**LE SQL = "LANGAGE DE REQUETES STRUCTURE"**

**A – Langage de manipulation de données**

1 - Ajouter des **enregistrements**

INSERT INTO nom de la table (nom des champs à remplir)

VALUES (valeurs que doivent prendre les champs, séparés par des ,)

Autre possibilité :

INSERT INTO nom de la table

VALUES (valeurs à prendre dans l'ordre où les champs apparaissent dans la relation. Si des champs n'ont pas de valeur, mettre "" ou NULL, selon la structure des champs

2 – Mise à jour des enregistrements

UPDATE nom de la table

SET nom du champ à modifier = nouvelle valeur à prendre

WHERE critère de sélection

Si les conditions concernent des champs de plusieurs tables, la syntaxe sera :

UPDATE nom de la table mise à jour, nom des autres tables

SET nom du champ à modifier = nouvelle valeur à prendre

WHERE jointures

AND critères de sélection

3 – Suppression d'enregistrements

DELETE nom de la table.\* (DELETE \* marche parfois !)

from nom de la table

WHERE …. S'il y a des critères de sélection

Une requête action change les données d'une table et est irréversible.

**Application 1 :**

Extrait de la table Employe

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **mat** | **nomEmploye** | **fonction** | **date**  **Embauche** | **Salaire**  **annuel** | **Commission**  **annuelle** | **numServ**  **Employe** |
| 001 | Dupond | Commercial | 14/04/1992 | 25000 | 2500 | 20 |
| 002 | Martin | Commercial | 20/01/2005 | 20000 |  | 10 |
| 003 | Dupond | Ingenieur | 05/03/2000 | 15000 |  | 30 |
| … |  |  |  |  |  |  |

Extrait de la table Service Extrait de la table Produit

|  |  |  |
| --- | --- | --- |
| **NumService** | **NomService** | **Lieu** |
| 10 | Recherche | Lyon |
| 20 | Vente | Marseille |
| 30 | Direction | Lyon |
| 40 | Fabrication | Nantes |
| 50 | SAV | Nantes |

|  |  |
| --- | --- |
| **RefProduit** | **Désignation** |
| A1 | Carton 6 primeurs |
| A2 | Carton 12 primeurs |
| B1 | Carton 6 bourgogne |
| B2 | Carton 12 bourgogne |

Extrait de la table Commander

|  |  |  |
| --- | --- | --- |
| **Mat** | **RefProduit** | **Quantité** |
| 001 | A1 | 3 |
| 002 | B2 | 2 |
| 002 | A2 | 1 |
| 001 | B1 | 5 |

Soit le modèle relationnel :

*Employe (mat, nomEmploye, fonction, dateEmbauche, salaire, commission, numServEmploye,) ou #numService*

*Service (numService, nomService, lieu)*

*Produit (RefProduit, Désignation)*

*Commander (Mat, RefProduit, Quantité)*

Ecrivez les requêtes suivantes :

1. *Enregistrement d'un nouvel employé : M. Jean, comptable, matricule 010, embauché le 15/08/2006 dans le service 30 pour un salaire annuel de 25 000, sans commission.*
2. *Déménagement du service 20 à Rennes*
3. *Augmentation de 100 € de tous les salaires annuels*
4. *Suppression de tous les administratifs du service 30*

**B – Langage de définition de données**

On touche à la structure de la table (ne pas confondre colonnes et enregistrements).

Va permettre de créer une table (ou vue ou des contraintes), modifier une colonne, supprimer des tables (vues, contraintes)

1 – Création d'une table

Pour chaque colonne créée, il faut préciser le type de données que le champ va contenir.

Les différents types de données sont les suivants :

1. type numérique

-nombres entiers : SMALLINT(sur 2 octets), INTEGER (sur 4 octets)

- nbres décimaux avec un nombre fixe de décimales

- numériques non exacts à virgule flottante : REAL (simple précision avec au moins 7 chiffres significatifs), DOUBLE PRECISION ou FLOAT (double précision avec au moins 14 chiffres significatifs)

- le type BIT permet de ranger une valeur booléenne.

1. type chaîne de caractères

Les constantes chaînes de caractères sont entourées par des apostrophes (')

- CHAR (longueur) : chaîne de caractère de longueur constante.

- VARCHAR (longueur) pour les colonnes qui contiennent des chaînes de longueur variable.

c) type temporel

- DATE : mm/jj/aaaa

- TIME : hres, mn, secondes

d) type binaire

Pour enregistrer des données telles que les images et les sons.

CREATE (nom des champs, type, taille …séparés par des ,

PRIMARY KEY nom de la clé primaire)

Si la table doit contenir une ou des clés étrangères (qui sont donc clés primaires d'autres tables et qui ont déjà été créées), on ajoute

FOREIGN KEY (nom de la clé 2daire) REFERENCES (nom de la table) contenant la clé 2daire (nom de la clé 2daire))

2 – Modification de la structure d’une table

ALTER TABLE permet la modification de la structure d'une table :

ALTER TABLE nom\_de\_la\_table

ADD nom\_du\_champ type (taille)

Ajoute une colonne (un nouveau champ)

ALTER TABLE nom\_de\_la\_table

DROP nom\_du\_champ

Supprime un champ de la table

ALTER TABLE nom\_table MODIFY nom\_champ [taille,Type]

ALTER TABLE Service MODIFY désigantion CHAR(50)

Modifier la taille du champs désigqntion

ALTER TABLE nom\_de\_la\_table ADD FOREIGN KEY (nom de la clé 2daire) REFERENCES nom\_de\_la\_table\_contenant\_la\_clé primaire (nom de la clé primaire).

*Ajouter le champ budget à la table Service*

ALTER TABLE Service

ADD budget decimal (7,2) CHECK (budget>0)

3 – Suppression d'une table

DROP TABLE nom\_table

*Suppression de la table Service*

DROP TABLE Service

1. *Création de les tables Employé, Service, Produit, Commander*
2. *Modifier la taille du champ Désignation de la table produit.*
3. Ajouter le champ Adresse dans la table Employé.
4. *Ajouter le champ budget à la table Service*
5. Supprime le champ Date embauche de la table Employé

**C – Langage de contrôle de l'accès aux données**

Les BDD comptent souvent plusieurs utilisateurs, notamment lorsqu'elles sont partagées en réseau, ces utilisateurs pouvant y accéder simultanément.

Ils n'ont pas forcément les mêmes besoins : certains peuvent, par exemple, avoir besoin de modifier des données dans une table tandis que d'autres ne l'utiliseront que pour la consulter.

Ainsi, il est possible d'associer des permissions (droits, privilèges) pour chaque profil d'utilisateur.

Seul l'utilisateur qui a créé un élément (table ou vue) a la possibilité d'accorder ou de retirer des droits sur cet élément.

Le langage SQL permet d'effectuer ces opérations grâce à 2 clauses :

* GRANT permet d'accorder des droits à un utilisateur (SELECT, INSERT, UPDATE, DELETE, ALTER, ALL = tous les droits précités)
* REVOKE permet de retirer des droits.
* CREATE USER userXXa IDENTIFIED BY 'userXXa';

**Exemple**

GRANT UPDATE (cotisationReglee)

ON Membres

TO Jean, Pierre

Jean et Pierre ont le droit de mettre à jour la donnée "cotisationReglee) de la table Membres.

GRANT INSERT, SELECT

ON Membres

TO PUBLIC

Le droit d'ajouter et de consulter des enregistrements de la table Membres est accordé à tous les utilisateurs.

GRANT ALL

ON Membres

TO Jean

WITH GRANT OPTION

Jean obtient tous les privileges (ALL) sur la table Membres et a même le droit d'accorder des privilèges à d'autres utilisateurs.

REVOKE UPDATE (cotisationReglee)

ON Membres

FROM Jean, Pierre

Le droit de mettre à jour la donnée (cotisationReglee) de la table Membres est retiré à Jean et Pierre.

REVOKE ALL

ON Membres

FROM PUBLIC

Tous les droits (ALL) sont retires à tous les utilisateurs sur la table Membres

REVOKE GRANT OPTION FOR DELETE

On Membres

FROM Jean

Jean n'a plus la possibilité d'accorder le droit d'effacer (DELETE) à d'autres utilisateurs.

**Exercice : Contrôle de données**

On vous donne le schéma relationnel d'une société de ventes par correspondance :

Fournisseur (codeFourn, raisonSocialeFourn)

Produit (refInterneProd, designProd)

Catalogue (numCat, designCat, dateDebCat, dateFinCat)

Proposer (codeFourn, refInterneProd, refFournProd, prixFourn)

Présenter (numCat, refInterneProd, prixVenteHT)

Ecrivez les requêtes suivantes :

a) Tous les utilisateurs ont le droit de consulter toutes les tables

b) Marine a le droit de créer des catalogues

c) Marine a le droit de modifier le contenu des catalogues et peut déléguer ce droit.

d) Thomas et Pierre peuvent mettre à jour les prix proposés par les fournisseurs

e) Sylvie peut supprimer des propositions de fournisseurs qui ne sont plus valables.

f) On lui retire ce droit.