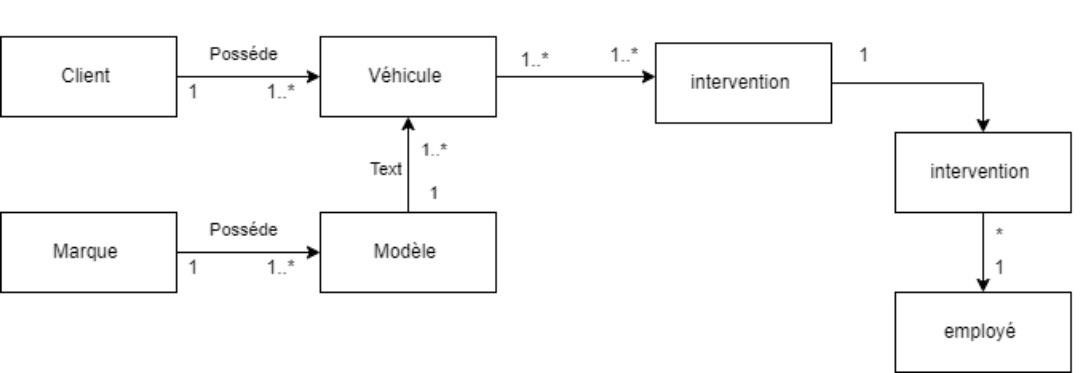
***Bases de Données Avancées Projet : SQL3-Oracle et NoSQL (MongoDB)***

**Partie 1 :** **Relationnel-Objet**

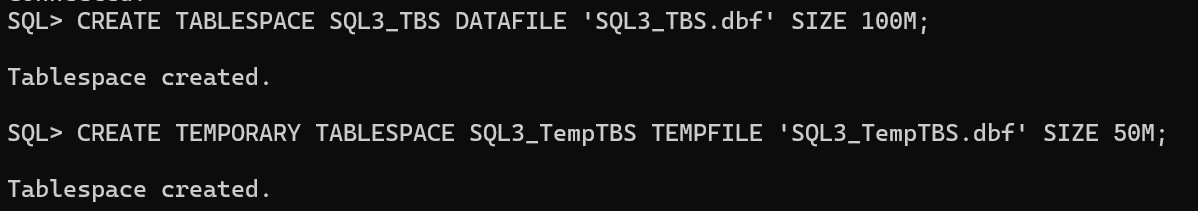
1. ***Modélisation orientée objet***

1) Transformez ce schéma relationnel en un schéma Objet :

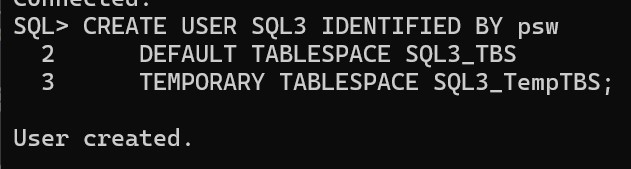


1. ***Création des TableSpaces et utilisateur :***

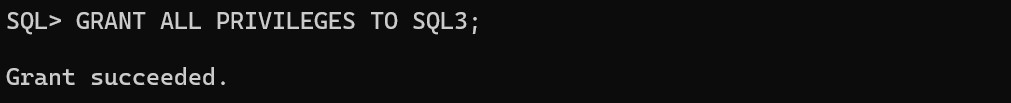
* + 1. Créer deux TableSpaces SQL3\_TBS et SQL3\_TempTBS :



* + 1. Créer un utilisateur SQL3 en lui attribuant les deux tablespaces créés précédemment :



* + 1. Donner tous les privilèges à cet utilisateur :



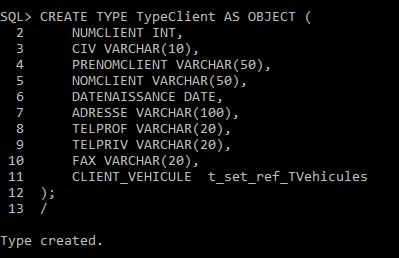
1. ***Langage de définition de données :***

*5)* définir tous les types nécessaires :

* 1. Type client :

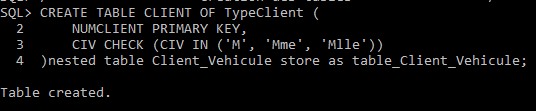
*Ce type représente les détails d'un client, y compris son numéro, son nom, son prénom, sa date de naissance, son adresse et ses coordonnées de contact*.

*Il est utilisé pour créer des enregistrements de clients dans la table* ***client*** *.*



* 1. Table client :

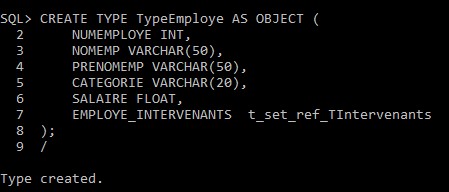
*on a une contrainte sur l’état civile d’un client CIV qui est ( M; Mlle ; Mme) et* nested table stocke une collection de références vers des véhicules associés à chaque client .



* 1. Type Employée :

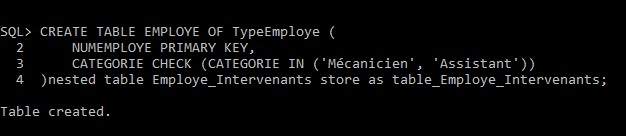
*Ce type contient les informations relatives à un employé, telles que son numéro, son nom, son prénom, sa catégorie d'emploi et son salaire.*

*Il est utilisé pour enregistrer les détails des employés dans la table* ***EMPLOYE***



* 1. Table Employée :

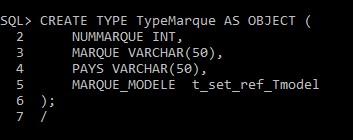
*On a Contrainte de vérification (CHECK) sur la colonne CATEGORIE pour s'assurer que la valeur est soit 'Mécanicien' ou 'Assistant' et nested table qui stocke une collection de références vers des intervenants associés à chaque employé.*



* 1. Type Marque :

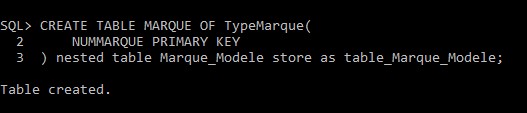
*Ce type représente les caractéristiques d'une marque de véhicule, y compris son numéro, son nom et le pays d'origine.*

*Il est utilisé pour enregistrer les détails des marques dans la table* ***MARQUE***



* 1. Table Marque :

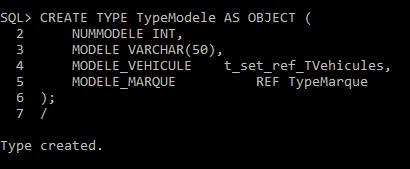
*On a nested table qui stocke une collection de références vers des modèles de véhicules associés à chaque marque.*



* 1. Type Model :

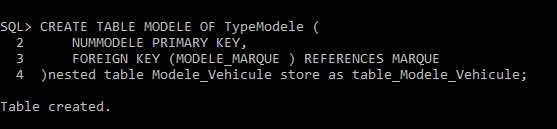
*Ce type contient les attributs d'un modèle de véhicule, tels que son numéro, son nom et la marque a associée.*

*Il est utilisé pour enregistrer les détails des modèles de véhicules dans la table* ***MODELE***



* 1. Table Model :

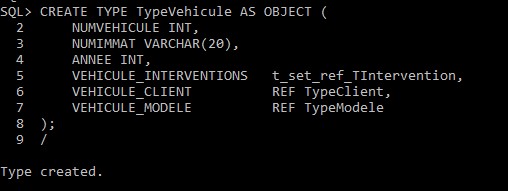
*On a Contrainte de clé étrangère sur la colonne MODELE\_MARQUE faisant référence à la table MARQUE*.et nested table stocke une collection de références vers des véhicules associés à chaque modèle.



* 1. Type Véhicule :

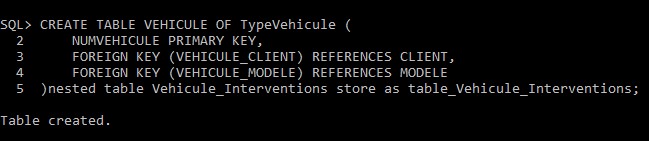
*Ce type représente les détails d'un véhicule, y compris son numéro d'identification, son immatriculation, son année de fabrication et les références aux clients et modèles associés.*

*Il est utilisé pour enregistrer les détails des véhicules dans la table* ***VEHICULE****.*



* 1. Table Véhicule :

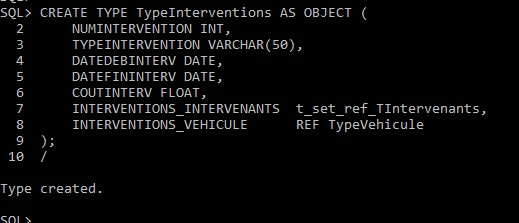
*On a Contrainte de clé étrangère sur la colonne VEHICULE\_CLIENT faisant référence à la table CLIENT et Contrainte de clé étrangère sur la colonne VEHICULE\_MODELE faisant référence à la table MODELE et nested table stocke une collection de références vers des interventions associées à chaque véhicule.*



* 1. Type Intervention :

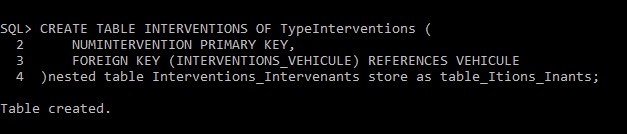
*Ce type contient les informations sur une intervention sur un véhicule, comme le type d'intervention, l les dates de début et de fin, le coût et les références au véhicule concerné.*

*Il est utilisé pour enregistrer les détails des interventions dans la table* ***INTERVENTIONS***



* 1. Table Intervention :

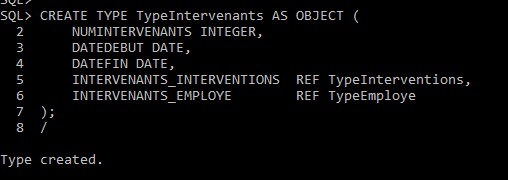
*On a Contrainte de clé étrangère sur la colonne INTERVENTIONS\_VEHICULE faisant référence à la table* ***VEHICULE***et *nested table stocke une collection de références vers des intervenants impliqués dans chaque intervention.*



* 1. Type Intervenant :

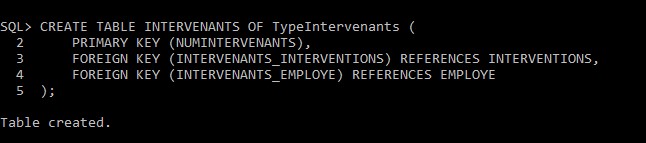
*Ce type représente les intervenants impliqués dans une intervention, avec des détails tels que leur numéro, leurs dates de début et de fin d'intervention, ainsi que leurs références à l'intervention et à l'employé associé.*

*Il est utilisé pour enregistrer les détails des intervenants dans la table* ***Intervenant***



* 1. Table Intervenant :

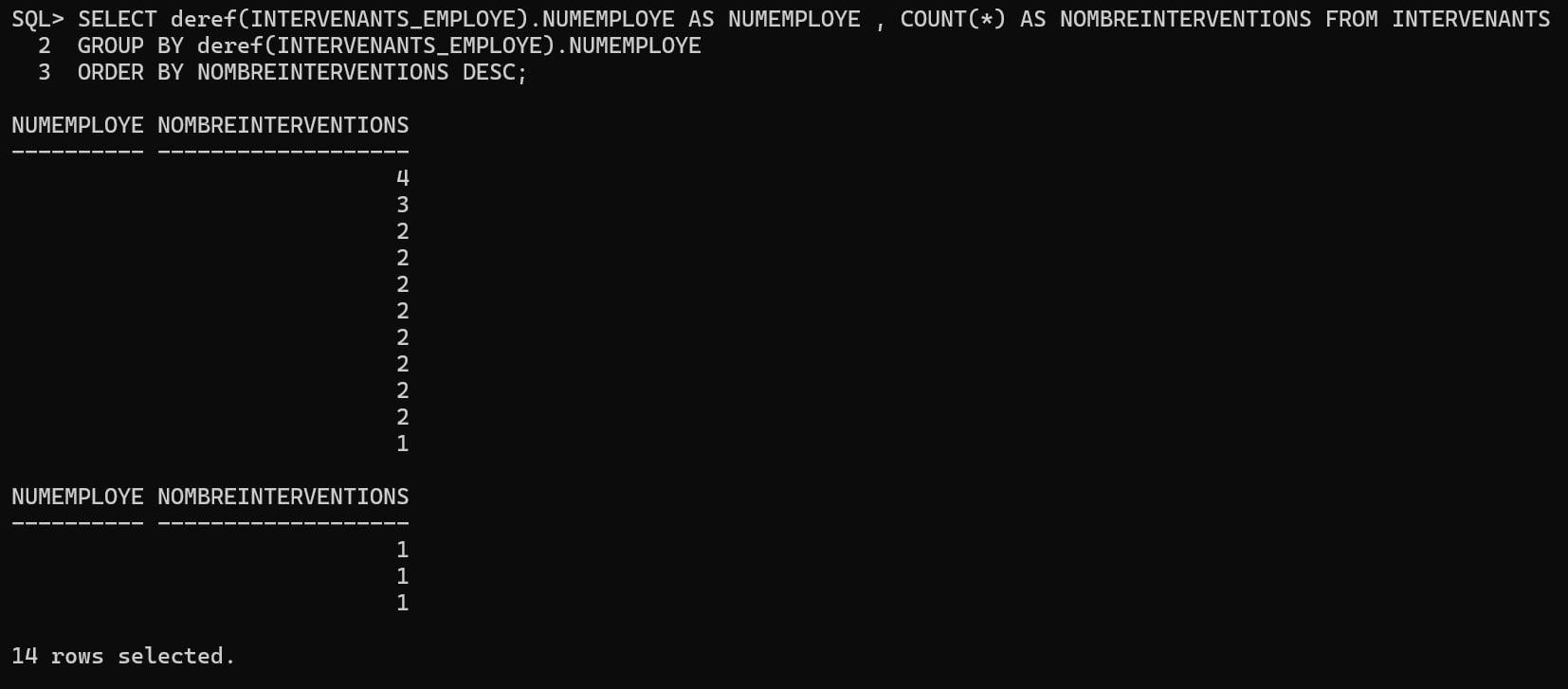
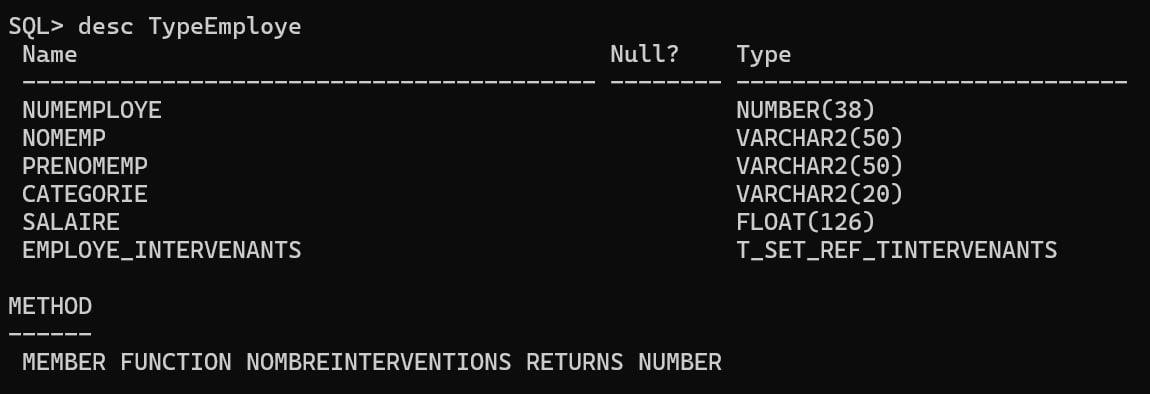
*On a Contrainte de clé étrangère sur la colonne INTERVENANT\_INTERVENTIONS faisant référence à la table* ***INTERVENTIONS***  *Contrainte de clé étrangère sur la colonne INTERVENANT\_EMPLOYE faisant référence à la table* ***EMPLOYE***



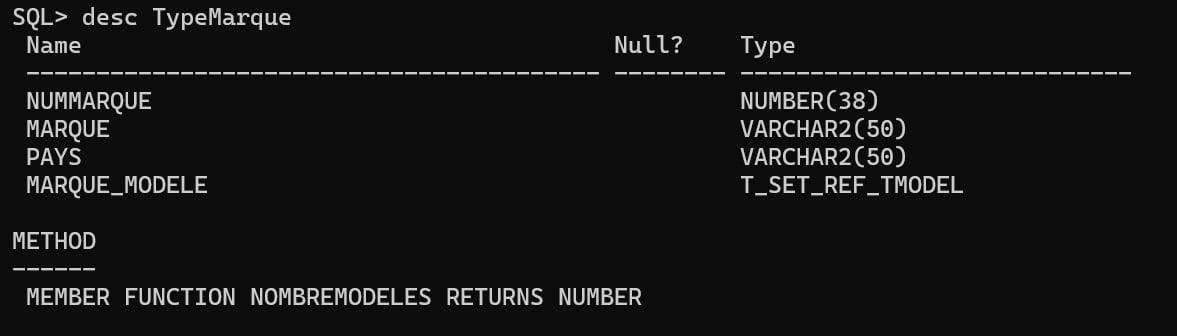
6) Définition des méthodes : qui permettant de :

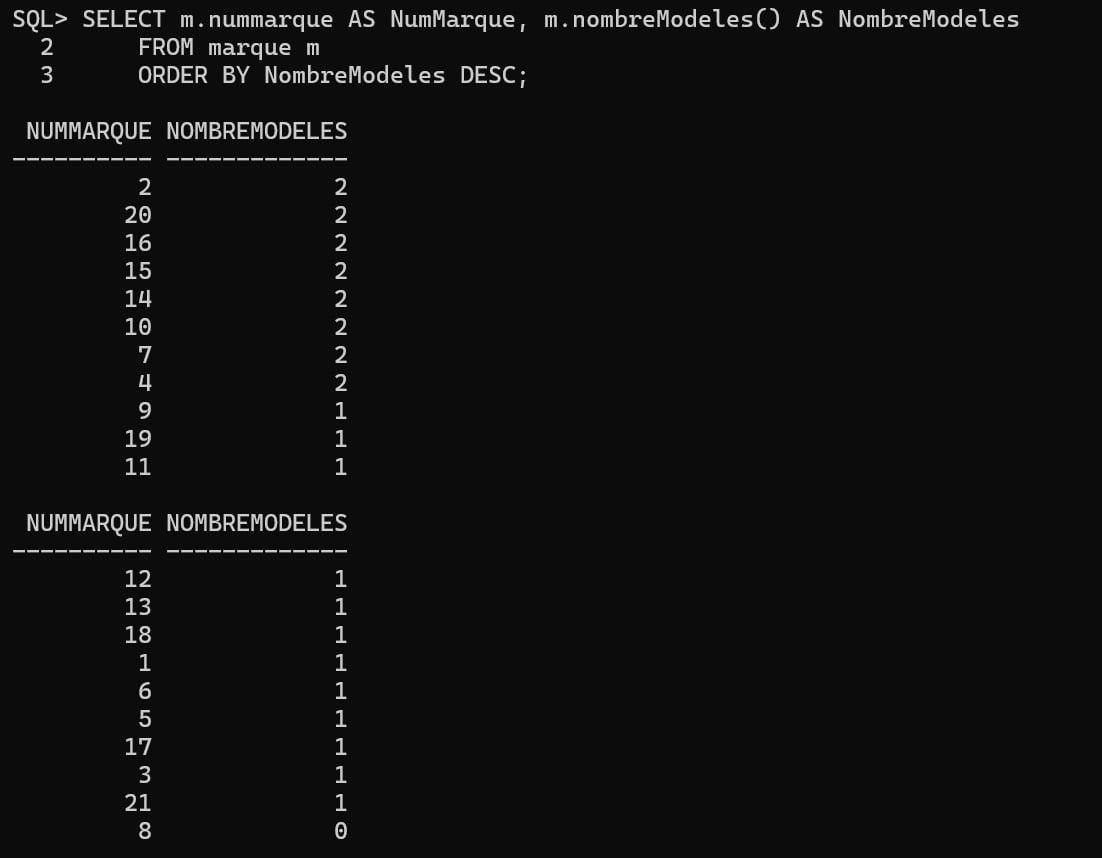
1. Calculer pour chaque employé, le nombre des interventions effectuées :

*Notice : Nous n'allons pas stocker le résultat dans la table 'EMPLOYEE'. Nous allons simplement l'afficher car la modification des tables objets n'est pas autorisée. Nous pourrions créer une autre table pour stocker le résultat, ou bien refaire la manipulation des données sans modifier la structure de la table existante.*

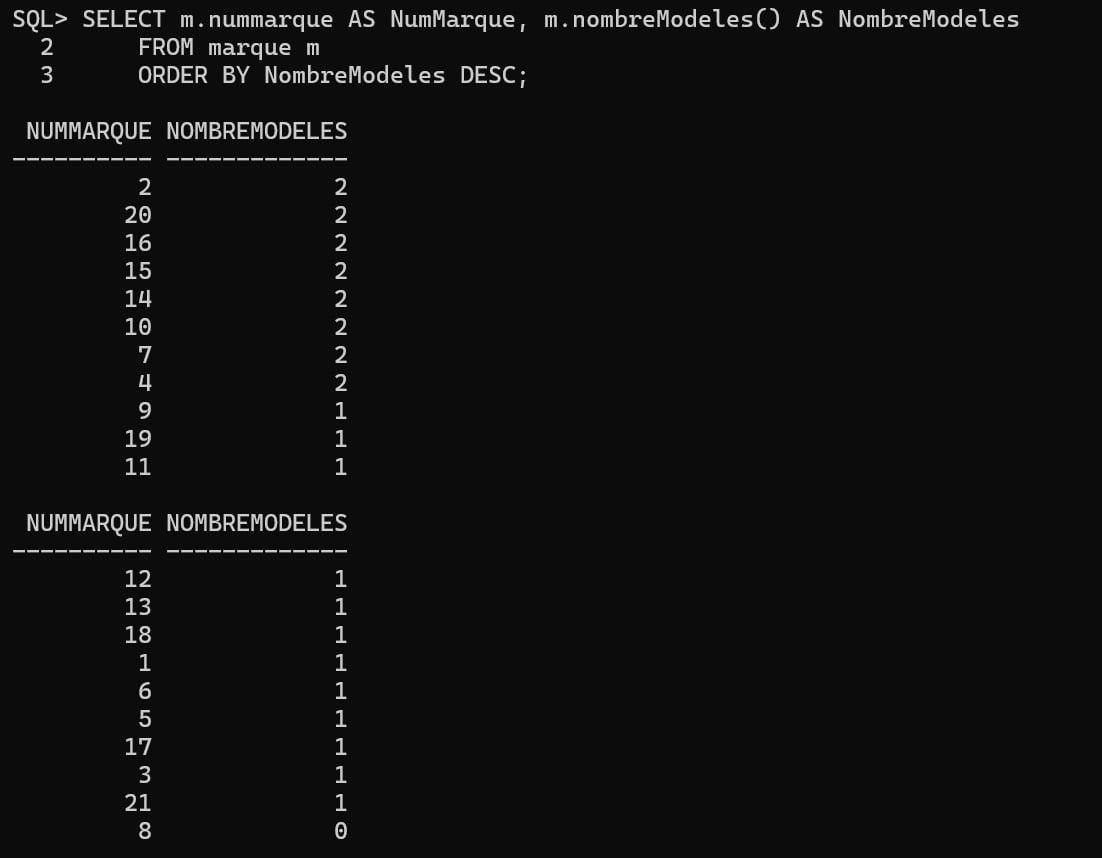
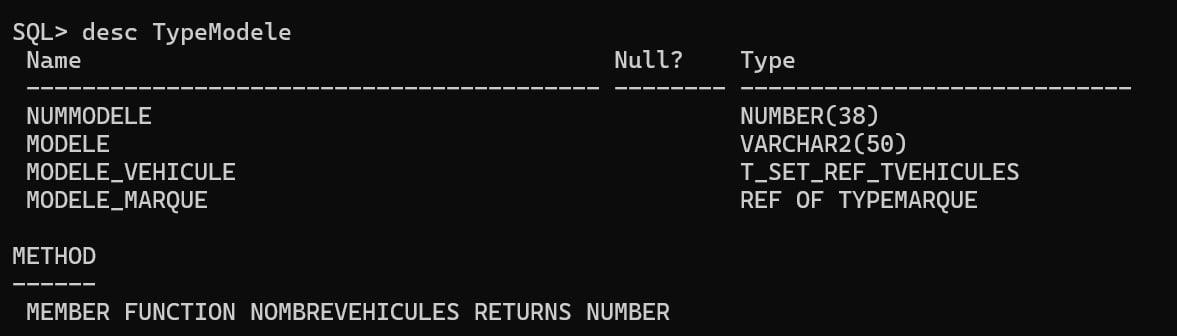


1. Calculer pour chaque marque, le nombre de modèles :

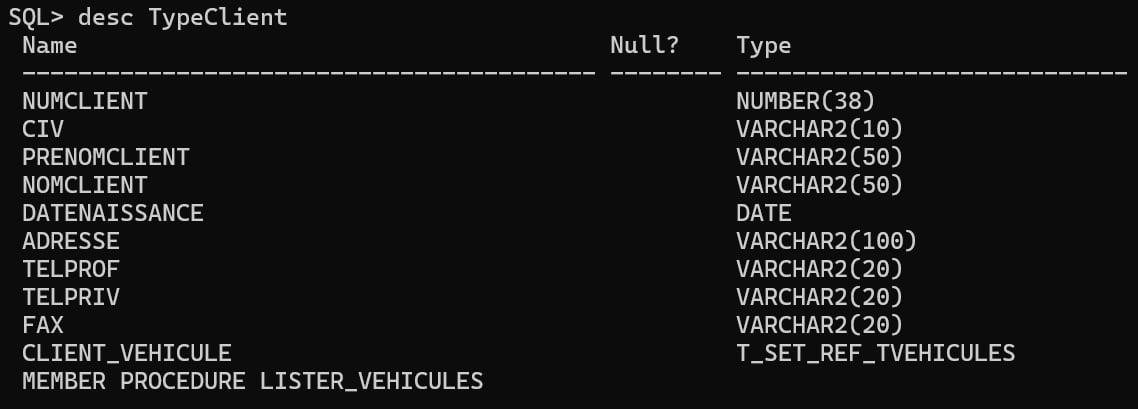




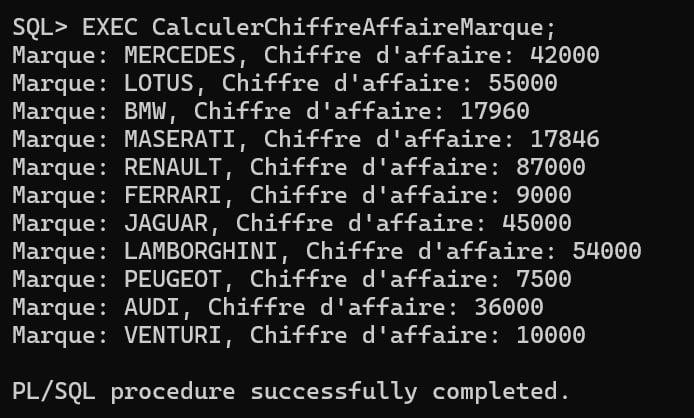
1. Calculer pour chaque modèle, le nombre de véhicules :



1. Lister pour chaque client, ses véhicules :



1. Calculer pour chaque marque, son chiffre d’affaire :



**D) Langage de manipulation de données :**

1. Les caractères spéciaux de SQL (en utilisant oracle) :

*Lors de l'insertion de données dans une base de données SQL, il est important d'éviter d'utiliser certains caractères spéciaux qui pourraient interférer avec la syntaxe SQL ou causer des erreurs d'exécution. Exemple :* ***(‘)***

*Au lieu d’utiliser : 'CITE JEANNE D'ARC ECRAN B2-GAMBETTA – ORAN'*

*On fait : 'CITE JEANNE D''ARC ECRAN B2-GAMBETTA – ORAN' double* ***(‘’)***

*\*\* Utilisez deux slashes inversés (\) pour représenter un slash inversé*

*\*\*…*

1. Les technique d’insertion :

*Dans Oracle SQL, on ne peut pas insérer plusieurs enregistrements en utilisant une seule instruction INSERT INTO ... VALUES. On doit soit répéter l'instruction INSERT INTO ... VALUES pour chaque enregistrement, soit utiliser une autre approche, comme l'instruction INSERT INTO ... SELECT.*

*Exemple :*

***‘’’INSERT INTO CLIENT (NUMCLIENT, CIV, PRENOMCLIENT, NOMCLIENT, DATENAISSANCE, ADRESSE, TELPROF, TELPRIV, FAX)***

***VALUES (23, 'M', 'Lamine', 'MERABAT', TO\_DATE('1965-09-13', 'YYYY-MM-DD'), 'CITE JEANNE D''ARC ECRAN B2-GAMBETTA – ORAN', '0561724538', '0561724538', NULL);’’’*** *OU :*

***‘’’INSERT INTO MARQUE (NUMMARQUE, MARQUE, PAYS)***

***SELECT 1, 'LAMBORGHINI', 'ITALIE' FROM DUAL UNION ALL***

***SELECT 2, 'AUDI', 'ALLEMAGNE' FROM DUAL UNION ALL’’’*** 3) Les erreurs de coherence :

*Par exemple dans la ligne :*

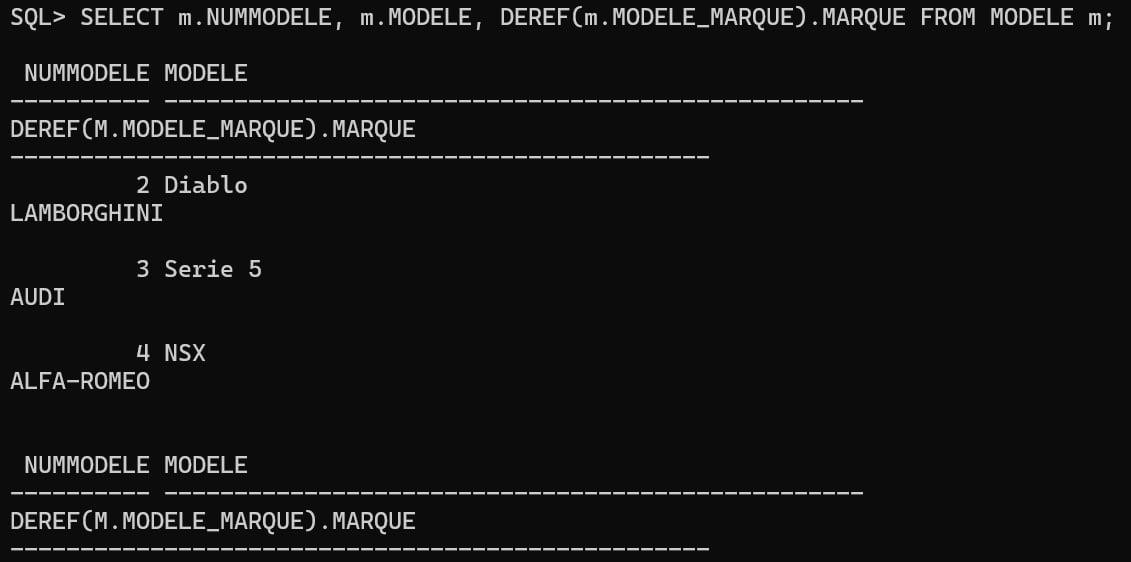
***INSERT INTO VEHICULE (NUMVEHICULE, NUMCLIENT, NUMMODELE, NUMIMMAT, ANNEE) VALUES***

***(24, 80, 9, '1789519816', 1998) ;***

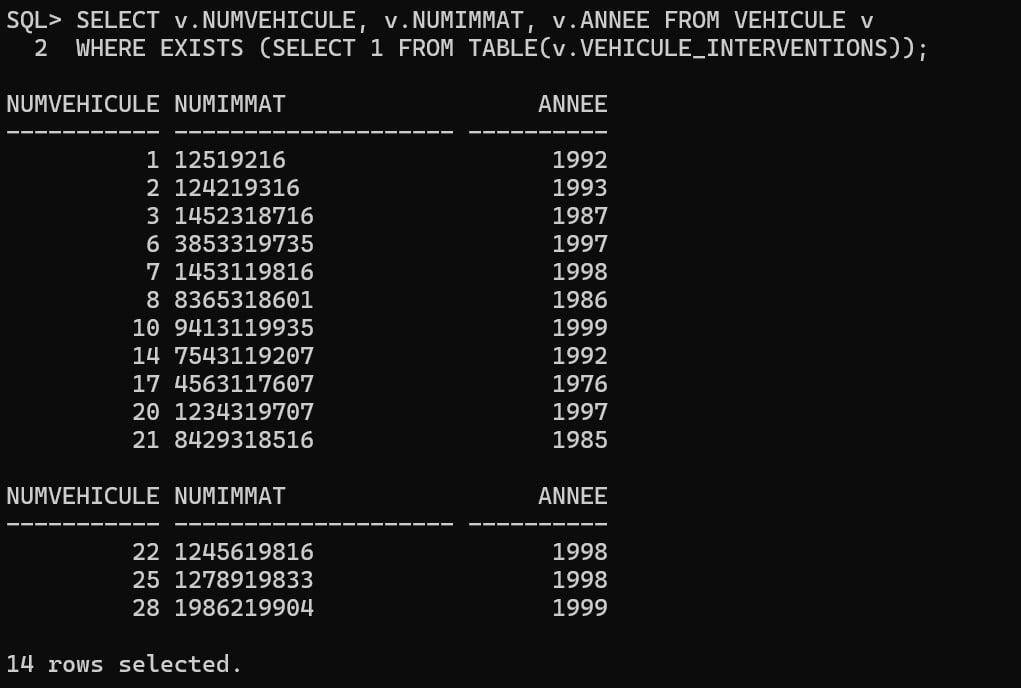
*le client Num° 80 n’existe pas donc on ne peut pas affecter un client qui n’existe pas à un véhicule soit on insert le client 80 dans la table client soit on change le num de client dans l’insertion de véhicule*

***E)* Langage d’interrogation de données *:***

1. Lister les modèles et leur marque : *portion de résultat*



1. Lister les véhicules sur lesquels, il y a au moins une interventi



1. Quelle est la durée moyenne d’une intervention?:

*Notice : on utilise AVG() pou calculer la moyenne d’un intervalle*



1. Donner le montant global des interventions dont le coût d’intervention est supérieur à 30000 DA?:

*Notice : on utilise SUM(colonne) pour faire la sommation de colonne spécifier de chaque ligne*



1. Donner la liste des employés ayant fait le plus grand nombre d’interventions :

*Notice : on utilise COUNT()pour incrémenter par 1 a chaque fois qu’on rencontre le même numemploye*

*Et DESC pour les afficher dans l’ordre décroissant*



# Partie 2: : NoSQL – Modèle orienté « documents »

***A) Modélisation orientée document :***

1. Proposition de modélisation orientée document :

Collections :

vehicules : La collection contient des informations sur un véhicule, y compris les détails de son client, les détails de son modèle et les détails des interventions effectuées sur le véhicule.

CLIENT: Les informations sur le client associé au véhicule

MODELE: Les détails du modèle du véhicule

INTERVENTIONS: Les interventions effectuées sur le véhicule, et les détails des **intervenants** (avec les détails d'**employé**).

1. Illustration :

*Véhicule :*

*{*

*"NUMVEHICULE": 28,*

*"CLIENT": {*

*"NUMCLIENT": 10,*

*"CIV": "Mme",*

*"PRENOM": "Houda",*

*"NOM": "GROUDA",*

*"DATENAISSANCE": "20/02/1950",*

*"ADRESSE": "RUE DR OUIDIR AMAR CITE 1251 LOGTS BT 17 N 8",*

*"TELPROF": "0561385967",*

*"TELPRIV": "0562471956",*

*"FAX": ""*

*},*

*"MODELE": {*

*"NUMMODELE": 3,*

*"MARQUE": {*

*"NUMMARQUE": 2,*

*"MARQUE": "AUDI",*

*"PAYS": "ALLEMAGNE"*

*},*

*"MODELE": "Série 5"*

*},*

*"NUMIMMAT": "5165420013",*

*"ANNEE": 1999,*

*"INTERVENTIONS": [*

*{*

*"NUMINTERVENTION": 11,*

*"TYPEINTERVENTION": "Réparation",*

*"DATEDEBINTERV": "2006-04-08 09:00:00",*

*"DATEFININTERV": "2006-04-09 18:00:00",*

*"COUTINTERV": 36000,*

*"INTERVENANTS": [*

*{*

*"NUMEMPLOYE": 59,*

*"NOMEMP": "BELHAMIDI",*

*"PRENOMEMP": "Mourad",*

*"CATEGORIE": "Mécanicien",*

*"SALAIRE": 19500*

*},*

*{*

*"NUMEMPLOYE": 64,*

*"NOMEMP": "BADI",*

*"PRENOMEMP": "Hatem",*

*"CATEGORIE": "Assistant",*

*"SALAIRE": 14000*

*},*

*{*

*"NUMEMPLOYE": 53,*

*"NOMEMP": "LACHEMI",*

*"PRENOMEMP": "Bouzid",*

*"CATEGORIE": "Mécanicien",*

*"SALAIRE": 25000*

*}*

*]*

*}*

*]*

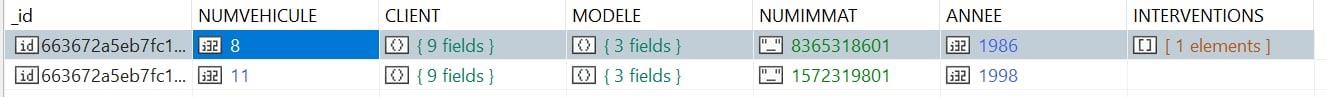
*}*

***3.*** *Justification de la conception orientée document :*

* **Simplicité de la structure** : Les données sont représentées de manière hiérarchique et autoexplicative, ce qui facilite la compréhension et la manipulation.
* **Flexibilité de schéma** : Les documents peuvent avoir des structures différentes, ce qui permet d'ajouter ou de modifier des champs sans avoir à altérer la structure de base.
* **Performance des requêtes** : Pour les requêtes principalement axées sur les véhicules, leurs marques et les interventions par les employés, les requêtes orientées document sont souvent plus efficaces car elles récupèrent l'ensemble de l'objet avec toutes ses données en une seule opération.
* **Capacité à représenter des relations imbriquées** : Les relations entre les clients, les véhicules, les marques, les modèles et les interventions sont représentées de manière naturelle grâce à l'imbrication des documents.

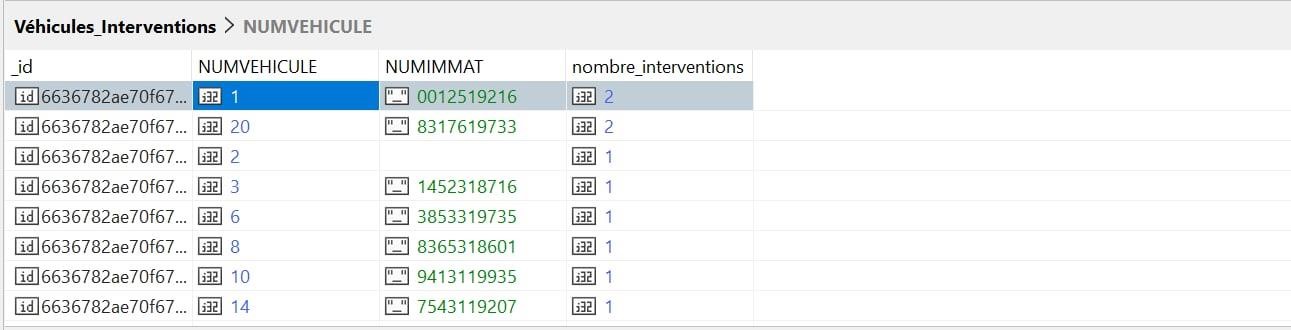
1. ***Les requêtes :*** 
   1. *Afficher tous les véhicules de la marque «PORSCHE» :*





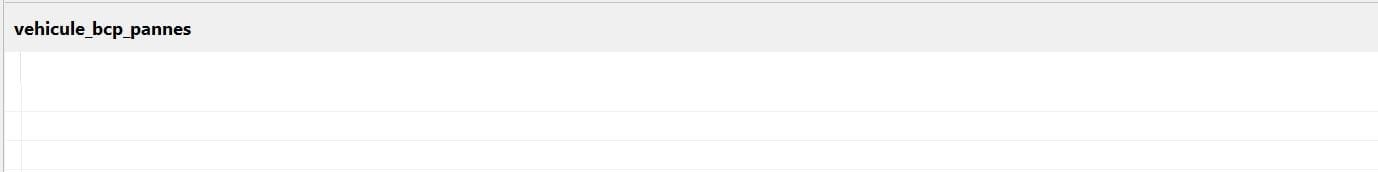
* 1. *Récupérer dans une nouvelle collection Véhicules\_Interventions, les matricules des véhicules et le nombre total de s interventions par véhicule ; la collection devra être ordonnée par ordre décroissant du nombre des interventions.*



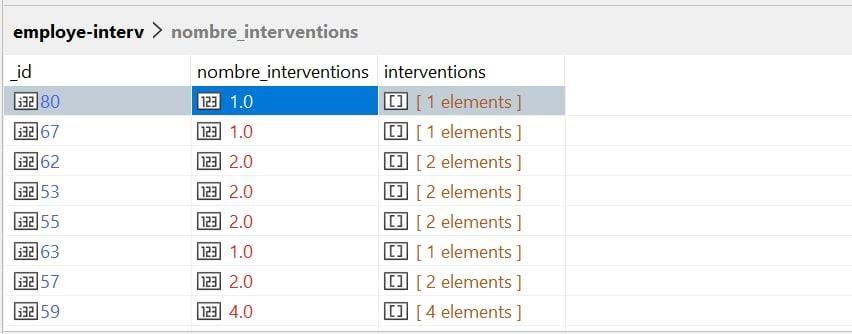


* 1. *Dans une collection véhicule\_bcp\_pannes, récupérer les véhicules dont le nombre des interventions dépasse 6 pannes.*

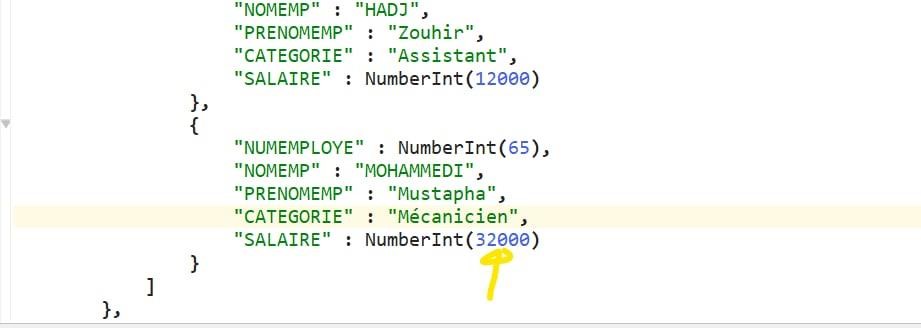
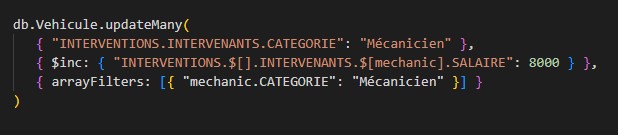




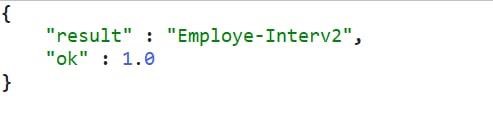
* 1. *Récupérer dans une collection employe-interv, toutes les interventions d’un employé.*



* 1. *Augmenter de 8000DA, le salaire des employés de catégorie « Mécanicien» :*



* 1. *Reprendre la 4ième requête à l’aide du paradigme Map-Reduce.*



1. **Analyse :**

*Les requêtes présentées couvrent une variété de besoins de manipulation et de traitement de données dans le contexte d'une base de données orientée document. Voici quelques observations générales :*

* **Lisibilité des données** : La modélisation orientée document permet une représentation intuitive des données, ce qui facilite la compréhension et l'analyse.
* **Flexibilité de requête** : Les requêtes peuvent être formulées de manière flexible en parcourant les documents et en appliquant des filtres, des agrégations et des tris selon les besoins spécifiques.
* **Performance des requêtes simples** : Les opérations de lecture simples, telles que l'affichage des véhicules d'une marque particulière, sont généralement efficaces car elles impliquent un accès direct aux documents correspondants.
* **Complexité des requêtes avancées** : Les requêtes qui nécessitent des agrégations, des tris ou des filtrages complexes peuvent être plus exigeantes en termes de performance et de ressources, en particulier lorsque les données sont volumineuses.
* **Opérations de transformation des données** : Certaines requêtes, comme l'augmentation des salaires des employés ou la création de nouvelles collections basées sur des critères spécifiques, impliquent des opérations de modification ou de création de données.
* **Utilisation de paradigmes de traitement distribué** : Pour les requêtes complexes ou nécessitant un traitement massivement parallèle, des paradigmes comme Map-Reduce peuvent être utilisés pour améliorer les performances et la scalabilité des opérations.

*En résumé, la modélisation orientée document offre une flexibilité et une expressivité qui permettent de répondre à une variété de besoins en matière de manipulation et de traitement de données, bien que des considérations de performance et de complexité doivent être prises en compte lors de la conception et de l'exécution des requêtes.*