

Bases de données - Concepts, utilisation et développement

5e édition • Dunod, 2022

Jean-Luc Hainaut

ERRATA et COMPLEMENTS

Ce document répertorie et corrige les erreurs repérées à ce jour par les lecteurs et par l'auteur dans **la cinquième édition**, et susceptibles d'entraîner des problèmes de compréhension (la plupart des autres coquilles, fautes d'orthographe ou de mise en page ne sont pas reprises). Il signale également certains compléments d'information.

Merci aux lecteurs qui me signalent des erreurs et suggestions de toute nature.

Création : 10 août 2022

Dernière mise-à-jour : 18 avril 2023

Avant-propos

Chapitre et annexe 1

Chapitre et annexe 2

Chapitre et annexe 3

Chapitre et annexe 4

Chapitre et annexe 5

Chapitre et annexe 6

Chapitre et annexe 7

Chapitre et annexe 8

Chapitre et annexe 9

Chapitre et annexe 10

- [2023-04-18] Chapitre 10, page 344, 4^e puce (**n preceding**), 3^e tiret (- si l'unité est range), remplacer :
si l'unité est range, désigne la première ligne du groupe dont $COMPTE \geq 60 - n$
(soit 45) si l'ordre est ascendant et dont **$COMPTE \leq 60 - n$ (soit 45)** si l'ordre est descendant ;
par:
si l'unité est range, désigne la première ligne (du premier groupe) dont $COMPTE \geq 60 - n$

(soit 45) si l'ordre est ascendant et dont **COMPTE $\leq 60 + n$ (soit 75)** si l'ordre est descendant ;

- [2023-04-18] Chapitre 10, page 344, 5^e puce (**n following**), 3e tiret (- si l'unité est *range*), remplacer :
si l'unité est *range*, désigne la dernière ligne du groupe dont **COMPTE $\leq 60 + n$ (soit 75)** si l'ordre est ascendant et dont **COMPTE $\geq 60 + n$ (soit 75)** si l'ordre est descendant ;
par:
si l'unité est *range*, désigne la dernière ligne (du dernier groupe) dont **COMPTE $\leq 60 + n$ (soit 75)** si l'ordre est ascendant et dont **COMPTE $\geq 60 - n$ (soit 45)** si l'ordre est descendant ;

Les frames construits via les clauses ***range***, ***preceding*** et ***following*** sont particulièrement puissants mais en revanche leur définition est assez délicate. On suggère d'explorer le résultat des requêtes suivantes pour bien comprendre leur fonctionnement :

```
createDB InMemory;

create table WIN(N integer);
insert into WIN values
    (30),(35),(40),(40),(45),(45),(45),(50),(55),(60),(65),(70),(70),(80),(80);

write-b ASCENDING;
select row_number() over () as Seq,
       N,
       count(*) over (order by N asc range between 15 preceding and current row) as "Prec",
       count(*) over (order by N asc range between current row and 15 following) as "Foll",
       count(*) over (order by N asc range between 15 preceding and 15 following) as "All"
from   WIN;

write-b DESCENDING;
select row_number() over () as Seq,
       N,
       count(*) over (order by N desc range between 15 preceding and current row) as "Prec",
       count(*) over (order by N desc range between current row and 15 following) as "Foll",
       count(*) over (order by N desc range between 15 preceding and 15 following) as "All"
from   WIN;

closeDB;
```

Chapitre et annexe 11

Chapitre et annexe 12

Chapitre et annexe 13

Chapitre et annexe 14

Chapitre et annexe 15

Chapitre et annexe 16

Chapitre et annexe 17

Chapitre et annexe 18

Chapitre et annexe 19

Chapitre et annexe 20

Chapitre et annexe 21

Chapitre et annexe 22

Chapitre et annexe 23

Chapitre et annexe 24

Annexe 25

Annexe 26

Annexe 27

Annexe 28

Annexe 29

Annexe 30