

Secteur Tertiaire Informatique  
Filière « Etude et développement »

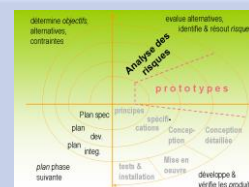
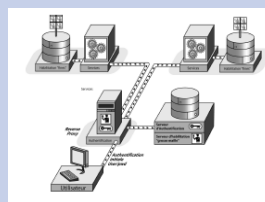
Séquence « Développer des composants d'accès  
aux données »

Cas PAPYRUS SQL Server :  
Programmer dans le langage du SGBD

Apprentissage

Mise en pratique

Evaluation





## TABLE DES MATIERES

Table des matières .....	3
1. L'EXISTANT .....	5
2. LES FONCTIONS .....	6
2.1 Fonction fn_DateFormat .....	6
2.2 Création d'une fonction scalaire fn_Satisfaction .....	6
2.3 Création d'une fonction table en ligne .....	7
3. LES PROCEDURES STOCKEES .....	8
3.1 Création d'une procédure stockée sans paramètre .....	8
3.2 Création d'une procédure stockée avec un paramètre en entrée .....	8
3.3 Création d'une procédure stockée avec des paramètres en entrée et en sortie .....	8

## **Préambule**

Ce document expose les besoins de programmation du SGBD relatifs au Cas Papyrus pour lequel il a été créé une base de données.

## **Objectifs**

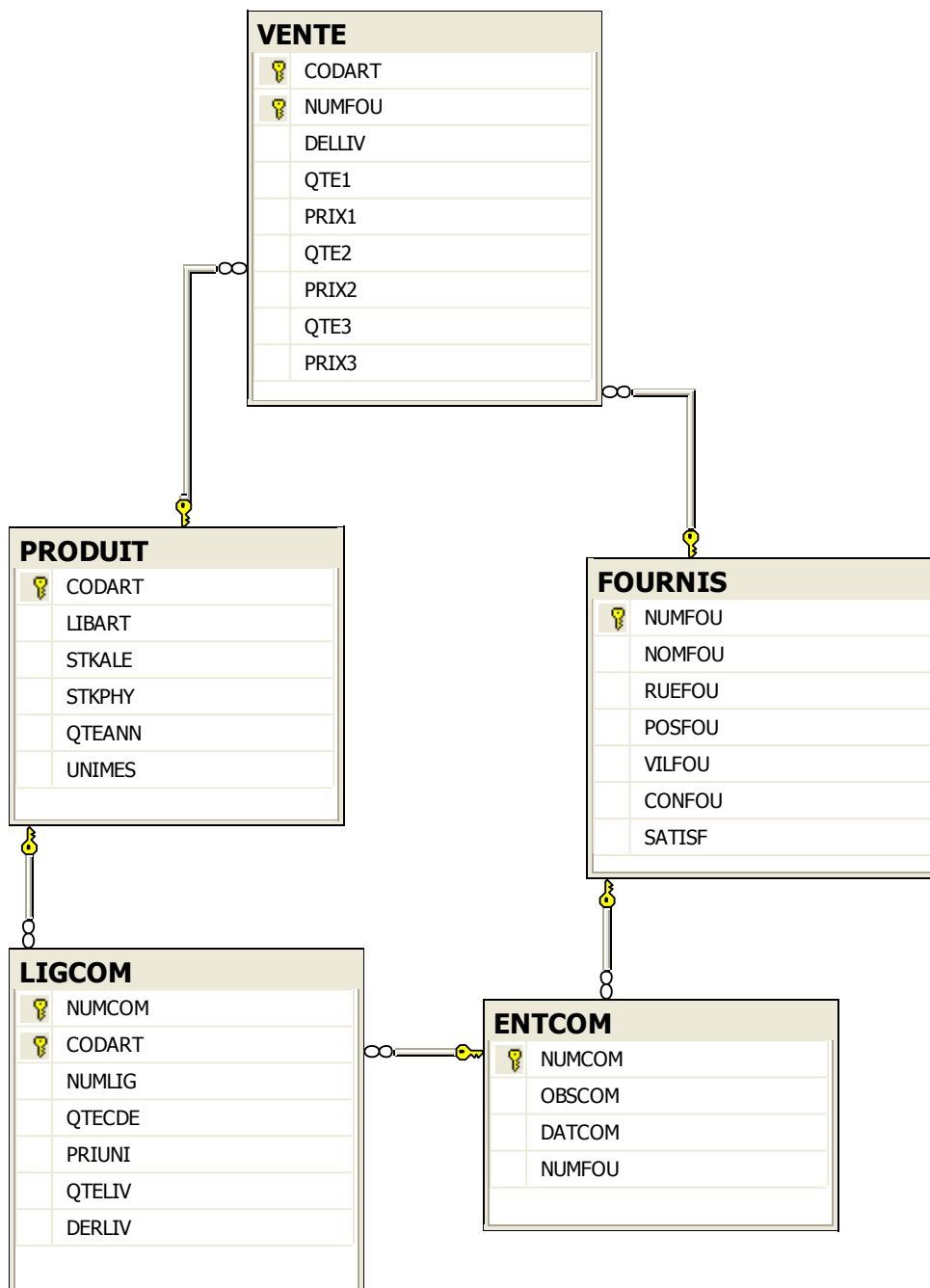
Mettre en pratique la programmation du SGBD SQL Server tant au niveau des fonctions que des procédures stockées.

## **Méthodologie**

Réalisez les exercices progressivement, après étude des ressources de formation concernant chacun des chapitres, programmation des fonctions et programmation des procédures stockées.

# 1. L'EXISTANT

La base de données relationnelle PAPYRUS est constituée des relations suivantes :



**PRODUIT** (CODART, LIBART, STKALE, STKPHY, QTEANN, UNIMES)

**ENTCOM** (NUMCOM, OBSCOM, DATCOM, NUMFOU)

**LIGCOM** (NUMCOM, CODART, NUMLIG, QTECDE, PRIUNI, QTELIV, DERLIV)

**FOURNIS** (NUMFOU, NOMFOU, RUEFOU, POSFOU, VILFOU, CONFOU, SATISF)

**VENTE** (CODART, NUMFOU, DELLIV, QTE1, PRIX1, QTE2, PRIX2, QTE3, PRIX3)

Le jeu d'essai a été constitué lors d'une séance précédente.

Cas Papyrus SQL Server - Programmer dans le langage du SGBD

Afpa © 2015 – Section Tertiaire Informatique – Filière « Etude et développement »

## 2. LES FONCTIONS

### 2.1 FONCTION `fn_DateFormat`

Pour bien comprendre l'utilité d'une fonction, créez, dans votre base de données SQL Server Papyrus, la fonction `fn_DateFormat` expliquée dans le support d'apprentissage, et utilisez la dans la requête 5 qui affichait les commandes d'un mois donné, de manière à afficher la date des commandes avec ce format de date (très français avec le séparateur / entre jour et mois et entre mois et année).

**Besoin :** « Quelles sont les commandes passées au mois de mars et avril ? »  
(Affichage numéro de commande, date de commande)

**Requête SQL réalisée à adapter pour utiliser la fonction `fn_DateFormat` :**

```
SELECT numcom, datcom
FROM entcom
WHERE MONTH (datcom) IN (3,4)
```

### 2.2 CREATION D'UNE FONCTION SCALAIRE `fn_Satisfaction`

Créer la fonction scalaire `fn_Satisfaction`, qui avec l'indice de satisfaction en entrée, affiche un niveau de satisfaction en clair :

- Indice = Null, 'sans commentaire'
- Indice = 1 ou 2, 'Mauvais'
- Indice = 3 ou 4, 'Passable'
- Indice = 5 ou 6, 'Moyen'
- Indice = 7 ou 8, 'Bon'
- Indice = 9 ou 10, 'Excellent'

Adaptez au besoin votre jeu d'essai.

Testez votre fonction, en affichant le niveau de satisfaction des fournisseurs lors d'une nouvelle requête renvoyant le numéro fournisseur, le nom fournisseur et la satisfaction en clair.

## 2.3 CREATION D'UNE FONCTION TABLE EN LIGNE

Créer la fonction `fn_CA_Fournisseur`, qui en fonction d'un code fournisseur et d'une année entrés en paramètre, restituera le CA de ce fournisseur pour l'année souhaitée. (adaptation de la requête 10).

**Besoin :** « Calculer le chiffre d'affaires par fournisseur pour l'année en cours sachant que les prix indiqués sont HT et que le taux de TVA est 20,00% ».

**Requête SQL réalisée à adapter pour utiliser la fonction `fn_CA_Fournisseur` :**

```
SELECT fournis.numfou, nomfou, sum(qtecde * priuni * 1,20) as CA
FROM fournis
    INNER JOIN entcom
ON fournis.numfou = entcom.numfou
    INNER JOIN ligcom
ON entcom.numcom = ligcom.numcom
WHERE YEAR(datcom)=YEAR(GETDATE())
GROUP BY fournis.numfou, nomfou
```

### Test de l'option SCHEMABINDING

Créez une table `FOURNIS_IND` de structure identique à `FOURNIS` et insérez les lignes de la table `FOURNIS` dont la colonne indice de satisfaction est 'Bon'

Créez la fonction `fn_Compte`, avec l'option `SCHEMABINDING` qui délivre le nombre de lignes de la table `FOURNIS_IND`.

Supprimez la table `FOURNIS_IND`, et expliquez le processus.

### 3. LES PROCEDURES STOCKEES

#### 3.1 CREATION D'UNE PROCEDURE STOCKEE SANS PARAMETRE

Créez la procédure stockée **prc\_Lst\_fournis** correspondant à la requête 2

« Afficher le code des fournisseurs pour lesquels une commande a été passée ».

Exécutez-la pour vérifier qu'elle fonctionne conformément à votre attente.

Exécutez les procédures stockées système **sp\_help** et **sp\_helptext** pour obtenir des informations sur cette procédure stockée.

#### 3.2 CREATION D'UNE PROCEDURE STOCKEE AVEC UN PARAMETRE EN ENTREE

Créer la procédure stockée **prc\_Lst\_Commandes**, qui liste les commandes ayant un libellé particulier dans le champ **OBSCOM**

(Cette requête sera construite à partir de la requête 9).

#### 3.3 CREATION D'UNE PROCEDURE STOCKEE AVEC DES PARAMETRES EN ENTREE ET EN SORTIE

Créer la procédure stockée **prc\_CA\_Fournisseur**, qui pour un code fournisseur et une année entrés en paramètre, calcule et restitue le CA potentiel de ce fournisseur pour l'année souhaitée

(adaptation de la requête 10).

Le chiffre d'affaires est renvoyé dans le paramètre de sortie. On n'exécutera la requête que si le code fournisseur est valide, c'est-à-dire existe dans la table **FOURNIS**, sinon on renverra un code d'état égal à -100.

Testez la procédure stockée avec différentes valeurs des paramètres.

Créez le message d'erreur : « Fournisseur inexistant »

Créez une nouvelle procédure stockée **prc\_CA\_Fournisseur2** qui lorsque le code **FOURNIS** n'existe pas, renvoie le message d'erreur s'inscrivant dans le journal d'erreurs du serveur et dans le journal des événements.

Vérifiez la présence du message dans les 2 journaux.

Vous pouvez prolonger cet exercice, en créant un message paramétré, incluant le paramètres d'appel code fournisseur :

« Fournisseur x inexistant »



## **CREDITS**

### **ŒUVRE COLLECTIVE DE l'AFPA**

**Sous le pilotage de la DIIP et du centre d'ingénierie sectoriel Tertiaire-Services**

#### **Equipe de conception (IF, formateur, mediatiseur)**

B. Hézard - Formateur

Ch. Perrachon – Ingénieure de formation

**Date de mise à jour : 12/10/15**

### **Reproduction interdite**

Article L 122-4 du code de la propriété intellectuelle.

« Toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droits ou ayants cause est illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction par un art ou un procédé quelconque. »