SQL: Αιτήματα

Κεφάλαιο 5

Ενημέρωση: 23/12/2008

Στιγμιότυπα

- Στιγμιότυπα των σχέσεων Sailors Reserves και Boats.
- Αν στο κλειδί της σχέσης Reserved δε συμμετείχε το γνώρισμα *date*, ποια η σημασιολογία του Σχήματος;

S1

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

B1

bid	bcolor
101	green
102	red
103	red
104	green

R1

sid	bid	day
22	101	10/10/96
58	103	11/12/96

Βασικό Αίτημα SQL

SELECT [DISTINCT] *λίστα-γνωρισμάτων*
FROM *λίστα-σχέσεων*
WHERE *καταλληλότητα*

- λίστα-σχέσεων Μιά λίστα σχέσεων (πιθανότατα με μια μεταβλητή διαστήματος τιμών μετά από κάθε σχέση).
- λίστα-γνωρισμάτων Λίστα γνωρισμάτων των σχέσεων της λίστας-σχέσεων
- καταλληλότητα Συγκρίσεις (Γνώρισμα - Τελεστής - Σταθερά ή Γνώρισμα - Τελεστής - Γνώρισμα). Τελεστές: $>, =, \leq$

  Συνδυασμός με: AND, OR, NOT.
- DISTINCT προαιρετική λέξη κλειδί. Ορίζει ότι στο αποτέλεσμα δε θα περιέχονται διπλοεγγραφές. Προκαθορισμένη επιλογή είναι η διατήρηση των διπλοεγγραφών.

Ιδεατή Στρατηγική Υπολογισμού

- Το επιστρεφόμενο σύνολο ενός αιτήματος υπολογίζεται βάσει της ακόλουθης ιδεατής στρατηγικής:
  - Υπολογισμός του καρτεσιανού γινομένου της λίστας-σχέσεων (αν υπάρχουν περισσότερες από μία σχέσεις).
  - Απόρριψη των ακατάλληλων πλειάδων.
  - Απόκρυψη γνωρισμάτων εκτός λίστας-γνωρισμάτων.
  - Αν δηλωθεί το DISTINCT διαγραφή διπλοεγγραφών.
- Η στρατηγική αυτή είναι πιθανότατα η λιγότερο αποδοτική για τον υπολογισμό του αποτελέσματος. Ο βελτιστοποιητής θα ανακαλύψει αποτελεσματικότερες στρατηγικές για τον υπολογισμό του ίδιου αποτελέσματος.

Παράδειγμα Ιδεατού Υπολογισμού (1)

```
SELECT  S.sname
FROM    Sailors S
WHERE   S.age > 40.0
```

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Παράδειγμα Ιδεατού Υπολογισμού (2)

```
SELECT  S.sname
FROM    Sailors S, Reserves R
WHERE   S.sid=R.sid AND R.bid=103
```

(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/
22	dustin	7	45.0	58	103	11/12/
31	lubber	8	55.5	22	101	10/10/
31	lubber	8	55.5	58	103	11/12/
58	rusty	10	35.0	22	101	10/10/
58	rusty	10	35.0	58	103	11/12/

Μεταβλητές Διαστήματος

- Χρησιμοποιούνται υποχρεωτικά μόνο αν μια σχέση επανεμφανίζεται στη FROM. Διαφορετικά είναι προαιρετικές Το προηγούμενο αίτημα διαφορετικά διατυπωμένο:

```
SELECT  S.sname
FROM    Sailors S, Reserves R
WHERE   S.sid=R.sid AND bid=103
```

Ή

```
SELECT  sname
FROM    Sailors, Reserves
WHERE   Sailors.sid=Reserves.sid
        AND bid=103
```

Γενικώς είναι καλή πρακτική η χρήση Μεταβλητών Διαστήματος

Εδώ δεν χρησιμοποιούνται μεταβλητές διαστήματος

Να βρεθούν οι ναύτες που έχουν τουλάχιστο μία κράτηση

```
SELECT  S.sid
FROM  Sailors S, Reserves R
WHERE  S.sid=R.sid
```

- Πως αλλάζει το επιστρεφόμενο σύνολο αν προστεθεί το DISTINCT;
- Τι γίνεται αν αντικαταστήσουμε το S.sid με το S.sname στη SELECT; Πως επηρεάζει η DISTINCT αυτή την εκδοχή του αιτήματος;

Παραστάσεις και Συμβολοσειρές

```
SELECT  S.age, age1=S.age-5, 2*S.age AS age2
FROM  Sailors S
WHERE  S.sname LIKE 'B_%B'
```

- Αριθμητικές παραστάσεις και ταύτιση συμβολοσειρών: *Προβολή ηλικίας και δύο επιπρόθετων πεδίων που προκύπτουν από αριθμητικές παραστάσεις. Επιλέγονται οι ναύτες με όνομα που έχει ως πρώτο και τελευταίο χαρακτήρα το B και έχει τουλάχ. 3 χαρακτήρες.*
- AS και = τρόποι απόδοσης ονόματος σε παραστάσεις.
- LIKE ταυτίζει συμβολοσειρές. '_': ένας οποιοσδήποτε χαρακτήρας. `%': 0 ή περισσότεροι.

UNION (1)

- Μπορούμε να χρησιμοποιήσουμε το UNION για τον υπολογισμό της ένωσης δύο συμβατών ως προς την ένωση συνόλων (που προκύπτουν από αιτήματα SQL).
- *Π.χ. Εύρεση του sid των ναυτών που έχουν κράτηση σε κόκκινη ή πράσινη βάρκα*

```
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND
R.bid=B.bid
            AND B.color='red'
UNION
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND
R.bid=B.bid
            AND B.color='green'
```

EXCEPT (Τι θα συμβεί αν αντικαταστήσουμε το UNION με το EXCEPT?)

UNION (2)

- Εναλλακτικά μπορούμε να υπολογίσουμε το ερώτημα της προηγούμενης διαφάνειας ως εξής

```
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND R.bid=B.bid
  AND (B.color='red' OR B.color='green')
```

Τι θα συμβεί αν αντικαταστήσουμε το OR με το AND;

INTERSECT (1)

- INTERSECT: Χρησιμοποιείται για τον υπολογισμό της τομής δύο συμβατών ως προς την ένωση συνόλων.
- Συμπεριλαμβάνεται στο πρότυπο SQL/92 αλλά δεν υποστηρίζεται απ' όλα τα συστήματα.
- *Π.χ. Εύρεση του sid των ναυτών που έχουν κράτηση σε κόκκινη και σε πράσινη βάρκα*

Πεδίο κλειδί!

```
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND
R.bid=B.bid
            AND B.color='red'
INTERSECT
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND
R.bid=B.bid
            AND B.color='green'
```

INTERSECT (2)

- Εναλλακτικά μπορούμε να υπολογίσουμε το ερώτημα της προηγούμενης διαφάνειας ως εξής

```
SELECT  S.sid
FROM  Sailors S, Boats B1, Reserves R1,
          Boats B2, Reserves R2
WHERE  S.sid=R1.sid AND R1.bid=B1.bid
  AND  S.sid=R2.sid AND R2.bid=B2.bid
  AND (B1.color='red' AND B2.color='green')
```

Δεν αρκεί η απλή αλλαγή του OR σε AND της εναλλακτικής περίπτωσης της UNION

Εμφωλευμένα Αιτήματα (1)

Εμφάνιση ονομάτων ναυτών με κράτηση στη βάρκα #103:

```
SELECT  S.sname
FROM  Sailors S
WHERE  S.sid IN  (SELECT  R.sid
                  FROM  Reserves R
                  WHERE  R.bid=103)
```

- Για την κατανόηση του τρόπου υπολογισμού των εμφωλευμένων αιτημάτων αντιπαραβολή με τον τρόπο εκτέλεσης των εμφωλευμένων βρόχων στον προγραμματισμό. Για κάθε πλειάδα του Sailors ελέγχεται η καταλληλότητα αφού υπολογιστεί το υποαίτημα.

Εμφωλευμένα Αιτήματα (2)

- Στο προηγούμενο παράδειγμα η NOT IN επιστρέφει τους ναύτες χωρίς κρατήσεις.
- Στη συνιστώσα WHERE μπορεί να συμμετέχει ένα αίτημα SQL. (Ανάλογα: στη FROM και στη HAVING).

Εμφωλευμένα Αιτήματα με Αμοιβαία Εξάρτηση (1)

Εμφάνιση ονομάτων ναυτών με κράτηση στη βάρκα #103:

```
SELECT  S.sname
FROM  Sailors S
WHERE  EXISTS  (SELECT  *
                FROM  Reserves R
                WHERE  R.bid=103 AND S.sid=R.sid)
```

- EXISTS: Τελεστής σύγκρισης για σύνολα (όπως το IN). Αν ένα σύνολο δεν είναι κενό τότε ικανοποιείται η συνθήκη και η υπό εξέταση εγγραφή επιστρέφεται στο χρήστη

Εμφωλευμένα Αιτήματα με Αμοιβαία Εξάρτηση (2)

- Στο προηγούμενο παράδειγμα αν χρησιμοποιηθεί το UNIQUE, και το το R.bid αντικαταστήσει το *, επιστρέφονται οι ναύτες με μία κράτηση στη βάρκα #103 το πολύ. (Το UNIQUE ελέγχει για διπλοεγγραφές. Το * συμβολίζει όλα τα γνωρίσματα. Γιατί είναι απαραίτητη η αντικατάσταση του * με το R.bid;)
- Αναδεικνύει τους λόγους επανυπολογισμού του υποαιτήματος για κάθε πλειάδα του Sailors.

Υπόλοιποι Τελεστές Συνόλων

- Αντίστοιχα με τα IN, EXISTS και UNIQUE υπάρχουν τα NOT IN, NOT EXISTS και NOT UNIQUE.
- Επίσης διαθέσιμοι: τελ. ANY, τελ. ALL με >, <, ≥, ≤, =, ≠
- Να βρεθούν οι ναύτες με διατίμηση μεγαλύτερη από του Horatio:

```
SELECT  *
FROM  Sailors S
WHERE  S.rating > ANY  (SELECT  S2.rating
                        FROM  Sailors S2
                        WHERE S2.sname='Horatio')
```

Επαναδιατύπωση με IN των Αιτημάτων INTERSECT (1)

Εμφάνιση του sid των ναυτών με κράτηση σε κόκκινη και πράσινη βάρκα:

```
SELECT  S.sid
FROM  Sailors S, Boats B, Reserves R
WHERE  S.sid=R.sid AND R.bid=B.bid AND B.color='red'
        AND S.sid IN  (SELECT  S2.sid
                       FROM  Sailors S2, Boats B2, Reserves R2
                       WHERE  S2.sid=R2.sid AND R2.bid=B2.bid
                              AND  B2.color='green')
```

Επαναδιατύπωση με IN των Αιτημάτων INTERSECT (2)

- Για τα ονόματα (όχι τους κωδικούς) των ναυτών με κρατήσεις σε κόκκινες και πράσινες βάρκες, αντικατάσταση S.sid από S.sname στη SELECT. Αν θέλαμε την ίδια λύση αλλά με INTERSECT;

Εξάσκηση (1)

Να βρεθούν τα ονόματα και οι ηλικίες όλων των ναυτικών

```
SELECT  S.sname, S.age
FROM  Sailors S
```

Να βρεθούν όλοι οι ναυτικοί οι οποίοι έχουν τιμή διατίμησης (rating) μεγαλύτερη του 7

```
SELECT  S.sid, S.sname, S.rating, S.age
FROM  Sailors S
WHERE S.rating > 7
```

Εξάσκηση (2)

Να βρεθούν τα χρώματα των βαρκών στις οποίες έχει κάνει κράτηση ναυτικός που ονομάζεται Yiannis

```
SELECT  B.color
FROM  Sailors S, Reserves R, Boats B
WHERE S.sid = R.sid AND R.bid = B.bid
AND S.sname = 'Lubber'
```

Περίληψη

- Η SQL αποτέλεσε σημαντικό παράγοντα για την αποδοχή του σχεσιακού μοντέλου. Πιο φυσική από προηγούμενες, διαδικαστικές γλώσσες αιτημάτων.
- Χαρακτηρίζεται από σχεσιακή πληρότητα. Έχει περισσότερες δυνατότητες από τη σχεσιακή άλγεβρα.
- Αιτήματα που εκφράζονται σε ΣΑ εκφράζονται με φυσικότερο τρόπο σε SQL.
- Εναλλακτικοί τρόποι διεκπεραίωσης των αιτημάτων. Ο βελτιστοποιητής ανακαλύπτει τον αποδοτικότερο.
  - Στην πράξη ωστόσο οι χρήστες πρέπει να έχουν γνώση των σχεδίων εκτέλεσης/επεξεργασίας των αιτημάτων που καταθέτουν.