Document Technique

Giulia Governatori - PROJET 3 ¨Requêtez une base de données avec SQL¨

1. Introduction

Ce projet a pour objectif de créer et gérer une base de données pour analyser le marché des assurances habitation en France.

Objectifs

- Comprendre les types de données utilisées.
- Créer un schéma relationnel pour structurer les données.
- Charger les données dans un SGBD et vérifier leur intégrité.
- Réaliser des analyses avec des requêtes SQL.

2. Exploration des données

2.1 Fichiers utilisés

A. "**CONTRAT.CSV**" : Données sur les contrats d'assurance habitation (30 335 contrats clients)
B. "**REGION.CSV**" : Référentiel des régions françaises extrait de *data.gouv.fr*
(38 916 code_dep_code_commune distinctes)

2.2 Dictionnaire des données

A. "CONTRAT.CSV":

Nom des colonnes	Type de données	Taille	Clé	Description
Contrat_ID	VARCHAR	8	PK	Id unique pour les contrats. Les données sont seulement des chiffres avec une taille de 6.
No_voie	VARCHAR	5		Numéro de voie pour l'adresse du logement assuré.
B_T_Q	VARCHAR	1		Indicateur éventuel de répétition pour l'adresse du logement assuré sur un caractère
Type_de_voie	VARCHAR	5		Abbreviation du type de voie pour l'adresse du logement assuré: RUE, AV (Avenue), QUAI, ecc
Voie	VARCHAR	40		Libellé de la voie pour l'adresse du logement assuré
Code_dep_code_commune	VARCHAR	6	PFK	Concaténation du code départemental officiel avec le code commune pour avoir une clé unique
Code_postal	CHAR	5		Code postal pour l'adresse du logement assuré
Surface	MEDIUMINT UNSIGNED	5		La surface du logement assuré en mètres carrés
Type_local	VARCHAR	20		Le type du bien assuré (appartement, maison.. Ecc)
Occupation	VARCHAR	35		La personne qui occupe le bien (locataire, propriétaire… etc.)
Type_contrat	VARCHAR	50		Le type de contrat que l'assurance a stipulé avec le client.
Formule	VARCHAR	15		La formule choisie par le client avec le contrat stipulé
Valeur_declaree_biens	VARCHAR	12		La valeur déclarée pour les biens.
Prix_cotisation_mensuel	MEDIUMINT UNSIGNED	5		Le prix de la cotisation mensuelle pour le client

CONSIDERATIONS SUR LES DONNÉES DU “CONTRAT.CSV”:

1. **Contrat_ID:**
   - Taille: Nous ne savons pas si il y a une contrainte sur la taille pour les futurs contrats. Si l’ID vient déclaré d’une taille fixe, "Taille 6 CHAR” serait mieux.
   - Type de données: VARCHAR pour la même raison que pour la taille.

2. **No_voie:**
   - Taille: Taille 5 selon le système français, mais ça pourrait être 4
   - Type de données: VARCHAR parce que il pourrait contenir des lettres: "12b"

3. **B_T_Q:**
   - Taille: Taille 1 parce que c'est un système officiel qui prévoit 1 seule lettre
   - Type de données: VARCHAR parce que même si dans le système français c’est juste une lettre, il peut être NULL.

4. **Type_de_voie:**
   - Taille: Taille max des abréviations trouvées 4 caractèrs
   - Type de données: VARCHAR parce que les données n'ont pas la même taille (2, 3, 4)

5. **Voie:**
   - Taille: La taille max des données actuelles est de 26
   - Type de données: VARCHAR parce que le libellé est très variable

6. **Code_dep_code_commune:**
   - Taille: officiellement un max de 6 chiffres pour la France
   - Type de données: VARCHAR parce qu’il peut avoir juste 4 chiffres

7. **Surface:**
   - Taille: Taille max des données actuelles est de 3
   - Type de données: UNSIGNED parce que la superficie est une donnée quantitative uniquement positive, et MEDIUMINT pourrait représenter une limite maximale réaliste pour cela.

8. **Type_local:**
   - Taille: Taille 20 imaginant l'insertion de "Immeuble de rapports"
   - Type de données: VARCHAR parce que le type du bien est très variable

9. **Formule:**
   - Taille: Taille max des données actuelles est de 9
   - Type de données: VARCHAR parce que c’est des mots.

10. **Valeur_declaree_biens:**
    - Taille: ce sont des tranches déjà définies par l'assurances, il n'y a pas de libre insertion de la valeur, et sa taille max est 12
    - Type de données: VARCHAR pour la même raison que pour la taille.

11. **Prix_cotisation_mensuel:**
    - Taille: Nous ne savons pas si il y a une contrainte sur la taille pour les futurs contrats.
    - Type de données: VARCHAR parce que même si les tranches sont déjà déclarées, leur taille est variable.

B. “REGION.CSV”:

Nom des colonnes	Type de données	Taille	Clé	Description
Code_dep_code_commune	VARCHAR	6	PK	Concaténation du code départemental officiel avec le code commune pour avoir une clé unique
reg_code	VARCHAR	2		L'identifiant régional officiel sans préfixe.
reg_nom	VARCHAR	26		Le nom officiel de la région
aca_nom	VARCHAR	30		Le centre académique de la zone geographique.
dep_nom	VARCHAR	43		Le nom officiel du département
com_nom_maj_court	VARCHAR	45		Le nom de la ville
dep_code	VARCHAR	3		Le code départamental officiel
dep_nom_num	VARCHAR	28		Le nom du département avec le code départamental (redondance: c'est la concatenation de "dep_nom" + "dep_code")

CONSIDERATIONS SUR LES DONNÉES DE “REGION.CSV”:

1. **Code_dep_code_commune:**
   - Taille: officiellement un max de 6 chiffres pour la France
   - Type de données: VARCHAR parce qu’il peut y avoir avec juste 4 chiffres

2. **reg_code:**
   - Taille: toujours composé par 2 chiffres
   - Type de données: VARCHAR parce qu’il peut y avoir un seul caractère

3. **reg_nom:**
   - Taille: les régions ne changent pas et le max est 26: "Provence-Alpes-Côte d'Azur"
   - Type de données: VARCHAR parce que le nom de la région est très variable

4. **aca_nom:**
   - Taille: Taille possible 24: "Saint-Étienne-du-Rouvray". Taille max actuelle: 16
   - Type de données: VARCHAR parce que le nom de l'académie est très variable

5. **dep_nom:**
   - Taille: Les départements ne changent pas, et la taille max est 43: "Terres australes et antarctiques françaises"
   - Type de données: VARCHAR parce que le nom du département est très variable

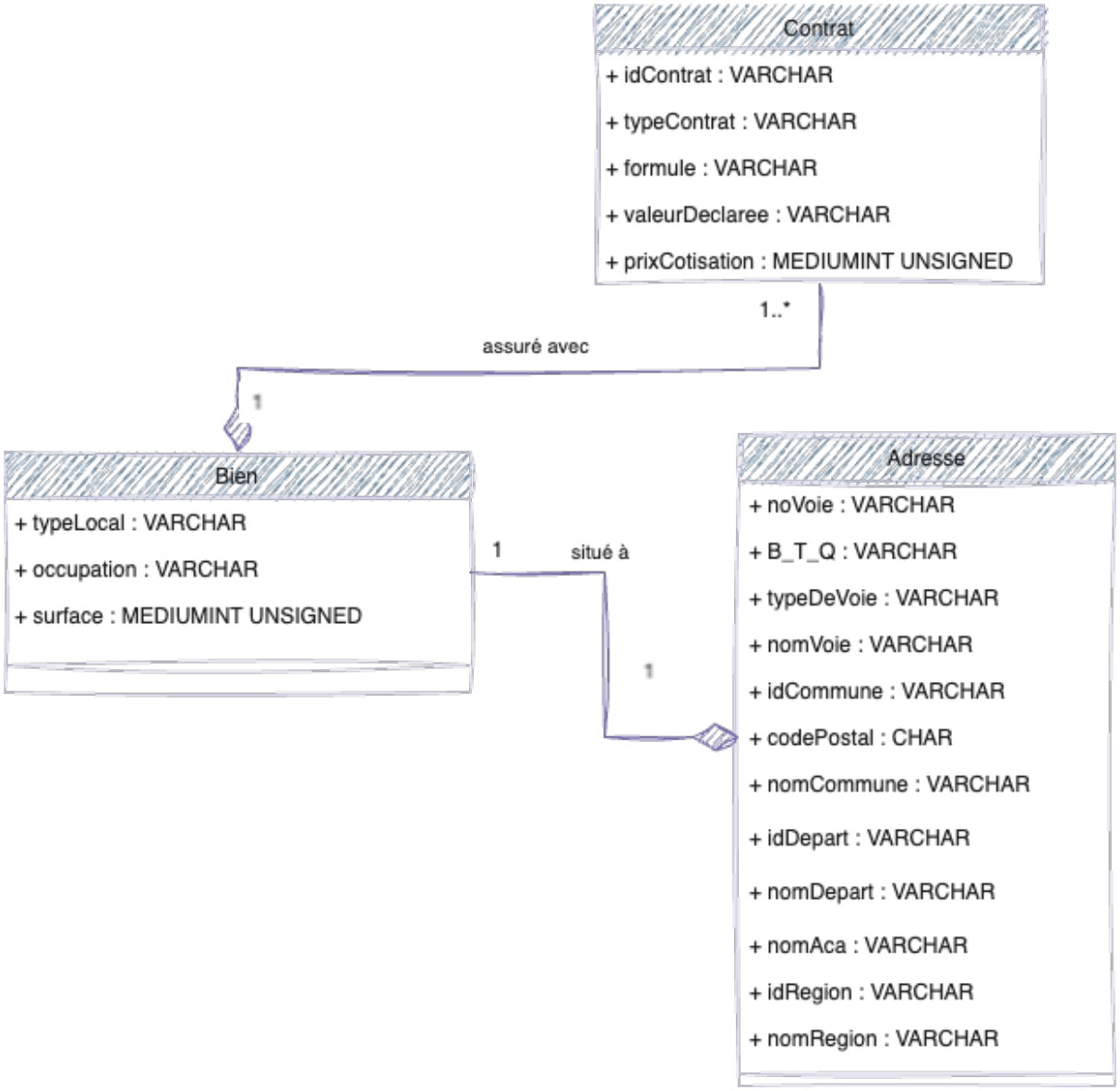
6. **com_nom_maj_court:**
   - Taille: Taille possible 45: "Saint-Remy-en-Bouzemont-Saint-Genest-et-Isson"
   - Type de données: VARCHAR parce que le nom de la ville est très variable

7. **dep_code:**
   - Taille: composé par 2, max 3 characters
   - Type de données: VARCHAR parce que il y a aussi 3 characters pour la Guyane

8. **dep_nom_num:**
   - Taille: 23 ("dep_nom") +2 ("dep_code") +3 (characters ajoutés)
   - Type de données: VARCHAR parce que même si la taille de "dep_code" ne varie pas, celle de "dep_nom" le fait.

3. Modèle conceptuel UML et la structure des tables

Dans cette étape, j'ai commencé par élaborer un modèle conceptuel des données en utilisant UML. Cette analyse initiale m'a conduit à envisager une structure avec trois tables, ce qui aurait permis de mieux représenter certaines relations et détails des données fournies.



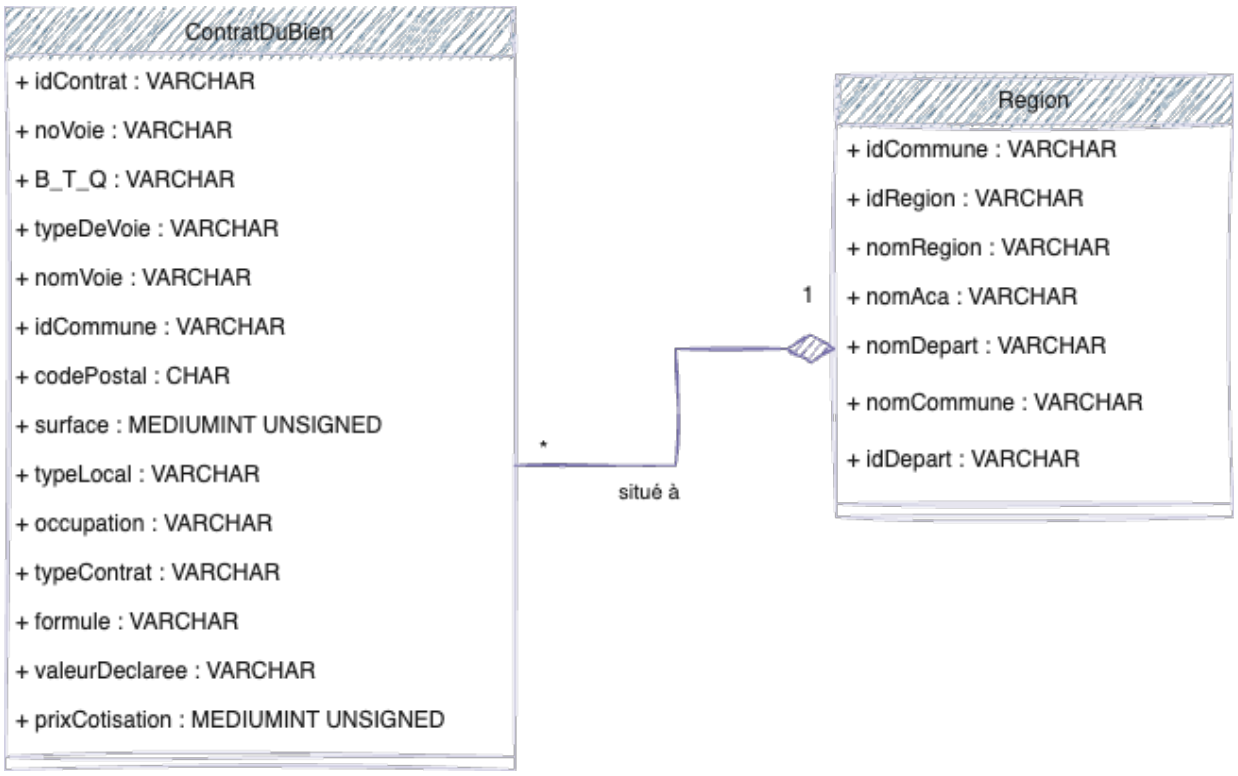
Cependant, après réflexion, j'ai décidé de revenir à la structure demandée dans le projet, avec seulement deux tables.

Cette décision a été motivée par deux raisons principales :

1. Le projet spécifie clairement une structure à deux tables, et j'ai souhaité respecter cette consigne pour m'aligner sur les attentes.
2. Modifier les fichiers de données fournis aurait nécessité un travail de révision important, risquant de m'éloigner des directives de base et de la méthodologie guidée proposée.

Ainsi, j'ai poursuivi le projet avec la structure simplifiée à deux tables, tout en conservant les apprentissages issus de cette première étape d'exploration conceptuelle.

Ces les modèles ont été créés avec draw.io



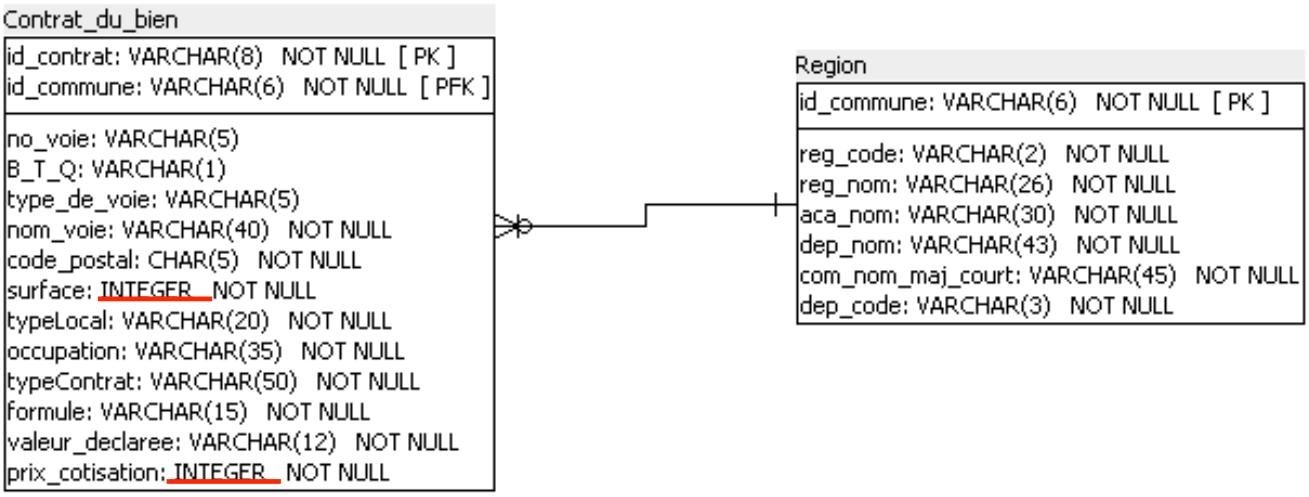
4. Création du schéma relationnel avec SQL Power Architect

Le schéma relationnel a été conçu en utilisant l'outil *SQL Power Architect*.
Bien que cet outil soit très utile pour la modélisation initiale, il présente certaines limitations techniques.

1. il ne permet pas de modifier certains types de données par défaut, comme passer de INTEGER à MEDIUMINT UNSIGNED.
2. De même, il n'est pas possible d'ajouter directement l'attribut UNIQUE aux clés primaires à ce stade de la conception.

Ces limitations ont été prises en compte, et j'ai apporté des ajustements manuels au code SQL généré par l'outil lors de la création des tables dans le système de gestion choisi.

Néanmoins, l'image ci-dessous représente le schéma relationnel tel qu'il a été généré par l'outil, avant ces modifications.



3.1 Schéma relationnel normalisé

Le schéma relationnel produit a été normalisé en respectant la troisième forme normale (**3NF**) pour garantir une structure optimale et cohérente des données.

3.3 Code SQL pour créer les tables

Voici le code SQL modifié pour créer la base de données dans l'SGBDR.

```sql
CREATE TABLE Region (
                id_commune VARCHAR(6) NOT NULL UNIQUE,
                reg_code CHAR(2) NOT NULL,
                reg_nom VARCHAR(26) NOT NULL,
                aca_nom VARCHAR(30) NOT NULL,
                dep_nom VARCHAR(43) NOT NULL,
                com_nom_maj_court VARCHAR(45) NOT NULL,
                dep_code VARCHAR(3) NOT NULL,
                PRIMARY KEY (id_commune)
);

CREATE TABLE Contrat_du_bien (
                id_contrat VARCHAR(8) NOT NULL UNIQUE,
                id_commune VARCHAR(6) NOT NULL,
                no_voie VARCHAR(5),
                B_T_Q VARCHAR(1),
                type_de_voie VARCHAR(5),
                nom_voie VARCHAR(40) NOT NULL,
                code_postal CHAR(5) NOT NULL,
                surface MEDIUMINT UNSIGNED NOT NULL,
                typeLocal VARCHAR(20) NOT NULL,
                occupation VARCHAR(35) NOT NULL,
                typeContrat VARCHAR(50) NOT NULL,
                formule VARCHAR(15) NOT NULL,
                valeur_declaree VARCHAR(12) NOT NULL,
                prix_cotisation MEDIUMINT UNSIGNED NOT NULL,
                PRIMARY KEY (id_contrat, id_commune)
);

ALTER TABLE Contrat_du_bien ADD CONSTRAINT region_contrat_du_bien_fk
FOREIGN KEY (id_commune)
REFERENCES Region (id_commune)
ON DELETE CASCADE
ON UPDATE CASCADE;
```

3.4 Les Schémas des deux tables:

```
+-----------------+--------------------+------+-----+---------+-------+
| Field           | Type               | Null | Key | Default | Extra |
+-----------------+--------------------+------+-----+---------+-------+
| id_contrat      | varchar(8)         | NO   | PRI | NULL    |       |
| id_commune      | varchar(6)         | NO   | PRI | NULL    |       |
| no_voie         | varchar(5)         | YES  |     | NULL    |       |
| B_T_Q           | varchar(1)         | YES  |     | NULL    |       |
| type_de_voie    | varchar(5)         | YES  |     | NULL    |       |
| nom_voie        | varchar(40)        | NO   |     | NULL    |       |
| code_postal     | char(5)            | NO   |     | NULL    |       |
| surface         | mediumint unsigned | NO   |     | NULL    |       |
| typeLocal       | varchar(20)        | NO   |     | NULL    |       |
| occupation      | varchar(35)        | NO   |     | NULL    |       |
| typeContrat     | varchar(50)        | NO   |     | NULL    |       |
| formule         | varchar(15)        | NO   |     | NULL    |       |
| valeur_declaree | varchar(12)        | NO   |     | NULL    |       |
| prix_cotisation | mediumint unsigned | NO   |     | NULL    |       |
+-----------------+--------------------+------+-----+---------+-------+
14 rows in set (0.01 sec)

mysql> SHOW COLUMNS FROM region;
+-------------------+-------------+------+-----+---------+-------+
| Field             | Type        | Null | Key | Default | Extra |
+-------------------+-------------+------+-----+---------+-------+
| id_commune        | varchar(6)  | NO   | PRI | NULL    |       |
| reg_code          | char(2)     | NO   |     | NULL    |       |
| reg_nom           | varchar(26) | NO   |     | NULL    |       |
| aca_nom           | varchar(30) | NO   |     | NULL    |       |
| dep_nom           | varchar(43) | NO   |     | NULL    |       |
| com_nom_maj_court | varchar(45) | NO   |     | NULL    |       |
| dep_code          | varchar(3)  | NO   |     | NULL    |       |
+-------------------+-------------+------+-----+---------+-------+
```

4. Création et chargement de la base de données

4.1 Système choisi

SGBDR utilisé : **MySQL:** ligne de commande simple, et en suite ***MySQL WORKBENCH***.

4.2 Identification de la directory sécurisée pour le chargement des données

MySQL utilise la variable *secure_file_priv* pour limiter les opérations de lecture et d'écriture de fichiers à un répertoire spécifique, renforçant ainsi la sécurité en empêchant l'accès non autorisé à d'autres parties du système.

J'ai identifié ce répertoire sécurisé en exécutant la commande suivante:

```
1 SHOW VARIABLES LIKE 'secure_file_priv';
```

```
1 +------------------+-----------------------------------------------------+
2 | Variable_name    | Value                                               |
3 +------------------+-----------------------------------------------------+
4 | secure_file_priv | C:\ProgramData\MySQL\MySQL Server 8.0\Uploads\ |
5 +------------------+-----------------------------------------------------+
```

Ensuite, j'ai copié mes fichiers dans ce répertoire pour pouvoir les charger dans ma base de données en toute sécurité.

4.2 Correction des Incohérences dans les Données Initiales

Un problème a été détecté dans les données initiales : la clé primaire étrangère (*code_dep_code_commune*) ne correspondait pas à la clé primaire de la table “REGION” dans trois cas spécifiques (*97434, 97460, 97470*). À la place du code commune, le code postal avait été inséré par erreur. Cette incohérence empêchait l’insertion des données.

Pour résoudre ce problème, les valeurs ont été corrigées manuellement. Cela a été fait en recherchant d’abord le code postal correspondant à chaque ville, puis en identifiant le code commune correct dans le document de référence “REGION”.

1. Créer une autre feuille “donneesCorrompues”;
2. copier tous les PK de “region.csv” et les PFK de “contrat.csv” sur deux colonnes A et B;
3. sur la troisième colonne appliquer la formule VLOOKUP:

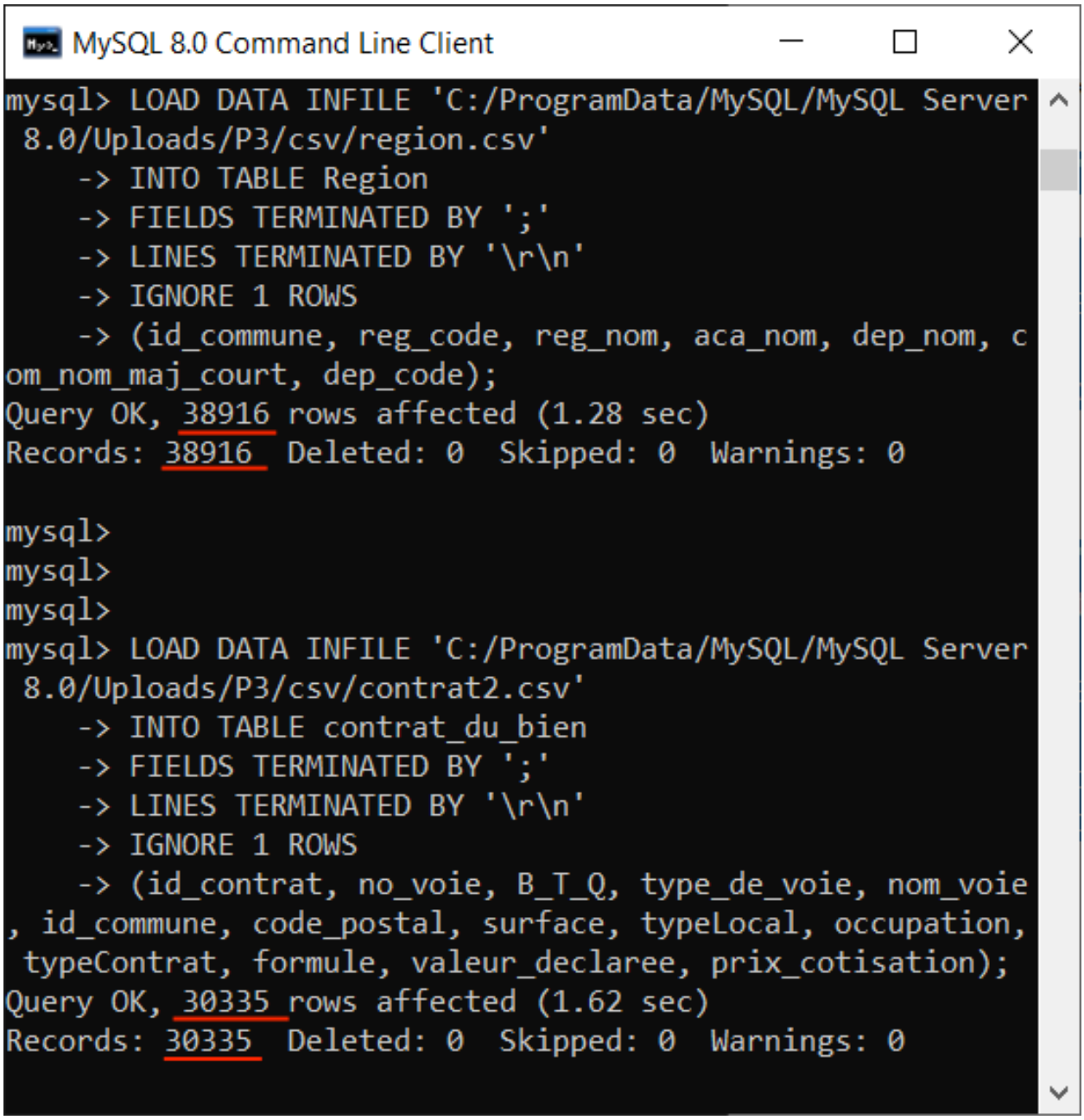
```
=VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])
```

4. on applique un filtre pour les erreurs #N/D: il y a 3 code postales;
5. on les cherche d’abord sur internet: ce sont les communes de *Saint-Paul* (La Saline-les-Bains), *Saint-Paul* (La Saline ou Saint-Gilles-les-Hauts) et Saint-Benoît.
6. on change les données erronées avec celles qui correspondent à leur communes dans le fichier “region.csv”.

PK Region	PFK Contrat	Correspondan
61378	97460	#N/D
61379	97434	#N/D
61380	97470	#N/D
61381	97460	#N/D
61382	97434	#N/D
61385	97434	#N/D
61387	97434	#N/D
61389	97460	#N/D
61391	97460	#N/D

4.3 Chargement des données

- Les fichiers CSV ont été chargés dans deux tables : “contrat_du_bien” et “region”.
- Vérification de l’intégrité :
  - **contrat_du_bien** : 30 335 lignes insérées.
  - **region** : 38 916 lignes insérées.



“*INTO TABLE*: spécifie dans quelle table insérer les données.
“*FIELDS TERMINATED BY*”: spécifie le caractère qui sépare chaque champ dans le fichier CSV.
“*LIGNE TERMINATED BY*”: spécifie le caractère qui sépare chaque ligne dans le fichier CSV.
“*IGNORE 1 ROWS*”: les premières lignes des fichiers CSV doivent être ignorées parce que ce sont les lignes des titres des colonnes.

6. Conclusion

6.1 Résumé

Ce projet a permis de :

- Comprendre et structurer les données d'assurance habitation.
- Créer une base de données fonctionnelle et bien modélisée.
- Réaliser des analyses pertinentes pour l'entreprise.

6.2 Résultats de l'Analyse

Voici les principaux résultats obtenus :

1. **Contrats à Caen** : 4 contrats couvrent des biens avec une superficie allant de 20 m² à 99 m².
2. **Maisons dans le département 71** : 48 contrats ont été identifiés, et la plupart en résidence principale.
3. **Référentiel géographique** : En plus des 18 régions, le référentiel ajoute une entité désignée comme *Collectivités d'Outre-Mer*.
4. **Superficies maximales** : Les 5 contrats avec les superficies les plus élevées vont de 815 m² à 559 m².
5. **Prix moyen des cotisations mensuelles** : Le montant moyen est de **19 euros**.
6. **Valeurs déclarées des biens** : La majorité des contrats concernent des biens d'une valeur déclarée inférieure à 25 000 euros. Seuls 104 contrats dépassent 100 000 euros.
7. **Formules intégrales dans les Pays de la Loire** : 589 contrats utilisent cette formule.
8. **Contrats de maisons dans le département 71** : 4 contrats ont été identifiés, répartis équitablement entre les formules classique et intégrale.
9. **Superficies moyennes des appartements à Paris** : la superficie moyenne des biens à Paris est de 52 m2
10. **Prix moyen des cotisations mensuelles par département** : Paris arrive en tête avec une moyenne de **36 euros** par mois, suivi des Hauts-de-Seine (26 euros) et du Val-de-Marne (20 euros).
11. **Commune avec le plus de contrats** : Paris se distingue comme la commune ayant le plus grand nombre de contrats.
12. **Région avec le plus de contrats** : L'Île-de-France domine avec **14 177 contrats enregistrés**.

Ces résultats mettent en lumière des tendances clés du marché des assurances habitation, telles que la répartition géographique des contrats, les caractéristiques des biens assurés et les différences de cotisation mensuelle. Ils constituent une base solide pour des analyses futures plus ciblées et détaillées.