# Introduction à la Déclaration Sociale Nominative

## Définition d’un fichier DSN

La Déclaration Sociale Nominative (DSN) est un fichier mensuel produit à partir de la paie destinée à communiquer les informations nécessaires à la gestion de la protection sociale des salariés aux organismes et administrations concernées.

La donnée est stockée dans ce fichier est en texte clair. On l’appelle aussi une bande DSN.

Les informations contenues dans une bande sont présentées séquentiellement ligne par ligne. Chaque ligne contient deux colonnes clé et valeur qui sont séparées par une virgule. La valeur est entourée par une paire d’apostrophes comme l’extrait suivant :

S21.G00.30.003,'TRAN'

S21.G00.30.004,'Xuan Tien'

S21.G00.30.007,'Antony'

S21.G00.30.009,'92160'

S21.G00.40.001,'01042017'

S21.G00.40.006,'Consultant technique junior'

S21.G00.40.001,'01042018'

S21.G00.40.006,'Consultant technique'

Les données au-dessus veulent dire qu’une personne qui s’appelle TRAN Xuan Tien travaillait en tant que « Consultant technique junior » en 2017 et travaille en tant que « Consultant technique » en 2018.

## La structure d’arbre du fichier DSN

Les rubriques se regroupent en bloc par les dix premiers caractères. Dans l’extrait, il y a un bloc « S21.G00.30 » et deux blocs « S21.G00.40 » dont le premier montre les données d’individu de cette personne et le deuxième et troisième montrent ses données d’affection dans un établissement de son entreprise.

Les blocs s’organisent dans une structure hiérarchique. Par exemple le bloc « S21.G00.30 » est un père qui peut avoir deux blocs fils « S21.G00.40 ». Cela s’interprète : une personne peut avoir plusieurs affections.

Voici une partie de la définition de cette structure avec la cardinalité:

S21.G00.30 - Individu (0,\*)

S21.G00.31 - Changements Individu (0,\*)

S21.G00.34 - Pénibilité (0,\*)

S21.G00.40 - Contrat (contrat de travail, convention, mandat) (1,\*)

S21.G00.41 - Changements Contrat (0,\*)

S21.G00.60 - Arrêt de travail (0,\*)

S21.G00.66 - Temps partiel Thérapeutique (0,\*)

D’où :

« 0 » signifie que le bloc peut être absent par rapport à son bloc père

« 1 » signifie que le bloc doit être présent au moins une fois par rapport à son bloc père

« \* » signifie que le bloc ne peut être présent plusieurs fois par rapport à son bloc père

## Outil de contrôle du fichier DSN

Un outil de contrôle du fichier DSN doit avoir trois fonctions :

* Chargement du fichier en mémoire,
* Visualisation des données chargées,
* Modification des données,
* Sauvegarde des données sur un fichier corrigé

Les données du fichier se base sur une structure d’arbre définie par un fichier en texte clair.

## L'objectif de l’exercice

En raison des contraintes de temps, on se limite à construire cet arbre.

L'objectif de cet exercice est donc d'implémenter un modèle de l’arbre spécifié sous la forme de tests unitaires. Autrement dit : l'objectif de l'exercice est de faire passer en vert un jeu de tests.

## Description de l’exercice

Vous disposez pour cela d'une solution comprenant 3 projets :

Arbre.Tests, contient les tests unitaires

Arbre.Model, contient 2 interfaces.

Arbre.Impl contient une classe statique, le CreerArbre qui est le seul point d’entrée pour manipuler le modèle. C’est cet assembly qui devra contenir toute l’implémentation. Pour faire tourner les tests unitaires il vous suffit de configurer le projet Arbre.Tests en tant que projet de démarrage puis d'exécuter la solution.

Votre implémentation ne doit pas modifier Arbre.Model.

Pour que les tests passent en vert, vous avez le droit de faire ce que bon vous semble dans le projet Arbre.Impl

Voici une courte description du modèle :

* Un fichier (IDsn) contient :
* Un dictionnaire des nœuds (INoeud) dont chaque nœud s’identifie par son id unique.
* Un arbre dont la racine (INoeud) est le point d’accès.
* Un nœud (INoeud) contient les propriétés :
* Id (identifiant unique)
* Niveau (string) : profondeur par rapport de la racine
* InsMin (string) : nombre minimum de instances du nœud que son parent peut avoir
* InsMax (string) : nombre maximum de instances du nœud que son parent peut avoir
* Libelle (string) : libellé du nœud
* Parent (INoeud) : le nœud parent – le nœud racine n’a pas de parent
* Une liste de ses enfants (INoeud)
* L’arbre se construit par la lecture de trois fichiers :
* Envoi.txt : l’entête de tous types d’arbre
* Envoi Totaux.txt : le pied de tous types d’arbre
* DADSU COMPLETE.txt : le corps d’un arbre complet
* L’arbre se sauvegarde sur un fichier ArbreResultat.txt. Ce fichier est utilisé pour comparer contre le fichier Arbre.txt. Si les deux fichiers sont identiques, l’arbre se construit de manière correcte.