

Abelos

Initiation au langage SQL

Gaëlle PERRIN -SID2 Grenoble
Le 10/04/2014



Introduction aux bases de données relationnelles

»



Bases de données relationnelles

- ▶ Base de Données (BD) :
 - une grande quantité de données, centralisées ou non, servant pour les besoins d'une ou plusieurs applications, interrogeables et modifiables par un groupe d'utilisateurs travaillant en parallèle
 - un ensemble d'informations sur un sujet : exhaustif, non redondant, structuré, persistant.
- ▶ Système de Gestion de Bases de Données (SGBD)
 - l'interface entre la base de données et les utilisateurs ou leurs programmes.
 - un logiciel qui permet de : décrire, modifier, interroger, administrer les données d'une base de données.



Types de données

- ▶ chaînes de caractères (alphanumérique) :
 - longueur fixe : CHAR(n)
 - longueur variable avec limite: VARCHAR(n)
- ▶ chaînes numériques
 - entier : INTEGER, SMALLINT, BIGINT
 - flottant non précis: REAL, DOUBLE PRECISION
 - flottant précis : NUMERIC
- ▶ valeurs de dates et heures :
 - DATE
- ▶ autres :
 - Text, booleen...



SQL

(Structured Query Language)

- ▶ Contient l'ensemble des instructions nécessaires à la spécification et à l'utilisation d'une base de données relationnelle.
- ▶ C'est un langage normalisé
- ▶ Il associe des mots clés et des clauses pour trouver et visualiser quasiment toutes les informations possibles
- ▶ C'est un langage très puissant, complexe mais qui peut être 'dangereux'!!
- ▶ Il est à la fois :
 - un langage d'interrogation de données (LID) : **SELECT**;
 - un langage de manipulation de données (LMD) : UPDATE, INSERT, DELETE ;
 - un langage de définition des données (LDD) : ALTER, CREATE, DROP;
 - un langage de contrôle des données et des utilisateurs (LCD) : GRANT, REVOKE.



Requête simple



La commande SELECT

- ▶ permet d'indiquer quelles colonnes, ou quelles expressions doivent être retournées par l'interrogation

- ▶ Syntaxe:

```
SELECT * FROM nom_table ;
ou
SELECT nom_col1, nom_col2, .. FROM nom_table ;
```



Clause Select

- ▶ Exemples :
 - ▶ Liste de tous les n° commandes
 - SELECT aadnaq FROM adqis;
 - ▶ Liste de tous les n° registres (n° inventaires)
 - SELECT * FROM conreg;
- ▶ * : tous les champs (colonnes) de la table
- ▶ nom_col1, nom_col2, : limite la sélection à certaines colonnes
- ▶ DISTINCT : suppression des doublons



Quelques fonctions de caractères

- ▶ modification de la casse
 - UPPER(chaine), LOWER(chaine)
- ▶ extraction de sous-chaine
 - SUBSTR(chaine, départ, longueur)
- ▶ position de sous-chaine
 - INSTR(chaine, départ, longueur)
- ▶ fonctions sur les chaînes
 - TRIM - RTRIM - LTRIM : enlève les blancs autour de la chaîne (ou un autre caractère)
- ▶ fonctions de dates
 - TO_CHAR : TO_CHAR(date, 'FORMAT') ; transformation date en caractères suivant le 'FORMAT' (exemple : 'DD/MM/YYYY')
 - SYSDATE : avoir la date et l'heure du système



Fonctions d'agrégation

- ▶ COUNT
 - compte les lignes d'une table ou d'un groupe de lignes
- ▶ SUM
 - total des valeurs d'une colonne numérique pour un nombre de lignes donné
- ▶ MIN et MAX
 - retourne la valeur minimale/maximale d'une colonne dans un groupe de lignes
- ▶ une fonction d'agrégation ne renvoie qu'une seule valeur !!
- ▶ GROUP BY
 - Regroupement
- ▶ ORDER BY
 - Toujours citer en dernier
 - Trier
 - ASC : ordre ascendant (par défaut)
 - DESC : ordre descendant
- ▶ HAVING (avec un regroupement)



Clause FROM

- ▶ donne la liste des tables participant à l'interrogation.
- ▶ Il est possible de lancer des interrogations utilisant plusieurs tables à la fois.



Exemples requêtes SELECT

- ▶ *Liste des lecteurs par ordre alphabétique*
 - `select leapel, lenomb, lefreg`
`from lector`
`order by leapel, lenomb`
- ▶ *Nombre de prêts actifs*
 - `select count(prbarc) from presta;`



Exemples requêtes SELECT

- ▶ *Liste des lecteurs par ordre alphabétique avec le nom en majuscule et leur date d'inscription*
 - `select upper(leapel),initcap(lenomb), to_char(lefreg,'DD/MM/YYYY')`
from lector
order by upper(leapel),initcap(lenomb)
- ▶ *Nombre de lecteurs par type de lecteur*
 - `select count(lenlec), lecolp`
from lector
group by lecolp



Clause WHERE et les opérateurs

- ▶ poser une condition sur les lignes via des opérateurs logiques
- ▶ comparaison : 6 opérateurs : =, <> ou !=, <, >, <=, >=
- ▶ BETWEEN/ NOT BETWEEN *valeur1* AND *valeur2* : rechercher des valeurs à l'intérieur d'une plage, bornes incluses (*entre ...et...*)
- ▶ IN / NOT IN (*ensemble_valeurs*) : comparer une valeur à une liste de valeurs spécifiées (dans)
- ▶ IS NULL/IS NOT NULL : savoir si la valeur est null (vide, non définie) ou pas (renseignée)



Clause WHERE et les opérateurs

- ▶ LIKE / NOT LIKE : comparer une valeur ressemblant au modèle donné (*comme*).
 - caractères jokers :
 - _ remplace 1 caractère exactement
 - % remplace une chaîne de caractères de longueur quelconque, y compris de longueur nulle
- ▶ conditions multiples : OR et AND
 - condition1 AND condition2
 - vrai si les 2 conditions sont vérifiées
 - condition1 OR condition2
 - vrai si l'une des 2 expressions est vérifiée
 - Priorité du AND sur le OR
 - Utilisation des parenthèses est conseillée



Exemples requêtes SELECT

- ▶ *Liste des exemplaires qui ne sont jamais sortis :*
 - `select contit, cobarc, cococl, cosign, cococs, cofrec, conpre`
`from copias`
`where conpre=0`
- ▶ *Liste des exemplaires les plus empruntés :*
 - `select conpre, cobarc`
`from copias`
`where conpre >10 order by conpre desc`



Exemples requêtes SELECT

- ▶ *Nombre d'exemplaires par type de document pour une succursale donnée:*
 - `select cococp, count(*)`
`from copias`
`where cocosu like 'BUL%'`
`group by cococp`
`order by cococp`
- ▶ *Liste des nouveaux titres depuis la rentrée universitaire :*
 - `select tintit, tititu, tiauto, tifsad`
`from titulo`
`where tibase='C'`
`and tifsad >= to_date('01/09/2013','DD/MM/YYYY')`
`order by tifsad desc`



Requête sur plusieurs tables



Jointure simple

- ▶ Effectuer une jointure, consiste à relier les deux tables par l'élément commun aux deux
- ▶ Permettre d'obtenir des résultats qui combinent les données de plusieurs tables
- ▶ La + simple des jointures est la jointure interne ; elle n'affiche que les lignes qui respectent le critère de jointure



Jointure simple (interne)

- ▶ Syntaxe :
 - SELECT les_cols_a_afficher FROM table1 , table2 WHERE table1.col=table2.col
 - ou
 - SELECT les_cols_a_afficher FROM table1 INNER JOIN table2 ON table1.col=table2.col
- ▶ *Liste des lecteurs avec leurs prêts :*
 - select lenlec, leapel, lenomb, prbarc from lector, presta where lenlec=prnlec



Jointure simple (interne)

- ▶ *Liste des titres qui ont été acquis il y a plus de 2 ans sur un compte budgétaire donné*
 - select tititu, tintit, aafped from titulo, adquis, adqdet where aanadq=adnadq and adntit=tintit and aafped < add_months(sysdate,-24) and adcocu like 'ART%' order by tintit desc
- ▶ *Compter le nombre de prêts par lecteur*
 - Select prnlec, leapel, lenomb, count(prbarc) nb_pret from presta, lector where prnlec=lenlec group by prnlec, leapel, lenomb [having count(prbarc)>10 (on ne retient que ceux qui ont plus de 10 prêts en cours)] order by leapel, lenomb



Jointure externe

- ▶ **LEFT JOIN** : retourne tous les enregistrements de la table de gauche (LEFT = gauche) même si la condition n'est pas vérifiée dans l'autre table.
- ▶ **RIGHT JOIN** : retourne tous les enregistrements de la table de droite (RIGHT = droite) même si la condition n'est pas vérifiée dans l'autre table.
- ▶ Remarque : ancienne syntaxe (+)



Jointure externe

- ▶ *Liste de tous les lecteurs avec ou sans prêt :*
 - `Select lenlec, leapel, lenomb, prbarc
from lector left join presta on lenlec=prnlec
order by prbarc desc`
- ▶ *Liste des exemplaires avec la date de retour de prêt le cas échéant pour une cote donnée :*
 - `Select cobarc, tititu, prbarc from copias
left join presta on cobarc=prbarc
inner join titulo on contit=tintit
where cosign like '930.26%'`



Requêtes imbriquées (sous-interrogations)

- ▶ consiste à exécuter une requête à l'intérieur d'une autre requête.
- ▶ Syntaxe :
 - `SELECT col1, col2... FROM nom_table
WHERE [Expression]
Opérateur (SELECT colA, colB
FROM nom_table2
WHERE [Expression2])`



Requêtes imbriquées

- ▶ *Liste des titres qui sont communs à 2 localisations données*
 - `select contit, tititu from titulo, copias`
`where contit=tintit`
`and cococl like '4%'`
`and contit in (select contit`
`from copias`
`where cococl like '5%')`
- ▶ *Le livre le plus emprunté*
 - `select cobar, conpre from copias`
`where conpre=(select max(conpre) from copias)`



Jointure combinatoire

- ▶ combiner le résultat de 2 requêtes ou plus
 - UNION: mettre en commun tous les enregistrements
 - INTERSECT (intersection): identifier les enregistrements similaires
 - MINUS (différence) : identifier les enregistrements appartenant à un ensemble mais pas à l'autre
- ▶ syntaxe :
`requete_1 { UNION | INTERSECT | MINUS } requete_2`
- ▶ même schéma pour requete_1 et requete_2
- ▶ même nombre de variables



Jointure combinatoire

- ▶ *Liste d'exemplaires ayants un mauvais TITN*
 - `select contit from copias`
`minus`
`select tintit from titulo`
- ▶ *Trouver les CB qui n'ont jamais été prêtés*
 - `select cobar from copias`
`where cobar not in (select prbar from presta`
`union`
`select phbar from prehis)`
`and cococp='C'`



Merci

Des questions ?

Gaëlle.perrin@upmf-grenoble.fr




Autres exemples de requêtes

»



Sur les exemplaires

- ▶ Liste des notices de la base CANC avec des ex. non pilonnés ou perdus
 - `select trim(tibase), tintit, trim(ticdus), trim(tititu), trim(tiauto), trim(tiedit)`
from titulo, copias
where tintit = contit
and tibase = 'N'
and costat not in ('P', 'X')
- ▶ Liste des exemplaires en dewey
 - `select cosign, cobarc`
from copias
where substr(cosign,4,1) in (',','')
and substr(cosign,1,1) between '0' and '9'



Sur les commandes

- Liste des commandes annulées sur une période donnée

```
select aanadq, aafped, aastat, aacosu, aafsmc
from acquis
where aastat='C'
and aafsmc between
to_date('01/06/2009','DD/MM/YYYY') and
to_date('30/06/2009','DD/MM/YYYY')
```



Sur les commandes

- Liste des commandes annulées du mois précédent

```
select trim(aacosu), trim(pvnomb), trim(aacocu), trim(adnadq),
to_char(aafped,'DD/MM/YYYY'), adntit, trim(tititu), adpedi,
adreci, adncanc, adfact, adesti, to_char(adfsmd,'DD/MM/YYYY')
from adqdet, acquis, titulo, provee
where adntit = tintit
and adnadq = aanadq
and aanprv = pvnprv
and adncanc > 0
and (aacosu='1' or aacosu ='24')
and to_char(trunc(adfsmd, 'J'), 'YYYYMM')=
to_char(trunc(sysdate - 1, 'J'), 'YYYYMM')
order by aanprv, aanadq
```



Sur les lecteurs

- Liste des lecteurs inactifs sur une période donnée : pas d'utilisation de la bibliothèque, pas de prêt depuis 1 an

```
select distinct leapel, lenomb, lenlec
from lector
left join presta on lenlec=prnlec
left join prehis on lenlec=phnlec
left join reserv on lenlec=renlec
where lefult < add_months(sysdate,-12)
and lefubi < add_months(sysdate,-12)
and prfult < add_months(sysdate,-12)
and phfdev < add_months(sysdate,-12)
order by leapel, lenomb
```



Sur les lecteurs

- ▶ Nombre de lecteurs TMP qui ont des prêts
 - `select count(*) from lector`
`where lecolp = 'TMP'`
`and lenlec in (select prnlec`
`from presta)`



Sur les lecteurs

- ▶ Nombre d'abonnés par âge et par sexe inscrits ou réinscrits au cours de l'année
 - `select count(*), letitu sexe,`
`to_char(sysdate, 'YYYY')-to_char(lefncac, 'YYYY') age`
`from lector`
`where lefren between`
`to_date('01/09/2013','DD/MM/YYYY') and`
`to_date('31/08/2014','DD/MM/YYYY')`
`or lefreg between`
`to_date('01/09/2013','DD/MM/YYYY') and`
`to_date('31/08/2014','DD/MM/YYYY')`
`group by to_char(sysdate, 'YYYY')-to_char(lefncac, 'YYYY'),`
`letitu`
`order by to_char(sysdate, 'YYYY')-to_char(lefncac, 'YYYY'),`
`letitu`



Sur les lecteurs

- ▶ Traitement LEL : Sélection des exemplaires et lecteurs à traiter
 - `select trim(lenlec), trim(lenomb), trim(leapel), trim(prcolp), trim(prcol1),`
`trim(prcol2), trim(prcol3), to_char(lefncad, 'DD/MM/YYYY'), trim(learpd),`
`trim(contit), trim(substr(tititu,1,20)), trim(prbarc), trim(prcocp),`
`trim(prcocl), trim(prcoce), trim(cosign), trim(cosigs),`
`to_char(prfpre, 'DD/MM/YYYY HH24:MI:SS'), to_char(prfult, 'DD/MM/YYYY`
`HH24:MI:SS'), trim(prnren), round(sysdate-prfdev), trim(lenlecp),`
`to_char(sysdate, 'DD/MM/YYYY HH24:MI:SS'),`
`to_char(sysdate, 'DD/MM/YYYY'), trim(prpsu), trim(prclas), trim(prorpr),`
`trim(prcosu), trim(prcoc1), trim(prlebi), trim(prcorn), trim(prcopt),`
`trim(prcomp), trim(prforc), trim(prcocs), trim(conpre), trim(lediso),`
`trim(leadul)`
`from copias, titulo, presta, lector`
`where prbarc = cobarc`
`and contit = tintit`
`and prnlec = lenlec`
`and prfdev < to_date('01/09/2011','DD/MM/YYYY')`
`and lecolp not in ('SRV', 'PEB')`
`order by prcosu, lenlec, contit, cobarc`



Sur les prêts

- ▶ Prêts en retard
 - ```
select cobarc, tititu, prnlec, lenomb, leapel
from copias, titulo, presta, lector
where prbarc = cobarc
and contit = tintit
and prnlec = lenlec
and prfdev < sysdate
and lecolp not in ('SRV', 'PEB')
order by lecolp, leapel, lenomb, prfdev, tititu
```
- ▶ Nombre de prêts par section selon le type d'exemplaire
  - ```
select count(prbarc), cococl, cococp
from copias, presta
where cobarc = prbarc
group by cococl, cococp
```



Sur les réservations

- ▶ Liste des exemplaires en fin de réservation
 - ```
select to_char(r1.renlec) "CB lect", ltrim(substr(leapel,1,25)) "Nom lecteur",
to_char(r1.rebarc) "CB exemp", ltrim(substr(cosign,1,20)) "Cote",
ltrim(substr(tititu,1,50)) "Titre"
from reserv r1, lector, copias, titulo
where r1.recosu='1'
and r1.renlec=lenlec
and r1.rebarc=cobarc
and contit=tintit
and to_date(to_char(r1.refin,'DD/MM/YYYY'))=
to_date(to_char(sysdate,'DD/MM/YYYY'))
and r1.refcom is not null
and r1.rebarc not in (select r2.rebarc
from reserv r2
where r2.rebarc=r1.rebarc
and to_date(to_char(r2.refin,'DD/MM/YYYY')) >
to_date(to_char(sysdate,'DD/MM/YYYY'))
and r2.refcom is null)
order by "Nom lecteur"
```




---

---

---

---

---

---

---

---

---

---