1. **Préparation de la fonction FUNCTION getEffectifData**

Créer des types de données personnalisés (**t\_effectif\_table** et **t\_effectif\_row**) qui seront utilisés dans la fonction **getEffectifData**

1. **Création du type t\_effectif\_row** :

Script 1 :

**create or replace TYPE t\_effectif\_row AS OBJECT(**

**EFF\_ID Number,**

**DR\_ID Number,**

**DEP\_ID Number,**

**TYPE\_BUR\_ID Number,**

**NOM\_PRENOM VARCHAR2(256),**

**USER\_MAIL VARCHAR2(256),**

**TYPE\_JOB VARCHAR2(256),**

**NUMERO\_BUREAU VARCHAR2(256),**

**MOBILE VARCHAR2(256)**

**);**

Ce script crée un type d'objet **t\_effectif\_row** qui représente une ligne de données dans la table des effectifs. Chaque objet de ce type contient les attributs suivants :

* **EFF\_ID** : Identifiant de l'effectif (type Number).
* **DR\_ID** : Identifiant de la direction (type Number).
* **DEP\_ID** : Identifiant du département (type Number).
* **TYPE\_BUR\_ID** : Identifiant du type de bureau (type Number).
* **NOM\_PRENOM** : Nom et prénom de l'employé (type VARCHAR2).
* **USER\_MAIL** : Adresse e-mail de l'utilisateur (type VARCHAR2).
* **TYPE\_JOB** : Type d'emploi (type VARCHAR2).
* **NUMERO\_BUREAU** : Numéro de bureau (type VARCHAR2).
* **MOBILE** : Numéro de téléphone mobile (type VARCHAR2).

1. **Création du type t\_effectif\_table** :

Script 2 :

**create or replace TYPE t\_effectif\_table AS TABLE OF t\_effectif\_row;**

Ce script crée un type de table **t\_effectif\_table** qui est une table de type **t\_effectif\_row**

**Cela permet de définir un type de variable qui peut contenir plusieurs lignes de données ; Chaque ligne étant représentée par le type t\_effectif\_row**

**Cette table sera utilisée pour stocker les résultats de la fonction getEffectifData**

**En résumé, ces deux scripts définissent des types de données personnalisés qui seront utilisés dans la fonction getEffectifData pour manipuler et renvoyer des ensembles de données structurées.**

1. **Création de la fonction FUNCTION getEffectifData**

**Code :**

**CREATE OR REPLACE FUNCTION getEffectifData (p\_current\_user in varchar2) RETURN t\_effectif\_table**

**AS**

**v\_result t\_effectif\_table;**

**v\_dr number;**

**v\_current\_user varchar2(256):=trim(p\_current\_user);**

**BEGIN**

**begin**

**select dr\_id into v\_dr from gpi\_users where users\_gpi=v\_current\_user;**

**dbms\_output.put\_line('dir id=,' ||v\_dr)**

**exception when no\_data\_found then**

**Rollback;**

**end;**

**v\_result := t\_effectif\_table();**

**if(v\_dr=1) then**

**for rec in (SELECT EFF\_ID,**

**DR\_ID,**

**DEP\_ID,**

**TYPE\_BUR\_ID,**

**NOM\_PRENOM,**

**USER\_MAIL,**

**TYPE\_JOB,**

**NUMERO\_BUREAU,**

**MOBILE**

**FROM EFFECTIF) loop**

**v\_result.extend; v\_result(v\_result.count):=t\_effectif\_row(rec.EFF\_ID,rec.DR\_ID,rec.DEP\_ID,rec.TYPE\_BUR\_ID,rec.NOM\_PRENOM,rec.USER\_MAIL,rec.TYPE\_JOB,rec.NUMERO\_BUREAU,rec.MOBILE);**

**end loop ;**

**RETURN v\_result;**

**end if;**

**if(v\_dr !=1) then**

**for rec in (SELECT EFF\_ID,**

**DR\_ID,**

**DEP\_ID,**

**TYPE\_BUR\_ID,**

**NOM\_PRENOM,**

**USER\_MAIL,**

**TYPE\_JOB,**

**NUMERO\_BUREAU,**

**MOBILE**

**FROM EFFECTIF where dr\_id=(select dr\_id from gpi\_users where users\_gpi=v\_current\_user) ) loop**

**v\_result.extend; v\_result(v\_result.count):=t\_effectif\_row(rec.EFF\_ID,rec.DR\_ID,rec.DEP\_ID,rec.TYPE\_BUR\_ID,rec.NOM\_PRENOM,rec.USER\_MAIL,rec.TYPE\_JOB,rec.NUMERO\_BUREAU,rec.MOBILE);**

**end loop ;**

**RETURN v\_result;**

**end if;**

**END getEffectifData;**

**Explication :**

**En résumé, cette fonction récupère l'identifiant de direction d'un utilisateur, puis sélectionne les données d'effectif en fonction de cet identifiant, avant de les renvoyer sous forme d'un ensemble de données structurées de type t\_effectif\_table.**

1. **Signature de la fonction:**
   * **La fonction s'appelle getEffectifData**
   * **Elle prend en paramètre un nom d'utilisateur actuel (p\_current\_user) de type varchar2.**
   * **Elle renvoie un ensemble de données de type t\_effectif\_table**
2. **Déclaration des variables locales:**
   * **v\_result : Une variable de type t\_effectif\_table pour stocker les résultats de la fonction.**
   * **v\_dr : Une variable de type number pour stocker l'identifiant de direction (dr\_id).**
   * **v\_current\_user : Une variable de type varchar2(256) pour stocker le nom d'utilisateur actuel, après suppression des espaces blancs.**
3. **Bloc exécutable:**
   * **Le bloc BEGIN marque le début de la section exécutable de la fonction.**
4. **Bloc de traitement de l'exception:**
   * **Le bloc begin imbriqué est utilisé pour traiter l'exception no\_data\_found qui peut se produire lors de la sélection de l'identifiant de direction.**
   * **Si aucune donnée n'est trouvée, la fonction effectue un rollback.**
5. **Initialisation de v\_result:**
   * **La variable v\_result est initialisée en tant qu'ensemble de données vide de type t\_effectif\_table**
6. **Traitement des données en fonction de v\_dr:**
   * **Si v\_dr est égal à 1, la fonction sélectionne toutes les lignes de la table EFFECTIF et les ajoute à v\_result**
   * **Sinon, la fonction sélectionne les lignes de la table EFFECTIF où l'identifiant de direction correspond à celui de l'utilisateur actuel, puis les ajoute à v\_result.**
7. **Retour des résultats:**
   * **Les résultats sont renvoyés à la fin de chaque bloc de condition.**
8. **Fin de la fonction:**
   * **END getEffectifData; marque la fin de la définition de la fonction.**

select \* from table(getEffectifData(:APP\_USER))

select USER\_MAILc1, EFF\_ID c2 from table(getDepData(:APP\_USER)) WHERE USER\_MAIL IS NOT NULL;