

Requêtes simples à fabriquer :

- 1) Prix maxima, minima, prix moyens, nombre de vendeur par espèce
- 2) Vendeurs de pins par code postal
- 3) Totaux de stocks par vendeurs
- 4) Vue d'ensemble de toute la DB sous forme de VUE
- 5) Requête pour augmenter de 10% le prix de vente de tous les pins
- 6) Requête pour afficher un tableau avec les prix (HTVA) et les prix (TVAC) pour les chênes de tous les vendeurs
- 7) requête permettant d'extraire les informations nécessaires pour créer des étiquettes d'adresses pour tous les vendeurs de 7531 Tournai.

Compléments :

- 1) Dans la table T_Fournisseur créez un champ calculé donnant le pays en toutes lettres
--> Créer une fonction SQL qui renvoie cette chaîne sur base de l'abréviation du pays
--> Mettre la fonction dans la formule du champ calculé

Exemples de fonctions - à remplacer par la vôtre

```
CREATE FUNCTION [EXPLtc3] (@aller char(1)) RETURNS varchar(10) AS
BEGIN
DECLARE @st varchar(10)
select @ST=rtrim(intiancinouv) from tc3 where codeancinouv=@aller
RETURN isnull(@ST,@ALLER)
END
```

```
CREATE FUNCTION [PaysEnFrancais] (@aller char(2))
RETURNS varchar(255) AS
BEGIN
    DECLARE @st varchar(255)
    select @ST= intisaturn from tpays where codesaturn=@aller
    RETURN isnull(@ST,'??PAYS INCONNU??')
end
```

- 2.) Champ calculé donnant la liste des fournisseurs capables de fournir un arbre donné séparées par des ;

- > Créer une fonction SQL qui renvoie cette chaîne sur base du n° de l'arbre
- > Mettre la fonction dans la formule du champ calculé

Attention, cette fonction doit utiliser des curseurs (plus d'une réponse)

Exemple de fonction - à remplacer par la vôtre - le curseur sert à parcourir la table et à construire le vecteur des champs

```
CREATE FUNCTION EnsECTS (@impl char(1),@annaca smallint,@codesouscours
varchar(20),@anncreatsouscours smallint)
RETURNS varchar(500) AS
BEGIN
DECLARE @Retour varchar(500),@np varchar(100)
DECLARE Curseur CURSOR FOR
select eprofs.nom+' ' +isnull(eprofs.prenom,'') from eprofs inner join
annprofssouscours AP on ap.impl=eprofs.impl and AP.abrevprof=
eprofs.abrevprof where ap.impl=@impl and ap.annaca=@annaca and
ap.codesouscours=@codesouscours and ap.anncreatsouscours=
@anncreatsouscours order by eprofs.nom
OPEN Curseur
FETCH NEXT FROM Curseur INTO @np
set @retour=''
WHILE (@@FETCH_STATUS=0)
BEGIN
    set @retour=@retour+@np+', '
    FETCH NEXT FROM Curseur INTO @np
END
```

```

END
CLOSE CURSEUR
DEALLOCATE CURSEUR
if len(@retour)>0 set @retour= (left(@retour,len(@retour)-1))
return(@retour)
END

```

3) Créez une PROCEDURE STOCKEE qui, lorsqu'elle reçoit un n° d'arbre et un n° de fournisseur crée automatiquement un enregistrement dans la table sources et renvoie le n° de la ligne créée.

```

Exemple de début de procédure stockée
CREATE PROCEDURE Ptraitercours @abrcours varchar(20),@anncreatcours
smallint, @moyenne smallmoney OUTPUT,@taquetinf smallmoney
OUTPUT,@taquetsup smallmoney OUTPUT
as
DECLARE @COMPTEUR int,@SOMMETOT smallmoney,@TOTAL smallmoney,@cotesess
smallmoney,@statsess char(1),@nbpoints smallint,
        @echec bit, @echecgrave bit, @impasse bit, @manquecote bit
SET @COMPTEUR=0
SET @SOMMETOT=0
SET @TOTAL=0
SET @echec=0
SET @echecgrave=0
SET @impasse=0
SET @manquecote=0

```

etc

4) Créez dans la table T_Sources un DECLENCHEUR qui, à chaque EFFACEMENT d'un enregistrement de la table enverra dans une table à créer le nom du fournisseur associé et le nom de l'arbre, ainsi que les infos de stock et de prix. Il faut aussi si possible noter la date et l'identificateur de la personne qui a effectué l'effacement