετε την αποτελεσματικότητα της διεξαγωγής χρήσιμων στατιστικών στοιχείων, μειώνοντας ταυτόχρονα τον χρόνο εκτέλεσης των επερωτήσεων. Στην συνέχεια να τροφοδοτήσετε την αποθήκη με τα δεδομένα του αρχείου “OLTP.TXT” και να εκτελέσετε ορισμένες επερωτήσεις για την παραγωγή χρήσιμων στατιστικών στοιχείων.

Περιγραφή Αρχείου OLTP.TXT

Το αρχείο OLTP.TXT περιέχει 28582 εγγραφές κωδικοποιημένες σε UNICODE. Κάθε εγγραφή αποτελείται από 17 πεδία τα οποία διαχωρίζονται με τον χαρακτήρα “|” (pipe).

OLTP.TXT	
storecode	Κωδικός υποκαταστήματος (φαρμακείου)
customer_id	Κωδικός πελάτη
receipt_no	Αριθμός απόδειξης
quantity	Ποσότητα
manufacturer_coce	Κωδικός εταιρείας παραγωγής προϊόντος
Brand_code	Κωδικός ονομασίας προϊόντος
cat2	Διψήφιος κωδικός που δηλώνει την κατηγορία του προϊόντος (π.χ. 11, 22 κλπ)
cat4	Τετραψήφιος κωδικός που δηλώνει υποκατηγορία της κατηγορίας cat2 (π.χ. 1132, 1124, 2212, 2245 κ.λπ.)
cat6	Εξαψήφιο κωδικός που δηλώνει την υποκατηγορία της κατηγορίας cat4 (π.χ. 113261, 113262, 224516, 224517 κλπ).
dob	Ημερομηνία γέννησης του πελάτη

sex	Κωδικός φύλου (1=Ανδρας, 2=Γυναίκα)
cost	Κόστος προϊόντος
discount	Ποσό έκπτωσης
profit	Ποσό κέρδους
Product_id	Κωδικός προϊόντος
Location_code	Κωδικός τοποθεσίας υποκαταστήματος (Π.χ. 3=Χαλάνδρι, 9=Περιστέρι). Σε μια τοποθεσία μπορεί να υπάρχουν πολλά καταστήματα.
dtm	Πεδίο datetime το οποίο δηλώνει την χρονική στιγμή της πώλησης (χρονική στιγμή έκδοσης της απόδειξης).

Ενδεικτική Λύση

Βήμα 1: Από το περιβάλλον του Microsoft Sql Server Management Studio δημιουργούμε μια βάση δεδομένων με όνομα dsDW.

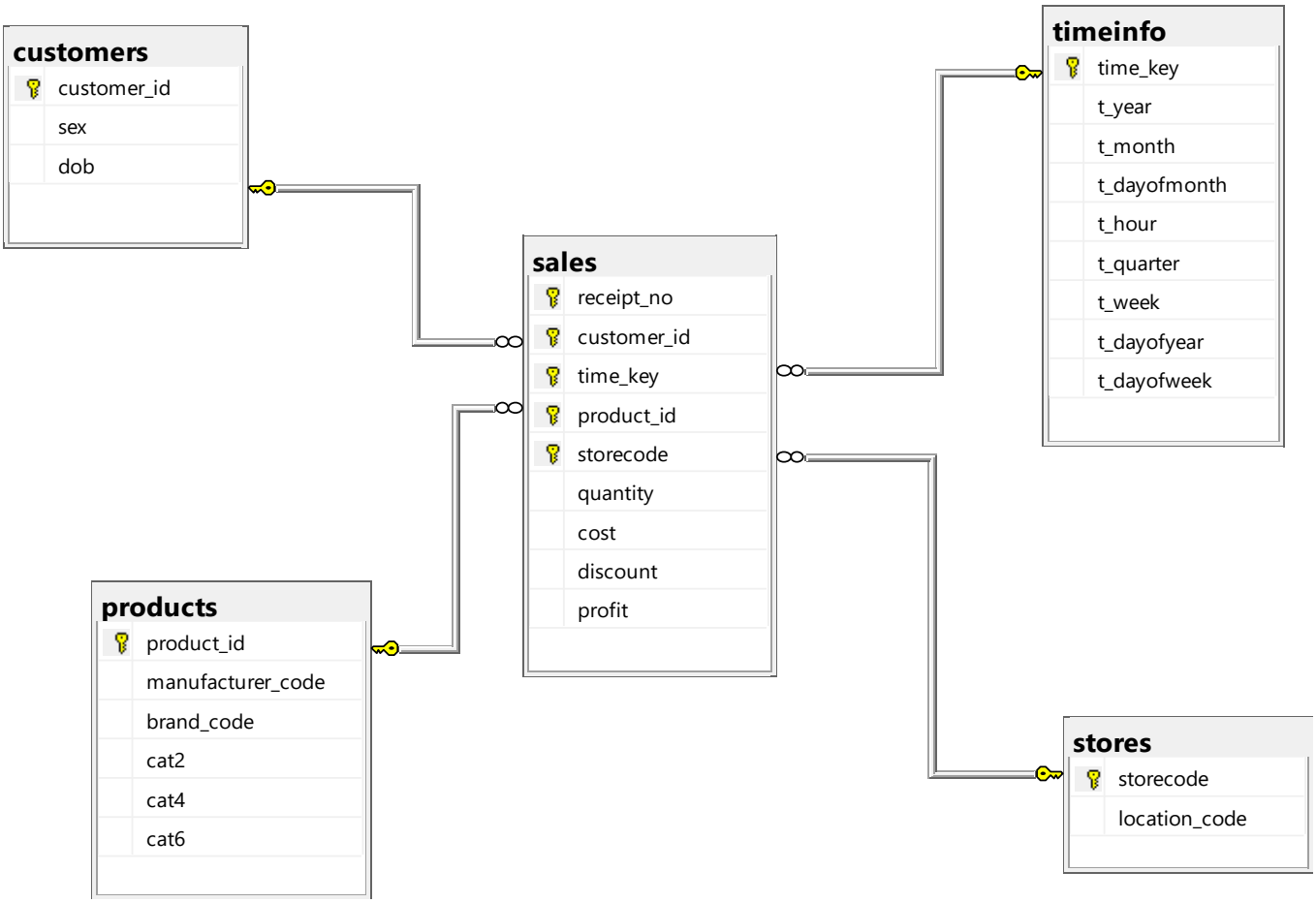
Βήμα 2: Δημιουργούμε τον πίνακα main και φορτώνουμε τα δεδομένα του αρχείου "OLTP.TXT".

```
Create table main
(storecode   numeric,
 customer_id numeric,
 receipt_no int,
 quantity  int,
 manufacturer_code int,
 brand_code numeric,
 cat2 int,
 cat4 int,
 cat6 int,
 dob  datetime,
 sex  int,
 cost numeric,
 discount numeric,
 profit numeric,
 product_id numeric,
 location_code int,
 dtm datetime
);

BULK INSERT main
 FROM 'C:\DATAWAREHOUSE\OLTP.TXT'
      WITH (DATAFILETYPE = 'widechar', FIRSTROW =2, FIELDTERMINATOR=
'|', ROWTERMINATOR = '\n');
```

Βήμα 3: Δημιουργούμε το σχήμα της αποθήκης δεδομένων. Στην συγκεκριμένη περίπτωση επιλέξαμε ένα μοντέλο αστέρα (star schema) με fact table τον πίνακα των πωλήσεων και τέσσερις πίνακες διαστάσεων (customers, products, stores, timeinfo).

```
create table customers
(customer_id numeric primary key,
 sex  int,
 dob  datetime,
);

create table products
(product_id numeric primary key,
 manufacturer_code int,
 brand_code numeric,
 cat2 int,
 cat4 int,
 cat6 int
 );

create table stores
(storecode   numeric primary key,
 location_code int
 );

create table timeinfo
(time_key  datetime primary key,
 t_year int,
 t_month int,
 t_dayofmonth int,
 t_hour int,
 t_quarter int,
 t_week int,
 t_dayofyear int,
 t_dayofweek int
 );

create table sales
(   receipt_no int,
    customer_id numeric ,
      time_key datetime,
      product_id numeric,
    storecode   numeric,
    quantity  int,
    cost numeric,
    discount numeric,
    profit numeric,

 primary key(receipt_no,customer_id,time_key,product_id,storecode),
 foreign key (customer_id) references customers(customer_id),
 foreign key (time_key) references timeinfo(time_key),
 foreign key (product_id) references products(product_id),
 foreign key (storecode) references stores(storecode),
 );
```

Διαγραμματική αναπαράσταση του σχήματος αστέρα της αποθήκης δεδομένων.



Βήμα 4: Τροφοδοτούμε με δεδομένα τους πίνακες της αποθήκης.

```
insert into customers
    select distinct customer_id, sex, dob
        from main


insert into products
    select distinct product_id, manufacturer_code, brand_code, cat2,
cat4,cat6
       from main;

insert into stores select distinct storecode, location_code from
main;


set datefirst 1;
insert into timeinfo
   select distinct dtm, datepart(year, dtm), datepart(month, dtm),
                        datepart(day,dtm),datepart(hour, dtm),
                        datepart(quarter,dtm), datepart(week,dtm),
                    datepart(dayofyear,dtm),datepart(dw,dtm)
     from main;


insert into  Sales
select  receipt_no, customer_id, dtm, product_id, storecode,
       sum(quantity),  sum(cost),  sum(discount), sum(profit)
    from main
  group by  receipt_no,customer_id, dtm, product_id, storecode
```

ΕΠΕΡΩΤΗΣΕΙΣ

1. Παράδειγμα επερώτησης με χρήση της διάστασης customers

Συνολική αξία πωλήσεων ανά φύλο (1=Ανδρας 2=Γυναίκα).

```
Select sex, sum(cost)
   from customers, sales
 where customers.customer_id=sales.customer_id
 group by sex
```

2. Παράδειγμα επερώτησης με χρήση της διάστασης stores

Συνολική αξία των πωλήσεων ανά περιοχή.

```
Select location_code, sum(cost)
   from stores, sales where stores.storecode=sales.storecode
   group by location_code
```

3. Παράδειγμα επερώτησης με χρήση της διάστασης products

Συνολικά κέρδη ανά κωδικό εμπορικής ονομασίας προϊόντων (brand_code).

```
Select brand_code, sum(profit)
   from products, sales where products.product_id=sales.product_id
  group by brand_code
```

4. Παράδειγμα επερώτησης με χρήση της διάστασης timeinfo

Συνολική αξία πωλήσεων και κερδών του έτους 1998 ανά μήνα.

```
select t_month, sum(cost), sum(profit)
      from timeinfo, sales
  where timeinfo.time_key = sales.time_key and t_year=1998
      group by t_month
        order by t_month
```

5. Παραδείγματα επερωτήσεων με συνδυασμό διαστάσεων.

Συνολική αξία των πωλήσεων ανά έτος, περιοχή και ημέρα της εβδομάδας.

```
select t_year, location_code, t_dayofweek, sum(cost)
      from timeinfo, stores, sales
  where timeinfo.time_key = sales.time_key and
        stores.storecode=sales.storecode
      group by   t_year, location_code, t_dayofweek
        order by t_year, location_code, t_dayofweek
```

6. Αριθμός πωλήσεων ανά έτος με αξία μεγαλύτερη των 20000.

```
create view v1 as
select t_year, receipt_no, sum(cost) as cost
      from timeinfo, sales
  where timeinfo.time_key = sales.time_key
      group by   t_year, receipt_no
        having sum(cost) > 20000
        order by t_year


select t_year, count(*) from v1 group by t_year
```

7. Παράδειγμα επερώτησης με χρήση του τελεστή ROLLUP

Ανάλυση κερδών ανά έτος, τοποθεσία, και κατηγορία (cat2) προϊόντων.

```
-- Ανάλυση κερδών ανά έτος, περιοχή και κατηγορία (cat2) προϊόντων

select t_year, location_code, cat2, sum(profit)
  from timeinfo, stores, products, sales
  where timeinfo.time_key=sales.time_key and
        stores.storecode=sales.storecode and
            products.product_id=sales.product_id
  group by ROLLUP (t_year, location_code, cat2)
```

Το αποτέλεσμα της επερώτησης είναι η ομαδοποίηση των κερδών με βάση τους παρακάτω συνδυασμούς:

- t_year, location_code, cat2 (group by t_year, location_code, cat2)
- t_year, location_code, null (group by t_year, location_code)
- t_year, null, null (group by t_year)
- null, null, null --**Συνολικό κέρδος** (group by none)

8. Data Cube

```
select t_year, location_code, cat2, sum(profit)
  from timeinfo, stores, products, sales
  where timeinfo.time_key=sales.time_key and
        stores.storecode=sales.storecode and
            products.product_id=sales.product_id
  group by CUBE (t_year, location_code, cat2)
```

Το αποτέλεσμα της επερώτησης είναι η δημιουργία ενός κύβου, κάθε κελί του οποίου περιέχει τα κέρδη των πωλήσεων για έναν συνδυασμό τιμών (έτους, περιοχής, κατηγορίας).

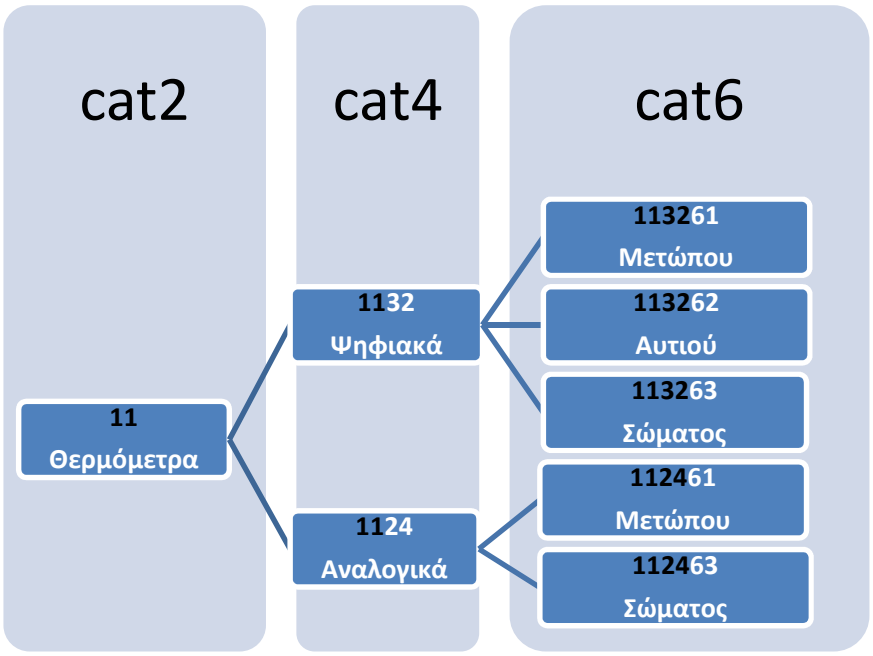
- t_year, location_code, cat2 (group by t_year, location_code, cat2)
- t_year,location_code , null (group by t_year, location_code)
- t_year, null, cat2 (group by t_year, cat2)
- null, location_code, cat2 (group by location_code, cat2)
- t_year, null, null (group by t_year)
- null, location_code, null (group by location_code)
- null, null, cat2 (group by cat2)
- null, null, null (group by none)

Σημείωση: Η τιμή **null** αναπαριστά την τιμή **ALL.**

9. Κατηγοριοποίηση Προϊόντων Φαρμακείου

cat2	Διψήφιος κωδικός που δηλώνει την κατηγορία του προϊόντος
cat4	Τετραψήφιος κωδικός που δηλώνει υποκατηγορία της κατηγορίας cat2
cat6	Εξαψήφιο κωδικός που δηλώνει την υποκατηγορία της κατηγορίας cat4

Παράδειγμα Ιεραρχίας:



Ανάλυση του κερδών ανά κατηγορία προϊόντος.

ROLLUP

```
select cat2, cat4, cat6,sum(profit)
   from products, sales
  where products.product_id=sales.product_id
   group by rollup (cat2, cat4, cat6)
 order by cat2, cat4, cat6
```

Το αποτέλεσμα της επερώτησης είναι η ομαδοποίηση των κερδών με βάση τους παρακάτω συνδυασμούς των κατηγοριών cat2, cat4 και cat6:

- cat2, cat4, cat6 (group by cat2, cat4, cat6)
- cat2, cat4, null (group by cat2, cat4)
- cat2, null, null (group by cat2)
- null, null, null -- **Συνολικό κέρδος** (group by none)

ΠΡΟΣΟΧΗ: Σε περιπτώσεις ιεραρχιών, όπως στην συγκεκριμένη περίπτωση, παρατηρούμε ότι το group by (cat2, cat4, cat6) είναι στην πραγματικότητα ισοδύναμο με το group by (cat6) και το group by (cat2, cat4) είναι ισοδύναμο με το group by (cat4).