

Laboratorul 8

De la BD

Tematica:

- gruparea înregistrărilor
- aplicarea funcțiilor de agregare

Cuprins

- 1 Funcții de agregare în SQL
- 2 Gruparea datelor
- 3 Filtrarea grupurilor
- 4 Exerciții

Funcții de agregare în SQL

Funcțiile de agregare sunt funcții care, utilizate în interogări, returnează o singură linie rezultat pe baza unui grup de linii. Pot fi utilizate în cadrul listei SELECT și în clauzele ORDER BY și HAVING.

ATENȚIE: nu utilizați funcțiile de agregare în clauza WHERE !!!

Funcțiile de agregare pot accepta clauzele

- DISTINCT – specifică luarea în considerare doar a valorilor distincte de către funcție
- ALL – este implicit, determină funcția de agregare să calculeze rezultatul pe baza tuturor valorilor

| | | |
|--|--|--|
| <pre> COUNT ((* [DISTINCT ALL] expr)) </pre> | <p>Returnează numărul de linii. Dacă argumentul este * se numără și valorile NULL, altfel se numără doar valorile nenule. Argumentul poate lua tipul CHAR, VARCHAR2, NUMBER, DATE.</p> | <pre> SELECT COUNT(*) AS "Total studenti" FROM studenti; SELECT COUNT(NVL(bursa,0)) AS "Total studenti" FROM studenti; SELECT COUNT(bursa) "Studenti bursieri" FROM studenti; </pre> |
|--|--|--|

| | | |
|----------------------------|--|--|
| | | SELECT COUNT(DISTINCT bursa) "Variante bursa" FROM studenti; |
| AVG ([DISTINCT ALL]n) | Returnează media valorilor. Argumentul trebuie să fie numeric. | SELECT AVG(valoare) FROM note; |
| MAX ([DISTINCT ALL]expr) | Returnează valoare maximă. Argumentul poate lua tipul CHAR, VARCHAR2, NUMBER, DATE | SELECT MAX(bursa) FROM studenti; |
| MIN ([DISTINCT ALL]expr) | Returnează valoare minimă. Argumentul poate lua tipul CHAR, VARCHAR2, NUMBER, DATE | SELECT MIN(bursa) FROM studenti; |
| STDDEV ([DISTINCT ALL]x) | Returnează varianța valorilor. Argumentul trebuie să fie numeric. | SELECT STDDEV(bursa) "Deviatie standard" FROM studenti; |
| VARIANCE ([DISTINCT ALL]x) | Returnează varianța valorilor. Argumentul trebuie să fie numeric. | SELECT VARIANCE(bursa) "Varianta" FROM studenti; |
| SUM ([DISTINCT ALL]n) | Returnează suma valorilor. Argumentul trebuie să fie numeric. | SELECT SUM(bursa) FROM studenti; |

Toate funcțiile de agregare ignoră valorile NULL. Excepție face funcția COUNT cu argumentul * care va considera NULL ca fiind o valoare distinctă. Pentru ca valoarea NULL să fie luată în considerare se poate utiliza NVL în cadrul funcției de agregare pentru a substitui NULL cu o valoare anume (ex. 0).

Dacă setul de date nu conține linii sau conține doar linii cu valori NULL ca argumente a funcțiilor de agregare, funcția va returna NULL. Excepție face COUNT care returnează 0 sau un alt număr.

Gruparea datelor

Toate exemplele ilustrate anterior aplică funcțiile de agregare asupra întregului conținut al tabelului, rezultatul interogării fiind alcătuit dintr-o singură linie.

Funcțiile de agregare sunt însă utilizate în general împreună cu clauza GROUP BY care specifică împărțirea liniilor tabelului interogat în mai multe grupuri; Oracle aplică funcțiile de agregare fiecărui astfel de grup returnând un singur rezultat pentru fiecare grup.

Reamintim sintaxa interogării în Oracle:

```
SELECT [DISTINCT | ALL] { * | [expresie_coloana [AS nume_nou]] [, ...] }
FROM nume_tabel [alias] [, ...]
[WHERE conditie]
[GROUP BY expresie [HAVING conditie] ]
[ORDER BY expresie1 [ASC|DESC] [, ...]]
```

Exemplu:

```
SELECT MAX(valoare)
  FROM note
  GROUP BY nr_matricol;
--Câte linii sunt returnate? Ce reprezintă fiecare linie?
```

Valori individuale nu pot fi utilizate în lista select decât dacă în clauză GROUP BY este specificată coloana individuală. Deci toate câmpurile din lista select care nu sunt utilizate în cadrul funcțiilor de agregare trebuie să apară în clauza GROUP BY.

Exemplu:

```
SELECT MAX(valoare), id_curs
  FROM note
  GROUP BY nr_matricol;
--De ce nu merge?
```

Grupurile pot fi create și pe baza mai multor atribute: Exemplu:

```
SELECT an, count(*)
  FROM cursuri
  GROUP BY an;
--Cate grupuri?

SELECT credite, count(*)
  FROM cursuri
  GROUP BY credite;
--Cate grupuri?

SELECT an, credite, count(*)
  FROM cursuri
  GROUP BY an, credite
  ORDER BY an, credite;
--Cate grupuri?
```

Nu este obligatoriu ca atributele utilizate în clauza GROUP BY să apară în lista select; este însă indicată includerea acestora pentru ca înregistrările rezultat să poată fi interpretate.

Filtrarea grupurilor

Clauza WHERE elimină valori individuale (linii) înainte de a avea loc gruparea.

Clauza HAVING este utilizată doar în conjuncție cu clauza GROUP BY având rolul de a elimina unele grupuri pe baza rezultatului funcțiilor de agregare și nu a valorilor individuale ale liniilor.

Exemplu:

```
SELECT id_curs, COUNT(valoare)
```

```
FROM note
GROUP BY id_curs;
--Cate grupuri?

SELECT id_curs, COUNT(valoare)
FROM note
WHERE valoare>8
GROUP BY id_curs;
--Cate grupuri? De ce?

SELECT id_curs, COUNT(valoare)
FROM note
GROUP BY id_curs
HAVING COUNT(valoare)>8;
--Cate grupuri? De ce?
```

Exerciții

1. Afișați numărul de studenți din fiecare an.
2. Afișați numărul de studenți din fiecare grupă a fiecărui an de studiu. Ordonați crescător după anul de studiu și după grupă.
3. Afișați numărul de studenți din fiecare grupă a fiecărui an de studiu și specificați câți dintre aceștia sunt bursieri.
4. Afișați suma totală cheltuită de facultate pentru acordarea burselor.
5. Afișați valoarea bursei/cap de student (se consideră că studentii care nu sunt bursieri primesc 0 RON); altfel spus: cât se cheltuiește în medie pentru un student?
6. Afișați numărul de note de fiecare fel (câte note de 10, câte de 9,etc.). Ordonați descrescător după valoarea notei.
7. Afișați numărul de note pus în fiecare zi a săptămânii. Ordonați descrescător după numărul de note.
8. Afișați pentru fiecare elev care are măcar o notă, numele și media notelor sale. Ordonați descrescător după valoarea mediei.
9. Modificați interogarea anterioară pentru a afișa și elevii fără nici o notă. Media acestora va fi null.
10. Modificați interogarea anterioară pentru a afișa pentru elevii fără nici o notă media 0.
11. Modificați interogarea de mai sus pentru a afișa doar studentii cu media mai mare ca 8.
12. Afișați numele, cea mai mare notă, cea mai mică notă și media doar pentru acei studenți care au cea mai mică notă mai mare sau egală cu 7.
13. Afișați numele și mediile studenților care au cel puțin un număr de 4 note puse în catalog.
14. Afișați numele și mediile studenților din grupa A2 anul 3.
15. Afișați cea mai mare medie obținută de vreun student.
16. Pentru fiecare disciplină de studiu afișați titlul acesteia, cea mai mică și cea mai mare notă pusă.

Adus de la „http://85.122.23.37/BD/index.php?title=Laboratorul_8&oldid=388”

-
- Ultima modificare efectuată la 19:08, 22 noiembrie 2015.
 - Conținutul este disponibil sub Creative Commons Atribuire, exceptând cazurile în care se specifică altfel.