

# TP

## Optimisation de requêtes

Si vous ressentez le besoin de lire la référence **ORACLE** : <https://docs.oracle.com/en/database>.

Quelques liens plus ciblés

La description globale de l'optimisation est là (attention, ne vous perdez pas dans la documentation) :

[http://docs.oracle.com/cd/B28359\\_01/server.111/b28274/ex\\_plan.htm](http://docs.oracle.com/cd/B28359_01/server.111/b28274/ex_plan.htm).

La description des colonnes de la table PLAN\_TABLE est là :

[http://docs.oracle.com/cd/B28359\\_01/server.111/b28274/ex\\_plan.htm#i18300](http://docs.oracle.com/cd/B28359_01/server.111/b28274/ex_plan.htm#i18300)

L'utilisation de EXPLAIN PLAN est là :

[http://docs.oracle.com/cd/B28359\\_01/server.111/b28274/ex\\_plan.htm#i17492](http://docs.oracle.com/cd/B28359_01/server.111/b28274/ex_plan.htm#i17492)

La description nécessaire à la bonne lecture d'un plan d'exécution est là :

[http://docs.oracle.com/cd/B28359\\_01/server.111/b28274/ex\\_plan.htm#i23461](http://docs.oracle.com/cd/B28359_01/server.111/b28274/ex_plan.htm#i23461)

### Conseils

1. Avant de vous connecter, créez un répertoire de travail pour cet enseignement et travaillez à partir de ce répertoire.
2. Utilisez un éditeur de texte pour concevoir vos requêtes puis effectuez des copier/coller dans l'interpréteur **ORACLE** (**sqlplus** ou **sqldeveloper**).
3. Une fois le script **SQL** au point, sauvegardez le fichier avec l'extension **.sql** pour une réutilisation future ou un rendu.

Comme nous l'avons vu en cours, il peut être fastidieux de rédiger une requête puis de consulter le plan d'exécution. Heureusement, il est possible de demander à l'interpréteur de requêtes **ORACLE** d'afficher le plan d'exécution réalisé pour chaque requête : **set AUTOTRACE ON EXP**.

### Récupération des données

Pour illustrer l'optimisation de requêtes, nous allons travailler sur le schéma suivant qui représente une gestion simplifiée de magasins (clé primaire soulignée, clé étrangère en **GRAS**) :

PRODUIT	( <u>IdP</u> , Libelle, PrixAchat, <b>CodeCat</b> )
CATEGORIE	(CodeCat, Designation, <b>CodeCatGen</b> )
MAGASIN	( <u>IdM</u> , Nom, Type, <b>IdL</b> )
LIEU	( <u>IdL</u> , Nomlieu, <b>IdLrattach</b> )
VENTE	( <b>IdP</b> , <b>IdM</b> , DateV, Quantite, PrixVenteUnitaire)

Afin de faciliter le lancement du TP, je vous fournis les fichiers suivants :

[01 dropDB.sql](#) : suppression de la base de données du TP.

[02 createDB.sql](#) : création la base de données.

[03 insertData.sql](#) : insertion des données dans la base de données.

## Plans d'exécution

Pour les plans d'exécution, vous utilisez la table `Vente` et une autre table pour les jointures.

Requêtes à exécuter, à comparer et expliquer les plans d'exécution obtenus (partie 1/2) :

1. `SELECT IdP FROM Vente;`
1. `SELECT IdP FROM Produit;`
2. `SELECT Quantite FROM Vente;`
3. `SELECT DISTINCT Quantite FROM Vente;`
4. `SELECT Quantite FROM Vente WHERE Quantite = <cst>;`
5. `SELECT IdP FROM Vente WHERE IdP = <cst>;`
6. `SELECT COUNT( Quantite ) FROM Vente;`
7. `SELECT COUNT( Quantite ) FROM Vente GROUP BY Quantite;`

## Comparer et expliquer les différents plans en présence de vues.

8. Rédiger une requête de votre choix pour laquelle vous afficherez le plan d'exécution.
9. Créer la vue correspondante à votre requête (`CREATE VIEW...`).
10. Afficher le plan d'exécution de la requête `SELECT * FROM VueCreee`.
11. Rédiger une requête de sélection et projection qui utilisent la vue créée.
12. Comparer les tris plans de requêtes obtenus.

## Comparer et expliquer les différents plans en présence de jointure.

Jointures entre la clé primaire et une clé étrangère non indexée

13. Rédiger une requête réalisant une jointure prédicative **dans la clause WHERE**.
14. Rédiger la même requête réalisant une jointure prédicative **dans la clause FROM**.
15. Rédiger la même requête réalisant une jointure imbriquée (**une sous-requête**).

Jointures entre la clé primaire et une clé étrangère indexée

16. Créer un index sur la clé étrangère choisie (`CREATE INDEX ...`)
17. Rédiger une requête réalisant une jointure prédicative (dans la clause `WHERE`).
18. Rédiger une requête réalisant une jointure prédicative (dans la clause `FROM`).
19. Rédiger une requête réalisant une jointure imbriquée (sous requête).
20. Supprimer l'index créé à la question 16 (`DROP INDEX...`)

Requêtes à exécuter, à comparer et expliquer les plans d'exécution obtenus (partie 2/2) :

21. `SELECT Quantite, COUNT( Quantite ) FROM Vente GROUP BY Quantite;`
22. `SELECT Quantite, COUNT( DISTINCT Quantite ) FROM Vente GROUP BY Quantite;`
23. `SELECT Quantite, COUNT( PrixVenteUnitaire ) FROM Vente GROUP BY Quantite;`
24. `SELECT COUNT( Quantite ), COUNT( DISTINCT PrixVenteUnitaire ) FROM Vente;`
25. `SELECT COUNT( Quantite ), COUNT( PrixVenteUnitaire ) FROM Vente GROUP BY Quantite, PrixVenteUnitaire;`