



Curseurs

SQL avancé

Fonctionnement

- ◉ Déclaration du curseur (avec son SELECT)
- ◉ Ouverture du curseur
- ◉ Parcours du contenu
- ◉ Fermeture du curseur

Déclaration

- `CURSOR nom_du_curseur IS`
`SELECT ;`

Ouverture

- OPEN nom_du_curseur ;
- Impossible de manipuler le curseur non ouvert
- On peut ne pas l'ouvrir juste après sa déclaration

Récupération des lignes

- `FETCH nom_du_curseur INTO variable1,variable2 ;`
 - Il doit y avoir autant de variables que de colonnes dans le `SELECT`
- `FETCH nom_curseur INTO enregistrement;`
 - On peut stocker le contenu de la ligne dans 1 seul enregistrement déclaré comme suit :
`enregistrement nom_curseur%TYPE;`

Fermeture

- CLOSE nom_curseur;

Attributs d'un curseur

- nom_curseur%ISOPEN
 - Permet de savoir si le curseur est ouvert (TRUE/FALSE)
- nom_curseur%FOUND
 - Permet de savoir si le dernier FETCH du curseur a retourné au moins une ligne (TRUE/FALSE)
- nom_curseur%NOTFOUND
 - Inverse du précédent

Attributs d'un curseur

- nom_curseur%ROWCOUNT
 - Nombre de FETCH ayant renvoyé des informations.
 - Appelé avant le premier FETCH, il renvoie 0

Mise à jour de données

- Dans le cadre d'une mise à jour des données (UPDATE) il est important de verrouiller les lignes à modifier.

```
DECLARE  
CURSOR nom_curseur IS  
SELECT champ1,champ2 FROM table  
FOR UPDATE;
```

FOR UPDATE : verrouille la ligne entière

FOR UPDATE champ1,champ2 : verrouille
uniquement certains champs de la ligne

Déclaration implicite

```
DECLARE  
CURSOR nom_curseur IS  
SELECT champ1,champ2 FROM table;  
variable1 table.champ1%TYPE;  
variable2 table.champ2%TYPE;  
BEGIN  
FOR maligne IN moncurseur LOOP  
...  
END LOOP;  
END;
```

Ici plus besoin de OPEN, FETCH, CLOSE