



M16 : SGBD 1

Langage de Définition de Donnée(LDD)

Formateur : Driouch (cfmoti.driouch@gmail.com)

Etablissement : OFPPT/CFMOTI 15/11/2012

<http://www.ista-ntic.net/>

Plan

- Introduction
- Objets et Dénomination d'une BD
- Type de donnée
- Create (DataBase, Table)
- Alter et Truncat Table
- Gestion des contraintes
- Sécurisé une BD (Vues, Autorisation)
- Rôles, GRAINT, DENY, REVOKE

Introduction

- LDD est le sous-ensemble du langage SQL, pour créer et modifier des objets de base de données, en particulier les tables.
- Le LDD nous permet de :
 - Créer des tables
 - Décrire les différents types de données utilisables pour les définitions de colonne
 - Modifier la définition des tables
 - Supprimer, renommer et tronquer une table

DRIOUCH B.

3

Objets d'une Base de Données

- Une base de données peut contenir de nombreuses structures de données.
- Chaque structure doit être prédéfinie lors de la conception de la base de données pour pouvoir être créée durant la phase de construction de la base.
- Vous avez la possibilité de créer des tables à tout moment, y compris lorsque la base de données est déjà en cours d'utilisation.

Objet	Description
Table	Unité de stockage élémentaire, composée de lignes et de Colonnes
Vue	Représente de manière logique des sous-groupes de données
Index	Améliore les performances de certaines requêtes

DRIOUCH B.

4

Conventions de Dénomination

- Utilisez des noms significatifs.
- Un nom :
 - Doit commencer par une lettre
 - Peut comporter de 1 à 249 caractères
 - Ne peut contenir que les caractères A à Z, a à z, 0 à 9, (_ , \$, et # , et espaces déconseillés)
 - Ne doit pas porter le nom d'un autre objet appartenant au même utilisateur
 - Ne doit pas être un mot réservé à SQL Server

DRIOUCH B.

5

L'Ordre CREATE DATABASE

Use Master
Go

CREATE DATABASE Emploi

ON PRIMARY
(NAME = N'emploi',
FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL.1\MSSQL\DATA\emploi.mdf' ,
SIZE = 3072KB ,
MAXSIZE = 20Mb,
FILEGROWTH = 1024KB)

LOG ON
(NAME = N'emploi_log',
FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL.1\MSSQL\DATA\emploi_log.ldf' ,
SIZE = 1024KB ,
MAXSIZE = 10Mb,
FILEGROWTH = 10%)
GO

DRIOUCH B.

6

L'Ordre CREATE TABLE

- Pour créer une table, l'utilisateur doit disposer du privilège CREATE TABLE et d'un espace de stockage dans laquelle il pourra créer des objets.
- L'administrateur de base de données utilise des ordres du LCD (langage de contrôle des données) pour accorder des privilèges aux utilisateurs.

- Syntaxe :

CREATE TABLE [*schema*.]*table*

(*column1 datatype* [DEFAULT *expr*], *column2 datatype*, ...)

- *schema* nom du schema
- *table* nom de la table
- *column1,column2* nom de la colonne
- *datatype* type de données et longueur de la colonne
- DEFAULT *expr* spécifie une valeur par défaut

- Exemple:

- Use emploi
- Go
- CREATE TABLE Auditeur(no_auditeur int, nom_auditeur varchar(30))
- Go

(Go: valider une instruction avant de passé au suivante)

DRIOUCH B.

7

L'Ordre CREATE TABLE

- Références à un schéma

Un *schéma* est une collection d'objets. Les schémas sont des structures logiques qui font directement référence aux données de la base.

Un schéma permet de regrouper les objets afin de les traiter comme une entité unique pour leur attribuer des droits.

Create SCHEMA *schema_name*;

Alter schema *schema_name* TRANSFER T1;

DROP SCHEMA *schema_name*

DRIOUCH B.

8

L'Ordre CREATE TABLE

■ L'Option DEFAULT

On peut déclarer une valeur par défaut pour une colonne en utilisant l'option DEFAULT.

Cette option empêche l'insertion de valeurs NULL dans une colonne lors de l'ajout d'une ligne qui ne comporte pas de données pour cette colonne.

La valeur par défaut peut être un littéral, une expression ou une fonction SQL telle que DATE() et USER

■ Expression colonne calculée

Expression définissant la valeur d'une colonne calculée. Une colonne calculée est une colonne virtuelle qui n'est pas stockée physiquement dans une table.

- (Column1 as (Formule avec d'autre colonne))

DRIOUCH B.

9

Types de Données

■ Les types de données dans SQL Server

Les types de données de base pour les valeurs d'une colonne proposés par le système :

bigint	binary	bit	char	datetime
decimal	float	image	int	money
nchar	ntext	numeric	nvarchar	real
smalldatetime	smallint	smallmoney	varchar	text
timestamp	tinyint	varbinary	uniqueidentifier	xml

- L'affectation d'un type de données à un objet définit quatre attributs de l'objet :
 - le type de données contenues par l'objet
 - la longueur ou la taille de la valeur stockée
 - la précision du nombre (types de données numériques uniquement)
 - l'échelle du nombre (types de données numériques uniquement).

DRIOUCH B.

10

Types de Données

Valeurs numériques exactes		Stockage
bigint	De -2^{63} (-9 223 372 036 854 775 808) à $2^{63}-1$ (9 223 372 036 854 775 807)	Huit octets
int	De -2^{31} (-2 147 483 648) à $2^{31}-1$ (2 147 483 647)	Quatre octets
smallint	De -2^{15} (-32 768) à $2^{15}-1$ (32 767)	Deux octets
tinyint	De 0 à 255	Un octet
bit	Données de type entier qui peuvent prendre la valeur 1, 0 ou NULL. Les valeurs de chaînes TRUE et FALSE peuvent être converties en bit : TRUE est converti en 1 et FALSE en 0	1 ou 2 octets
Decimal, numeric	Types de données numériques ayant une précision et une échelle fixes. decimal[(p[, s])] et numeric[(p[, s])]	
money	-922 337 203 685 477,5808 à 922 337 203 685 477,5807	8 octets
smallmoney	-214 748,3648 à 214 748,3647	4 octets

DRIOUCH B.

11

Types de Données

Valeurs numériques approximatives		
float [(n)]	- 1,79E+308 à -2,23E-308, 0 et 2,23E-308 à 1,79E+308	Selon la valeur de n
real	- 3,40E + 38 à -1,18E - 38, 0 et 1,18E - 38 à 3,40E + 38	4 octets
Date et heure		
Datetime	Du 1er janvier 1753 au 31 décembre 9999	8 octets
smalldatetime	Du 1er janvier 1900 au 6 juin 2079	4 octets
Chaînes de caractères		
[(n)]	d'une longueur fixe de n octets, n doit être compris entre 1 et 8000	Entre 1 et 8000 octets
varchar[(n max)]	Données de type caractère non-Unicode d'une longueur variable. n doit être compris entre 1 et 8000. max indique que la taille maximale de stockage est égale à $2^{31}-1$ octets	Selon n
text	Données non-Unicode de longueur variable figurant dans la page de codes du serveur et ne pouvant pas dépasser en longueur $2^{31} - 1$ (2 147 483 647) caractères	Le stockage est tout de même de 2 147 483 647 octets

DRIOUCH B.

12

Types de Données

Chaînes de caractères Unicode		
char	Données de type caractères Unicode de longueur fixe de n caractères. n doit être compris entre 1 et 4 000	La taille de stockage, en octets, est le double de n.
Nvarchar[(n max)]	Données de type caractères Unicode de longueur variable. n peut être compris entre 1 et 4 000. max indique que la taille de stockage maximale est de $2^{31}-1$ octets.	La taille de stockage, en octets, est le double du nombre de caractères entrés plus 2 octets
ntext	Données Unicode de longueur variable ne pouvant pas dépasser $2^{30} - 1$ caractères (c'est-à-dire 1 073 741 823).	La taille de stockage, en octets, est le double du nombre de caractères entrés

DRIOUCH B.

13

Types de Données

Chaînes binaires		
binary [(n)]	Données binaires de longueur fixe de n octets, où n est une valeur comprise entre 1 et 8 000	L'espace mémoire occupé est de n octets.
Varbinary[(n max)]	Données binaires de longueur variable. n est une valeur comprise entre 1 et 8 000. max indique que l'espace mémoire maximal occupé est de $2^{31}-1$ octets	La taille mémoire est la longueur réelle des données entrées, plus deux octets
image	Données binaires de longueur variable occupant de 0 à $2^{31} - 1$ (2 147 483 647) octets.	

Autres types de données	
cursor	timestamp
sql_variant	Uniqueidentifier
table	xml

DRIOUCH B.

14

Ordre ALTER TABLE

- Utilisez l'ordre ALTER TABLE pour :
 - Ajouter une colonne
 - Modifier une colonne existante
 - Définir une constraints pour une colonne
 - Supprimer une colonne

ALTER TABLE *ntable*

ADD *ncolumn datatype* [DEFAULT *expr* [with nocheck]]

ALTER TABLE *ntable*

ALTER COLUMN *ncolumn datatype* [DEFAULT *expr*]

Exec sp_rename '*nTable.nColonne*', '*Newname*', 'COLUMN'

ALTER TABLE *ntable*

DROP COLUMN *nColonne*

Exec sp_help *nObjet* :Infos sur l'objet, table, vues ...

DRIOUCH B.

15

Ordre ALTER TABLE

a) Ajout

ALTER TABLE Auditeur

ADD Moyenne Decimal(4,2);

La nouvelle colonne est placée à la fin.

b) Modification

Vous pouvez modifier le type de données, la taille d'une colonne.

ALTER TABLE Auditeur

ALTER COLUMN Moyenne Decimal(3,2);

c) Suppression

ALTER TABLE Auditeur

DROP COLUMN Moyenne;

DRIOUCH B.

16

Vider une table

■ Vider une Table

TRUNCATE TABLE est un autre ordre LDD, qui permet de supprimer toutes les lignes d'une table tout en libérant l'espace utilisé pour stocker cette table.

TRUNCATE TABLE *ntable*;

où : *ntable* est le nom de la table

L'ordre TRUNCATE TABLE ne peut être annulé.

L'ordre DELETE quant à lui supprime aussi les lignes d'une table, mais il ne libère pas l'espace de stockage.

Suppression de Table

■ Suppression de Tables

L'ordre DROP TABLE supprime la définition d'une table, Lorsque vous supprimez une table, la base de données perd toutes les données de la table ainsi que tous les index associés.

Syntaxe :

DROP TABLE Auditeur

L'ordre DROP TABLE est irréversible.

Commande de M. A. J. (LMD)

Les commandes de mise-à-jour (insertion, destruction, modification) sont considérablement plus simples que les requêtes select.

■ Insertion

L'insertion s'effectue avec la commande INSERT dont la syntaxe est la suivante :

```
INSERT INTO R( A1, A2, ... An) VALUES (v1, v2, ... vn)
```

■ Exemple :

```
INSERT INTO Client (id, nom, prenom)
VALUES (40, 'Moriarty', 'Dean')
```

Insertion multiple :

```
INSERT INTO Sites (lieu, region)
SELECT lieu, region FROM Station
```

Bien entendu le nombre d'attributs et le type de ces derniers doivent être cohérents.

DRIOUCH B.

19

Commande de M. A. J. (LMD)

■ Destruction

- La destruction s'effectue avec la clause DELETE dont la syntaxe est :

```
DELETE FROM R
[WHERE condition]
```

- Exemple : destruction de tous les clients dont le nom commence par 'M'.

```
DELETE FROM Client
WHERE nom LIKE 'M%'
```

■ Modification

- La modification s'effectue avec la clause UPDATE. La syntaxe :

```
UPDATE R SET A1=v1, A2=v2, ... An=vn
WHERE condition
```

- Exemple :

```
UPDATE Activite SET prix = prix * 1.1 WHERE nomStation = 'Passac'
```

DRIOUCH B.

20

Gestion des contraintes d'intégrité

- On fait appel à des *contraintes* pour empêcher l'entrée de données incorrectes dans des tables. Il s'agit d'une approche non spécifique à SQL Server mais mis en œuvre par tous les SGBD conformes au modèle relationnel et à la norme SQL.
- On utilise des contraintes pour :
 - Appliquer des règles au niveau d'une table chaque fois qu'une ligne est insérée, mise à jour ou supprimée dans cette table. La contrainte doit être satisfaite pour que l'opération réussisse.
 - Empêcher la suppression d'un enregistrement si il y a une dépendance avec un autre enregistrement de la table en relation.
 - Éviter la suppression d'une table dont dépend une autre table.
 - Appliquer des règles aux valeurs insérées dans une colonne pour vérifier qu'elles sont conformes aux règles définies

DRIOUCH B.

21

Types d'intégrité des données

- Intégrité de domaine
 - Spécifie un ensemble de valeurs pour une colonne
 - Détermine si les valeurs nulles sont autorisées
 - Implémentation par contrôle de validité
- Intégrité des entités
 - Toute ligne doit avoir un identifiant unique
 - Notion de clé primaire
- Intégrité référentielle
 - Assure l'intégrité des relations entre la clé primaire de la table référencée et les clés étrangères
 - On ne peut supprimer une ligne ni modifier la clé primaire si une clé étrangère y fait référence

DRIOUCH B.

22

Contraintes d'Intégrité

- Types de contraintes :

- DEFAULT
- NOT NULL
- UNIQUE
- PRIMARY KEY
- IDENTITY
- FOREIGN KEY
- CHECK

Toutes les contraintes sont stockées dans le dictionnaire de données.

DRIOUCH B.

23

Contraintes d'Intégrité

- DEFAULT
Spécifie la valeur fournie pour la colonne lorsque vous n'avez pas spécifié explicitement de valeur lors d'une insertion.
- La contrainte NOT NULL
Mots clés déterminant si les valeurs NULL sont permises ou non dans une colonne.
- La Contrainte PRIMARY KEY
Une contrainte PRIMARY KEY crée une clé primaire pour la table.
- IDENTITY (propriété)
Crée une colonne d'identité dans une table.
IDENTITY [(seed , increment)]
- La Contrainte de Clé UNIQUE
Une contrainte d'intégrité de type clé UNIQUE exige que chaque valeur dans une colonne ou dans un ensemble de colonnes soit unique
- La Contrainte FOREIGN KEY

DRIOUCH B.

24

Contraintes d'Intégrité

- Une contrainte peut être définie au niveau table ou colonne en fonction de sa nature.

```
CREATE TABLE [schema.]ntable
  (ncolumn datatype [DEFAULT expr]
   [column_constraint],
   ...
   [table_constraint]);
```

- *schema* : schéma [nom du propriétaire] de la table
- *table* : nom de la table
- *DEFAULT expr* : valeur par défaut à utiliser si une valeur est omise dans l'ordre INSERT
- *column* : nom de la colonne
- *datatype* : type de données et longueur de la colonne
- *column_constraint* : contrainte d'intégrité incluse dans la définition de la colonne
- *table_constraint* : contrainte d'intégrité incluse dans la définition de la table

DRIOUCH B.

25

Contraintes d'Intégrité

```
CREATE TABLE Stagiaire(
  id INT,
  nom VARCHAR(30),
  ...
  Groupe INT NOT NULL,
  CONSTRAINT st_id_pk
  PRIMARY KEY (id));
```

- Syntaxe de création d'une contrainte :
 - Contrainte au niveau colonne
column datatype [CONSTRAINT constraint_name] constraint_type,
 - Contrainte au niveau table
(column,... ,
[CONSTRAINT constraint_name] constraint_type
(column, ...), ...

constraint_name : nom de la contrainte
constraint_type : type de contrainte

DRIOUCH B.

26

Modifier, supprimer une contrainte

- Une contrainte fait partie des composants d'une table, donc Pour ajouter ou supprimé une contrainte on utilise la commande Alter Table.
- Syntaxe et Exemple:
 - Alter table *passee_ex* drop constraint *df__passee_ex__note__00551192*
 - Alter table *Passee_ex* Add CONSTRAINT *ntest DEFAULT (0) for note*

Pour récupérer le nom d'une contraint donnée par le système en utilise la procédure stockée *Exec sp_help nTable*

DRIOUCH B.

27

Contraintes d'Intégrité (Exemple)

- Exemple:
Soit le Schéma relationnel GestAud suivant:
Auditeur (*#IdAud*, Nom)
Examen (*#IdEx*, Salle, DateEx)
Passe_Ex (*IdAud*, *IdEx*, Note)

- Scripte :
Use master
Create database GestAud
On primary
 (name='GestAud_dat',
 filename='C:\GestAud_dat.mdf',
 size=3MB,
 filegrowth=10%)
Log on
 (name='GestAud_log',
 filename='C:\GestAud_log.ldf',
 size=1MB,
 filegrowth=10%)
Go
exec sp_helpdb GestAud
Go

DRIOUCH B.

28

Contraintes d'Intégrité (Exemple)

```
-- Creation des tables
USE GestAud
Create Table Auditeur(NuAud int identity, Nom varchar(30))
Create Table Examen(NuEx int identity, Salle varchar(20), DateEx smalldatetime)
Create Table Passe_Ex(NuAud int, NuEx int, note real default 0)
Go
-- Supprimer une table
Drop table Auditeur
Go
-- Alter ajouter, Modifier, renommer et supprimer
Alter Table Auditeur ADD TelAud char(10)
Alter Table Auditeur Alter Column TelAud varchar(10)
Exec sp_rename 'Auditeur.TelAud','Telephone','COLUMN'
Alter Table Auditeur Drop Column Telephone
Exec sp_help Auditeur

-- suppression de la contrainte avant d'ajouter une autre
--(nom donnée par le système et trouvé avec sp_help)
Alter table passe_ex drop constraint df__passe_ex__note__00551192
Alter table Passe_ex Add CONSTRAINT ntest DEFAULT (0) for note
```

DRIOUCH B.

29

Contraintes d'Intégrité (Exemple)

```
-- la contrainte not null
alter table Passe_ex Alter Column note real not null

-- la contrainte primary key
alter table Examen Alter Column NuEx int null
alter table Examen Add constraint ct_pk_nuex Primary key (NuEx)
alter table Auditeur Add constraint ct_pk_nuaud Primary key (NuAud)
alter table Examen Drop constraint ct_pk_nuex

-- Ajout sans clé primaire
insert into examen (NuEx , Salle , DateEx ) values (1,'S01','02/10/2009')
insert into examen (NuEx , Salle , DateEx ) values (2,'S02','12/11/2009')
-- Ajout avec clé primaire
insert into examen ( Salle , DateEx ) values ('S01','02/10/2009')
insert into examen ( Salle , DateEx ) values ('S02','12/11/2009')

insert into Auditeur ( Nom ) values ('Ali')
insert into Auditeur ( Nom ) values ('Ahmed')
select * from examen
select * from Auditeur
```

DRIOUCH B.

30

Contraintes d'Intégrité (Exemple)

```
-- la contrainte Foreign key
alter table Passe_ex Add constraint ct_fk_ex Foreign key (NuEx) REFERENCES Examen (NuEx)
alter table Passe_ex Add constraint ct_fk_aud Foreign key (NuAud) REFERENCES Auditeur (NuAud)
alter table Passe_ex Drop constraint ct_fk_passex
insert into Passe_ex (NuAud , NuEx , note) values (1,1,12.25)
insert into Passe_ex (NuAud , NuEx , note) values (1,3,12.25) -- ne peut pas etre ajouter

-- la contrainte UNIQUE
alter table Passe_Ex add constraint pex_unique unique (NuAud,NuEx)
insert into Passe_ex (NuAud , NuEx , note) values (1,1,13.5) -- ne peut pas etre ajouter

-- la contrainte check
alter table Passe_Ex add constraint pex_check check (note between 0 and 20)
insert into Passe_ex (NuAud , NuEx , note) values (1,2,25) -- ne peut pas etre ajouter
insert into Passe_ex (NuAud , NuEx , note) values (1,2,15)
```

DRIOUCH B.

31

Transaction

Une transaction est une suite d'opérations effectuées comme une seule unité logique de travail. Une unité logique de travail doit posséder quatre propriétés appelées propriétés ACID (Atomicité, Cohérence, Isolation et Durabilité).

- Atomique, au sens où on ne peut pas la diviser en une partie qui échoue et une partie qui réussit.
- Cohérence, au sens où une fois la transaction terminée, la base est de nouveau dans un état cohérent.
- Isolée, au sens où une transaction considère que, pendant son exécution, les données qu'elle manipule ne sont pas modifiées par une autre transaction.
- Durable, au sens où les modifications opérées par la transaction sont enregistrées de façon permanente (et recouvrables en cas de reconstruction de la base).

```
Begin TRANSACTION
Liste d'instruction SQL
si (Condition) ROLLBACK
...
COMMIT TRANSACTION
```

DRIOUCH B.

32

Les Indexes

- À l'instar de l'index d'un livre, l'index d'une base de données vous permet de retrouver rapidement des informations dans une table ou une vue indexée. Un index se compose de clés créées à partir d'une ou plusieurs colonnes dans la table ou la vue et de pointeurs qui mappent sur l'emplacement de stockage des données spécifiées. Un index bien conçu améliore de manière significative les performances des requêtes et des applications de base de données. Un index peut réduire la quantité de données qui doivent être lues par une requête pour retourner un ensemble de résultats.

if exists (select name from sys.indexes where name like N'Ind_Nom')

Drop index Ind_Nom on Auditeur

Go

Create index Ind_Nom

On Auditeur(Nom)

Go

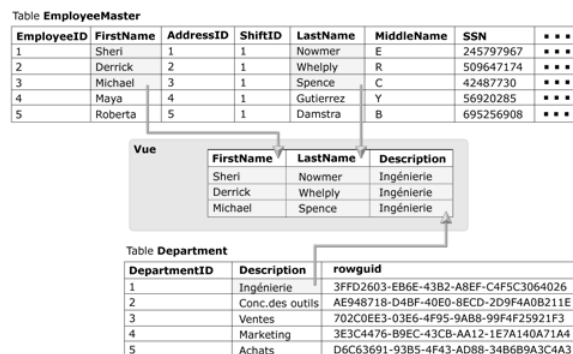
DRIOUCH B.

33

Sécurisé une BD (Vues)

■ Les Vues

Une vue est une table virtuelle dont la définition s'appuie sur des tables ou d'autres vues. Elle affiche des données qui ne lui sont pas propres, mais c'est comme une fenêtre par laquelle il est possible de visualiser ou de modifier des données venant des tables.



DRIOUCH B.

34

Sécurisé une BD (Vues)

■ Avantages d'une Vue

- Limite l'accès à la base de données en affichant une sélection de celle-ci.
- Permet aux utilisateurs de créer des requêtes simples pour extraire les résultats de requêtes complexes. Par exemple, vous pouvez extraire des informations de plusieurs tables sans savoir écrire un ordre de jointure.
- Garantit l'indépendance des données pour des utilisateurs et des programmes d'application contextuels. Une vue peut être utilisée pour extraire les données de plusieurs tables. Vous pouvez ainsi modifier la structure d'une table sans avoir à recompiler la vue.
- Permet à des groupes d'utilisateurs d'accéder aux données en fonction de leurs critères particuliers, notamment de règles de confidentialité. Ainsi, on peut proposer une vue restreinte d'une table à certains utilisateurs, en omettant une colonne par exemple (une vue sur la table des Salariés qui ne comporte pas la colonne salaire)

■ Une vue simple :

- Est dérivée des données d'une seule table
- Ne contient ni fonctions ni groupes de données
- Permet d'exécuter des opérations de mise à jour

■ Une vue complexe :

- Est dérivée des données de plusieurs tables
- Peut contenir des fonctions ou des groupes de données
- Ne permet pas toujours d'exécuter des opérations de mise à jour

DRIOUCH B.

35

Sécurisé une BD (Vues)

■ Création d'une Vue

```
CREATE VIEW [ schema_name. ] view_name  
AS select_statement  
[ WITH CHECK OPTION ] [ ; ]
```

■ WITH CHECK OPTION.

Cette option impose que toutes les instructions de modification de données exécutées sur la vue respectent le critère défini dans *select_statement*.

■ Modifier une vue : Alter

```
ALTER VIEW [ schema_name . ] view_name  
AS select_statement  
[ WITH CHECK OPTION ] [ ; ]
```

■ Suppression d'une Vue

L'ordre DROP VIEW *view_name* : permet de supprimer une vue.

La suppression d'une vue n'a aucun effet sur les tables sur lesquelles la vue est basée.

DRIOUCH B.

36

Exemple (Vues)

- Créez une vue qui permet d'extraire les coordonnées des pilotes dont l'anniversaire aura lieu dans les 5 jours. Calculez l'âge du pilote afin de l'inclure dans le texte du message.

Create view AnnPilote

AS

Select Nom, datenaissance, datediff(year,datenaissance,getdate())
as age

From pilote

Where cast(datepart(day,datenaissance) as
varchar)+'/'+cast(datepart(month,datenaissance) as
varchar)+'/'+cast(datepart(year,getdate()) as varchar)
between getdate() and dateadd(day,5,getdate())

DRIOUCH B.

37

Sécurisé une BD (Connexion)

- deux modes d'authentification
 - Authentification approuvée par Windows
 - Principe associé à :
 - Un compte utilisateur
 - Un groupe d'utilisateurs
 - Mise en œuvre rapide
 - Fonctionnalités de sécurité avancées
 - Authentification SQL Server
 - Pour permettre l'accès à des clients non Microsoft
 - Couche supplémentaire de sécurité
 - Plus lourd à mettre en œuvre
 - Un serveur peut mettre en œuvre le mode d'authentification Windows seul ou le mode mixte (Windows + SQL Server)

DRIOUCH B.

38

Sécurisé une BD (Connexion)

■ Création des comptes de connexions

- sp_addlogin permet d'ajouter un compte de connexion SQL Server :
 - EXEC sp_addlogin 'Bost', 'xxx123', 'Master'
Utilisateur, mot de passe, base données par défaut
- sp_grantlogin permet d'autoriser la connexion d'un utilisateur ou groupe d'utilisateurs
 - EXEC sp_grantlogin 'D-Afpabrive\Bost'
La procédure sp_denylogin permet elle d'interdire l'accès d'un utilisateur ou d'un groupe.

DRIOUCH B.

39

Sécurisé une BD (Connexion)

Use Master

```
CREATE LOGIN [OFPPT\ali] FROM WINDOWS;  
GO
```

Use Master

```
CREATE LOGIN Ahmed WITH PASSWORD = '321',  
    DEFAULT_DATABASE =VolAvion;  
GO
```

Use VolAvion

```
CREATE USER Ahmed FOR LOGIN Ahmed;  
GO
```

DRIOUCH B.

40

Sécurisé une BD (Rôle)

- Les rôles permettent de regrouper en une seule unité logique des utilisateurs ou groupes ayant des droits similaires
- Il existe des rôle fixes standard pour donner accès à des fonctions administratives courantes :
 - Au niveau des serveurs : rôle fixe de serveur
 - Au niveau db : rôle fixe de base de données
- Vous pouvez définir vos propres rôles afin de regrouper des droits sur des fonctions accessibles à une catégorie d'employés.
 - Rôles de base de données définis par l'utilisateur

DRIOUCH B.

41

Sécurisé une BD (Rôle)

■ Principaux rôles fixes de serveur

<i>Rôle</i>	<i>Autorisation</i>
sysadmin	Exécution de toutes les activités
dbcreator	Création et modification des bases de données
processadmin	Gestion des processus SQL Server
serveradmin	Configuration des paramètres de serveur
setupadmin	Gestion de la réplication
securityadmin	Gestion des connexions et autorisations

DRIOUCH B.

42

Sécurisé une BD (Rôle)

■ Rôles fixes de base de données

<i>Rôle</i>	<i>Autorisation</i>
public	Conservation de toutes les autorisations par défaut
db_owner	Exécution de toutes les activités sur une base
db_accessadmin	Ajout ou suppression d'utilisateurs, groupes et rôles
db_ddladmin	Ajout, modif ou supp. d'objets de base de données
db_security admin	Affectation d'autorisations d'instruction et sur les objets
db_backupoperator	Sauvegarde et restauration de la base de données
db_datareader	Lecture des données dans toutes les tables
db_datawriter	Ajout, modification ou suppression des données
db_denydatareader	Impossible de lire les données
db_denydatawriter	Impossible de modifier les données

DRIOUCH B.

43

Sécurisé une BD (Rôle)

■ Création et Affectation de rôle utilisateur

```
USE VolAvion
CREATE ROLE auditors
GO
```

```
USE VolAvion
CREATE ROLE auditors [AUTHORIZATION db_securityadmin]
GO
```

```
Sp_addrolemember [@rolename=] 'Role', [@membername=] 'security_account'
Exp : Exec sp_addrolemember 'auditors', 'Bost'
```

```
sp_droprolemember [@rolename=] 'Role', [@membername=] 'security_account'
Exp : Exec sp_droprolemember 'auditors', 'Bost'
```

DRIOUCH B.

44

Sécurisé une BD (Autorisation)

■ Attribution d'autorisations : GRANT

```
GRANT SELECT  
ON Note  
TO R-Formateur (Rôle défini par utilisateur)
```

```
GRANT INSERT, UPDATE, SELECT  
ON Note  
TO Ali, G-Stg (Utilisateurs ou groupes)
```

Utilisateur/rôle	Insert	Update	Select
Ali	☒	☒	☒
G-Stg	☒	☒	☒
R-Formateur	☐	☐	☒

DRIOUCH B.

45

Sécurisé une BD (Autorisation)

■ Refus d'autorisations : DENY

```
DENY SELECT,INSERT,UPDATE  
ON products  
TO Bost  
Même si Bost appartient au rôle spécifique R-Consult,  
il ne pourra pas accéder à la table products en  
consultation .
```

Utilisateur/rôle	Insert	Update	Select
Bost	☒	☒	☒
GG-IDG10	☒	☒	☒
R-Consult	☐	☐	☒

DRIOUCH B.

46

Sécurisé une BD (Autorisation)

■ Retrait d'autorisations : REVOKE

```
REVOKE SELECT,UPDATE,INSERT  
ON products  
TO Bost
```

Bost peut obtenir un droit de lecture implicite par un rôle le cas échéant mais il n'a plus de droit explicite sur l'objet Products

Utilisateur/rôle	Insert	Update	Select
Bost	☐	☐	☐
GG-IDG10	☒	☒	☒
R-Consult	☐	☐	☒

DRIOUCH B.

47

Exercice Sécurité

Soit le schéma relationnel GestStg Suivante :

- Stagiaire(idStg, Nom, Prenom)
 - Matiere (idMat, Libelle, Coeff, Note)
 - Note (idStg, idMat, Note)
- 1) Créer la base de donnée avec un jeu de test
 - 2) Configurer le serveur sur authentification mixte
 - 3) Créer trois utilisateurs SQL Server login/Mot de pass (Ali/123, Ahmed/321, Imane/abc)
 - 4) Créer trois Rôles (Surveillant, Formateur, Directeur)
 - 5) Donner le droit de lecture pour les trois utilisateur avec le rôle base de donnée db_datareader
 - 6) Donner le droit de modification (Insert, Update, Delete)
 - sur Stagiaire par Surveillant
 - sur Matiere par Directeur
 - sur Note par Formateur
 - 7) ajouter les utilisateurs
 - Ali dans Surveillant
 - Ahmed dans Formateur
 - Imane dans Directeur
 - 8) faire un test avec chaque utilisateur

DRIOUCH B.

48