

# $TP_0$ Miage FA

M1 MIAGE - 2017/2018

January 9, 2018

On considère un document XML issu de l'open data public

[www.prix-carburants.gouv.fr/rubrique/opendata/](http://www.prix-carburants.gouv.fr/rubrique/opendata/): *pxcarburants.xml*.

Ce document recense, par jour, par année, sur tous les points de distribution de carburants de France, le relevé des différents prix des carburants à la pompe. A un point de distribution est associé un ensemble d'information les caractérisant (adresse, coordonnées GPS, services complémentaires apportés à la clientèle).

**Question 1** Afin de réduire le volume d'information à traiter, vous allez re-traiter ce fichier (avec l'outil de votre choix) que sorte que:

- on ne conserve que les points de distribution des départements 59 et 62
- on oublie les éléments balisés **rupture**
- on oublie l'attribut **maj** sur les éléments balisés **prix**
- on ne conserve que les points de distribution pour lesquels on dispose de relevé de prix
- on recalcule les valeurs des longitudes et latitudes par division par 100000

Vous obtenez ainsi un fichier XML **carburants5962.xml**

On voudrait injecter les données présentes de ce fichier dans une base de données relationnelle. Considérant qu'un point de vente distribue un nombre très limité de carburants et dispense aussi un nombre limité de services, on choisit un schéma (non standard) exploitant les possibilités du relationnel étendu et notamment:

- définir un type structuré d'adresse
- réutiliser le type **SDO.GEOMETRY** pour représenter la localisation du point de vente
- utiliser les types collections (**NESTED\_TABLE** ou **VARRAY**) pour stocker, sous la forme d'une collection : les prix des différents carburants, les services présents sur le point de vente.

Ainsi, on pourrait mettre en œuvre les dispositions suivantes :

```
create or replace
type Tcar as object(
    nom varchar2(10),
    prix number(6)) ;

create or replace
type Lcar is varray(20) of tcar;

create or replace
type Tservice is varray(15) of varchar2(50);

create or replace
TYPE ADRESSE_TYPE as object
(rue varchar2(100),
 code char(5),
 ville varchar2(50)) ;

create table PDV(
id varchar2(10),
adr adresse_type,
loggps sdo_geometry,
pxcarbu Lcar,
services Tservice )
```

**Question 2** Par quels moyens pourrait-on injecter les données du fichiers XML dans la table PDV.

Exemples d'initialisation de VARRAY

```
soit
type somevarchar is varray(4) of varchar2(15);
alors
team somevarchar := somevarchar('John','Paul','Ringo', 'George') ;

soit
type someadresse is varray(4) of adresse_type ;
alors
repertoire someadresse = sommeadresse(adresse_type(),adresse_type(), ...) ;
```

Ainsi, le fragment suivant du fichier XML devrait entraîner l'insertion suivante dans la table PDV

```
<pdv id="59111001" latitude="50.2924914941" longitude="3.30109415523" cp="59111" pop="R">
  <adresse>AVENUE KENNEDY</adresse>
  <ville>BOUCHAIN</ville>
  <ouverture debut="01:00:00" fin="01:00:00" saufjour=""/>
  <services>
```

```

        <service>Vente de gaz domestique</service>
        <service>Automate CB</service>
    </services>
    <prix nom="Gazole" id="1" valeur="1376"/>
    <prix nom="SP95" id="2" valeur="1462"/>
    <prix nom="SP98" id="6" valeur="1527"/>
</pdv>

insert into pdv value(
'59111001',
adresse_type('AVENUE KENNEDY','59111','BOUCHAIN'),
sdo_geometry(2001,8307,sdo_point_type(3.30109415523,50.2924914941, null), null, null),
lcar(tcar('Gazole',1376),tcar('SP95',1462),tcar('SP98,1527')),
Tservice('Vente de gaz domestique','Automate CB'));

```

Remplissez la table PDV (si vous le jugez utile, vous pouvez le faire en plusieurs étapes)