

## Oracle : Introduction complète à Oracle

### Objectif :

Une formation Oracle complète pour découvrir tous les concepts du système et les différentes façons de les utiliser concrètement tout en disposant d'une vue claire sur l'architecture globale d'ORACLE.

A l'issue de la formation vous serez capable de :

- Concevoir efficacement une base de données Oracle (modèle relationnel, MCD)
- Savoir mettre en œuvre une base Oracle (tables, contraintes d'intégrités)
- Comprendre le fonctionnement global d'Oracle (tablespaces, SGA, ?)
- Savoir dialoguer avec la base en utilisant le langage SQL
- Optimiser vos requêtes SQL
- Maîtriser votre environnement de travail sous Oracle
- Sécuriser l'accès aux données
- Comprendre le concept de transaction et des verrous

### Participants :

Cette formation Oracle s'adresse à toute personne participant à un projet autour d'Oracle (concepteur, développeur, chef de projet, futur administrateur).

### Pré-requis :

Il est nécessaire de connaître au moins un système d'exploitation de type graphique.

### Travaux pratiques :

Chaque participant va construire de façon progressive son environnement de développement. A partir d'un modèle conceptuel de données réalisé en TP les participants prennent conscience de l'origine des données dans une base Oracle et des liens qui les unissent. Ensuite on passe à la création des tables qui seront utilisées tout au long du stage pour les manipulations SQL ainsi que pour la sécurisation des données.

**Durée : 5 jours (35 heures)    Référence : BOPM**

### Contenu de la formation :

#### La conception d'une base Oracle

##### *Concevoir une base de données*

- Comment concevoir l'organisation des données ?
- Besoin d'un formalisme pour les représenter (Modèle Conceptuel des Données)
- Vocabulaire (entité, association, cardinalités, ?)
- L'impact des choix du concepteur
- Exemples de modèles faux ou mal conçus

- Exemples de modèles rencontrés dans des projets professionnels

### ***Le modèle relationnel***

- Pourquoi parle-t-on de base de données « relationnelle » (dépendances) ?
- Se rapprocher du schéma de la base (MLD : modèle logique des données)

### ***Le modèle physique***

- Traduire un modèle logique en base Oracle (création de tables)
- Qu'entend-on par contraintes d'intégrités (primary key, foreign key, not null,...)?
- Pourquoi doit-on typer les données (integer, varchar, date, ?) ?

### ***Remplir une base***

- Insérer des données dans les tables (insert)
- L'intérêt des séquences
- Attention aux violations de contraintes

## **L'environnement de travail**

### ***Les produits Oracle***

- Rôle d'ORACLE par rapport au système d'exploitation
- Les différents produits Oracle (Server, Spatial, Application Server, ?)
- Comparaison avec d'autres SGBDR (SQL\*Server, MySQL, Access, ?)

### ***Les applications utilisées***

- Comment se connecter à une base Oracle (tnsnames.ora) ?
- Spécificités de l'outil SQL\*Plus
- Exemples d'éditeurs pour les scripts SQL
- Mise en page des résultats (set, format, ?)
- Stockage des résultats (spool)
- Construire des requêtes SQL depuis un navigateur ( iSQL)

### ***Obtenir des informations sur la base***

- Philosophie du « rangement » des informations dans Oracle (catalogue)
- Quelques questions classiques (clés, contraintes, schéma d'une table, ?)
- Changer d'utilisateur ou de base

## **Le langage SQL**

### ***Extraction simple des données***

- Forme générale d'une requête (select ? from ? where)
- Filtrage de base (where, like, between, ?)
- Spécificités du SQL pour Oracle

### ***Mettre à jour les données***

- Les instructions insert, update, et delete

- L'impact des contraintes d'intégrité (cascade)

### **Les vues**

- Pourquoi associer une requête SQL à une vue ?
- Différences vue/table ?
- Et les contraintes d'intégrité ?

### **Requêtes plus complexes**

- Extraire les données de plusieurs tables (jointure)
- Trier les résultats (order by)
- Opérations ensemblistes (union, minus, intersect)
- Regroupement des données (group by)
- Les fonctions d'agrégats (somme, moyenne, nombre, ?)
- Jointures multiples
- Dans quel cas utiliser l'auto-jointure ?
- Les tests en SQL (where exists ?)

### **Le PL/SQL**

- Les apports du PL/SQL par rapport au SQL
- Rôle des triggers (déclencheurs)

## **L'architecture globale d'Oracle**

### **Comment Oracle stocke les données ?**

- La séparation logique/physique (tablespace)
- Les fichiers de données (datafile)
- La finesse du stockage dans Oracle (extension, bloc, segment)
- Comprendre la saturation d'un tablespace

### **Rôle des fichiers journaux**

- Que contient un fichier de log ?
- Comment Oracle les utilise ?

### **Le fonctionnement d'Oracle en mémoire**

- Pourquoi Oracle doit structurer la mémoire ?
- En quoi consiste la construction de la SGA (System Global Area) ?
- Se connecter à une base ou une instance Oracle ?
- Rôle de chaque processus (dbwr, lgwr, ?)

## **Les transactions**

### **Principes**

- Qu'est qu'une transaction ?
- Comportement standard d'Oracle
- Le rôle de l'utilisateur (commit, rollback)

- La problématique des accès concurrents

### ***Verrous et accès concurrents***

- Exemples de blocages
- Comment ne pas bloquer les autres (commit, type de transaction?) ?

## **La sécurité des données**

### ***Utilisateurs***

- Qu'est-ce qu'un compte utilisateur Oracle ?
- Exemple de création de compte
- Paramétrage de base d'un compte utilisateur (profil, droits)

### ***Protéger les données***

- Les droits proposés par Oracle sur les tables
- Mise en oeuvre (grant, revoke)
- L'utilisation des vues pour masquer certaines données

## **L'optimisation des requêtes SQL**

### ***Comment Oracle exécute une requête SQL ?***

- Les différents choix possibles pour Oracle
- Obtenir des informations sur une requête (plan d'exécution)
- Interpréter le temps d'exécution
- Quelques statistiques normales selon le volume des tables, le type de requête, ?

### ***Optimiser***

- Les règles de l'optimiseur
- Créer des index
- Générer des statistiques
- Pourquoi les index peuvent nuire aux performances ?
- Changer le plan d'exécution d'une requête