

# Méthodologie

Présentation de la méthodologie.





# Etape 01 : dictionnaire de données 1/3

## Méthode de manipulation des données



Enregistrement automatique Emmanuel\_Marchand\_Données\_rerentiel - Enregistré

Fichier Accueil Insertion Mise en page Formules Données Révision Affichage Automate Aide Acrobat Création de tableau Requête Commentaires Partager

Obtenir des données Actualiser tout Requête et connexions Propriétés Workbook Links Récupérer et transformer des données Types de données Trier et filtrer

CASTELLION

Commune	Code_d eparte ment	Surface	Type_local	Occupation	Type_contrat	Formule	Valeur_declaree_b lens	Prix_co tisation mensuel	NBCAR D	NBCAR E	NBCAR F	NBCAR G	NBCAR H	NBCAR I	NBCAR J
CEAUX	1	87	Appartement	Propriétaire	Residence principale	Classique	0-25000	16	3	13	5	4	9	1	
VAX	1	138	Appartement	Propriétaire	Residence secondaire	Classique	0-25000	11	3	9	5	4	7	1	
VAX	1	68	Appartement	Propriétaire	Residence principale	Integral	0-25000	13	3	6	5	4	7	1	
NAT	1	71	Appartement	Propriétaire	Residence principale	Classique	0-25000	9	3	16	5	4	9	1	
VAX	1	88	Appartement	Propriétaire	Residence principale	Classique	0-25000	13	3	10	5	4	7	1	
VAX	1	64	Appartement	Locataire	Residence principale	Integral	0-25000	12	4	17	5	4	7	1	
T	1	48	Appartement	Propriétaire	Residence principale	Integral	0-25000	9	3	8	5	4	6	1	
NAT	1	47	Appartement	Propriétaire	Residence secondaire	Integral	0-25000	9	3	14	5	4	9	1	
D	1	64	Appartement	Propriétaire	Residence principale	Integral	0-25000	9	3	18	5	4	6	1	
LUEL	1	42	Appartement	Propriétaire	Residence principale	Classique	0-25000	11	3	5	5	4	8	1	
LUX	1	38	Appartement	Propriétaire	Residence principale	Integral	0-25000	10	3	9	5	4	7	1	
JA	1	108	Appartement	Locataire	Residence principale	Classique	0-25000	13	3	17	5	4	6	1	
ES LA FORET	77	135	Maison	Propriétaire	Residence principale	Classique	25000-50000	22	3	21	5	5	16	2	
EU	1	53	Appartement	Locataire	Residence principale	Integral	0-25000	12	2	17	5	4	7	1	
EU	1	70	Appartement	Locataire	Residence principale	Classique	0-25000	14	2	21	5	4	7	1	
EU	1	94	Maison	Propriétaire	Residence principale	Integral	0-25000	13	3	8	5	4	7	1	
EU	1	38	Appartement	Propriétaire	Residence principale	Classique	0-25000	9	2	11	5	4	7	1	
VULBAS	1	43	Appartement	Propriétaire	Residence principale	Integral	0-25000	14	3	17	5	4	12	1	
VULBAS	1	32	Appartement	Propriétaire	Residence principale	Classique	0-25000	11	3	9	5	4	12	1	
EU	1	71	Appartement	Propriétaire	Residence principale	Classique	0-25000	12	3	7	5	4	7	1	

Contrat+(2)+(1) Region+(2)+(1) Region\_corrige

J'ai importé les deux fichiers CSV dans un seul fichier Excel avec un onglet par fichier CSV. Cela m'a permis de faire mon analyse grâce à des filtres, des fonctions et des mises en forme conditionnelles.

Une fois le dictionnaire de données défini et les données corrigées, j'ai exporté chaque feuille Excel dans un fichier CSV. J'ai ensuite ouvert chacun des fichiers CSV avec l'outil Notepad++ afin de les convertir en UTF8. Les fichiers étaient ainsi prêts à être chargés dans la base de données.

## Etape 01 : dictionnaire de données 2/3

### Méthode d'analyse des données

Il est établi que chacun des fichiers CSV sera une table. Il est donc nécessaire d'utiliser les entêtes de fichiers CVS pour déterminer la liste des colonnes qu'il faudra créer dans chacune des tables.

Afin de garantir le minimum de maintenance sur la base de données, j'ai appliqué [une marge de sécurité](#) sur l'ensemble des longueurs d'attributs., je n'ai donc pas attribué strictement la longueur maximale de chaque attribut.

Pour déterminer la longueur maximum de chacun des attributs, j'ai utilisé la fonction Excel [NBCAR](#) , et j'ai ensuite trié par ordre décroissant.

Pour chacun des attributs j'ai utilisé les filtres d'Excel pour vérifier s'il y avait des [valeurs vides](#) dans la liste de donnée.

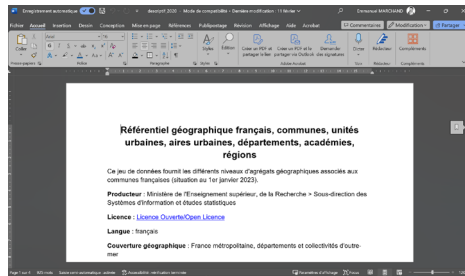
Pour déterminer la typologie de données, pour chaque attribut numérique, j'ai fait un filtre en recherchant des lettres, si la recherche ne retournait rien alors j'ai considéré le champ comme [entier](#). Il n'y avait aucun attribut avec des virgules, donc les type FLOAT ou DECIMAL, n'ont pas été utilisés.

### Méthode de correction des données

J'ai utilisé les mises en forme conditionnelle pour détecter les doublons de données par attributs et ainsi déterminé la nécessité de clé étrangère. Dans la table [Contrat](#), j'ai changé le nom de l'attribut '[Code\\_dep\\_code\\_commune](#)' par '[Lien\\_contrat\\_region](#)' pour mettre en évidence la clé étrangère en relation 0 à PLUSIEURS entre Région et Contrat.

J'ai corrigé les valeurs de l'attribut '[Code\\_dep\\_code\\_commune](#)' de la table Région, car tous les départements commençant par zéro n'avaient pas de zéro en première position de cet attribut. J'ai utilisé la fonction Excel [CONCATENER](#) pour l'ajouter. Sans cette modification, le lien entre les tables Contrat et Région aurait été perdu pour tous les départements commençant par un zéro.

# Etape 01 : dictionnaire de données 3/3



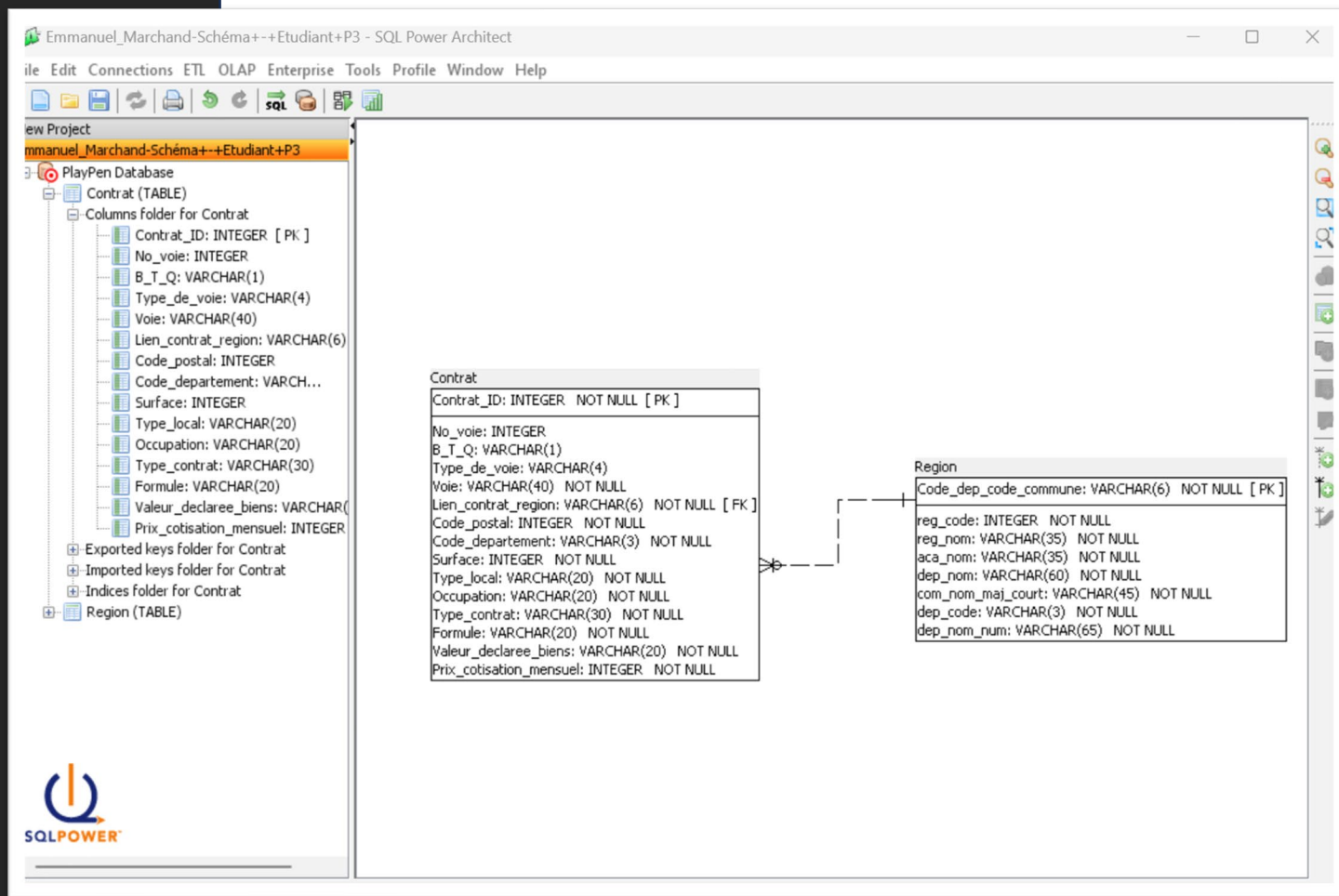
J'ai consulté le site de [data.gouv.fr](https://data.gouv.fr) pour trouver la documentation du fichier d'interface que j'ai téléchargé. Cela m'a permis d'identifier le contenu de chacun des attributs du fichier Région

	Nom des colonnes	Type de donnée	Taille	Clé	Contrainte	Description
CONTRAT.CSV	Contrat_ID	INT		Clé primaire	NOT NULL	Id unique pour les contrats
	No_voie	INT				Numéro dans la voie pour l'adresse du logement assuré
	B_T_Q	CHAR	1			Indicateur éventuel de répétition pour l'adresse du logement assuré sur un caractère
	Type_de_voie	VARCHAR	4			Type de voie pour l'adresse du logement assuré: rue, av (Avenue), rte (Route), ...
	Voie	VARCHAR	40		NOT NULL	Libellé de la voie pour l'adresse du logement assuré
	Lien_contrat_region	VARCHAR	6	Clé secondaire	NOT NULL	Lien pour la relation PLUSIEURS entre CONTRAT et REGION
	Code_postal	INT			NOT NULL	Code postal pour l'adresse du logement assuré
	Commune	VARCHAR	45		NOT NULL	Libellé de la commune de l'adresse du logement
	Code_departement	INT			NOT NULL	Code département pour l'adresse du logement assuré
	Surface	INT			NOT NULL	Surface du logement assuré
	Type_local	VARCHAR	20		NOT NULL	Type de logement : Appartement ou Maison
	Occupation	VARCHAR	20		NOT NULL	Le statut de l'occupant : Propriétaire ou Locataire
	Type_contrat	VARCHAR	30		NOT NULL	Le type de contrat d'assurance : Mise en location, Résidence principale, résidence secondaire
REGION.CSV	Formule	VARCHAR	20		NOT NULL	La formule souscrite par le client : Classique, Intégral.
	Valeur_declaree_biens	VARCHAR	20		NOT NULL	Une fourchette de la valeur des biens déclarés dans le logement.
	Prix_cotisation_mensuel	INT			NOT NULL	Le prix mensuel de la cotisation client.
	Code_dep_code_commune	VARCHAR	6	Clé primaire	NOT NULL	Concaténation du code département et code commune pour avoir une clé unique
	reg_code	INT			NOT NULL	Code région
	reg_nom	VARCHAR	35		NOT NULL	Libellé région
	aca_nom	VARCHAR	35		NOT NULL	Libellé de l'académie
	dep_nom	VARCHAR	60		NOT NULL	Libellé du département
	com_nom_maj_court	VARCHAR	45		NOT NULL	Libellé court de la commune en majuscule
	dep_code	VARCHAR	3		NOT NULL	Code département
	deo nom num	VARCHAR	65		NOT NULL	Code et libellé du département

**CONCLUSION** : L'observation du contenu de chacun des attributs présents dans les fichiers CSV, m'a permis de :

- Déterminer les longueurs de champs
- Déterminer si l'attribut accepte les valeurs NULL
- Déterminer les typologies de données
- Corriger les erreurs de données.

## Etape 02 : créer le schéma relationnel normalisé en 3NF



1 J'ai suivi les recommandations OpenClassroom de l'étape 02 à savoir :

- J'ai téléchargé et installé le logiciel [SQL Power Architect](#). Pour me familiariser avec celui-ci j'ai regardé les liens fournis pointant vers des vidéos tutoriels.
- J'ai suivi la partie du cours pour la normalisation 3NF.

2 Le respect de la normalisation 3NF implique :

- Que chaque table soit constituée d'une **clé primaire**
- La suppression des **redondances** de données. Pour cela j'ai supprimé l'attribut '**Commune**' de la table **Contrat**, car cette information se trouve déjà dans la table de référence **Région**, dans l'attribut '**com\_nom\_maj\_court**'.



## Etape 02 : point d'attention sur la forme normale 3NF



### ATTENTION

J'ai conservé les colonnes 'Code département' et 'dep\_code' respectivement dans les tables **Contrat** et **Région** alors qu'elles étaient redondantes. Pourquoi ? : il y a des incohérences de données dans ces tables et le nettoyage des données ne fait pas l'objet de cet exercice. En effet dans la table **Contrat**, il y a des attributs 'Code\_département' qui sont erronés, par exemple la ville de CHATENAY MALABRY est notée dans le 75 en lui attachant le Code\_dep\_code\_commune ' égal à 75019 et alors qu'elle devrait être dans le 92 avec le code\_dep\_code\_commune égal à 92019. Cela ne permet donc pas de faire des INNER JOIN fiables entre les deux tables.

2133																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Il faut également noter que dans la table 'Region' on retrouve toujours cette même ville présente dans deux départements différents.



Code_dep_code_commune	reg_co	reg_nom	aca_no	dep_nom	com_nom_maj_court	dep_code	dep_nom_num
5019	11	Ile-de-France	Paris	Paris	CHATENAY MALABRY	75	Paris (75)
2019	11	Ile-de-France	Versaille	Hauts-de-Seine	CHATENAY MALABRY	92	Hauts-de-Seine (92)

## Etape 03 : créer et charger la base de données 1/4

```
MySQL 8.0 Command Line Cli x + v
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 66
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| oclassroom |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> |
```

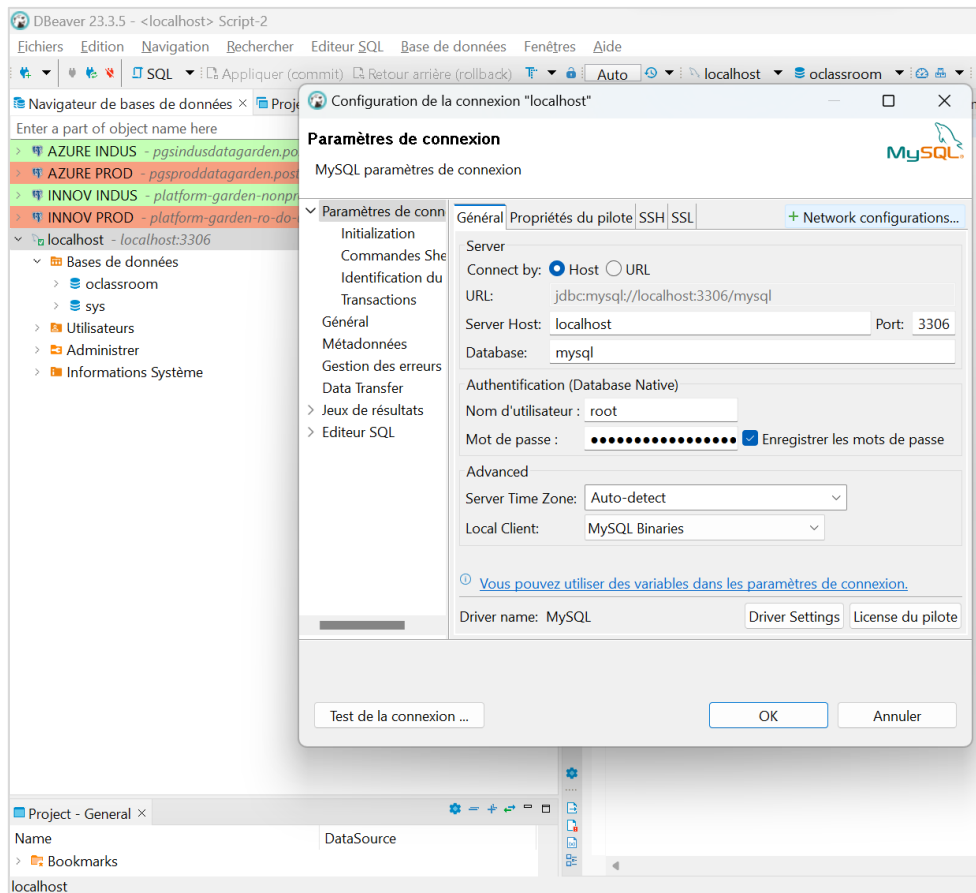
J'ai suivi les recommandations OpenClassroom de l'étape 03 à savoir :

- J'ai choisi le SGBD [MySQL](#) parmi ceux proposés. C'est celui avec lequel j'ai déjà travaillé.
- J'ai installé MySQL sur mon PC pour créer une base en local.
- Dans MySQL avec la commande [CREATE DATABASE \[IF NOT EXISTS\] oclassroom;](#) , j'ai créé une base dédiée aux projets OpenClassroom pour ne pas travailler directement sur la base Système SYS

## Etape 03 : créer et charger la base de données 2/4

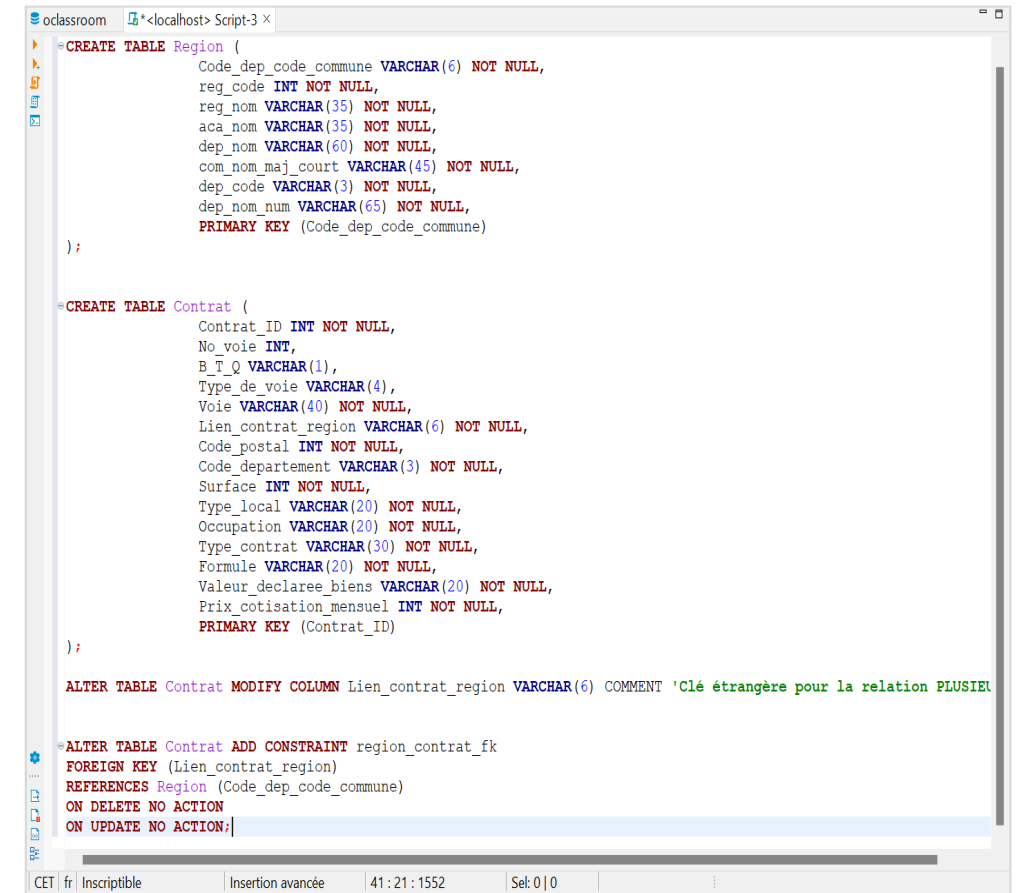
1

J'ai créé une connexion de base depuis le logiciel **Dbeaver** qui est le logiciel de gestion de base de données que j'utilise actuellement dans le cadre de mon travail.



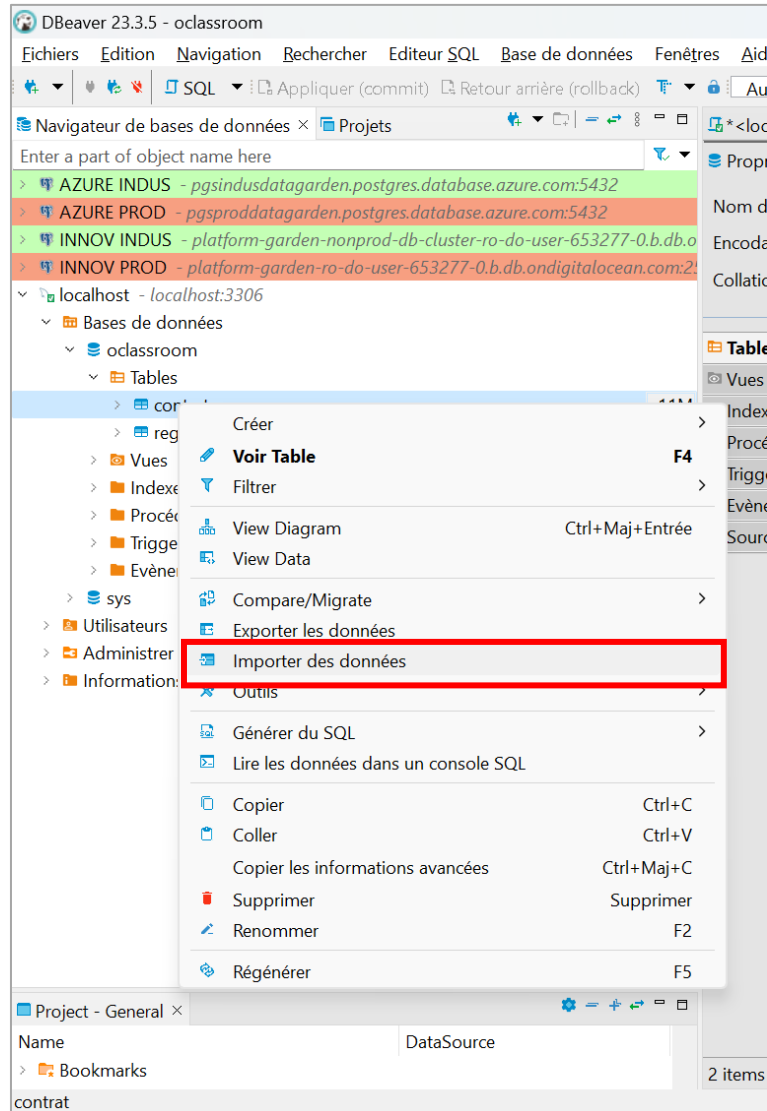
2

J'ai exécuté le script SQL de création de base issue du logiciel de modélisation de base SQL Power Architecte.

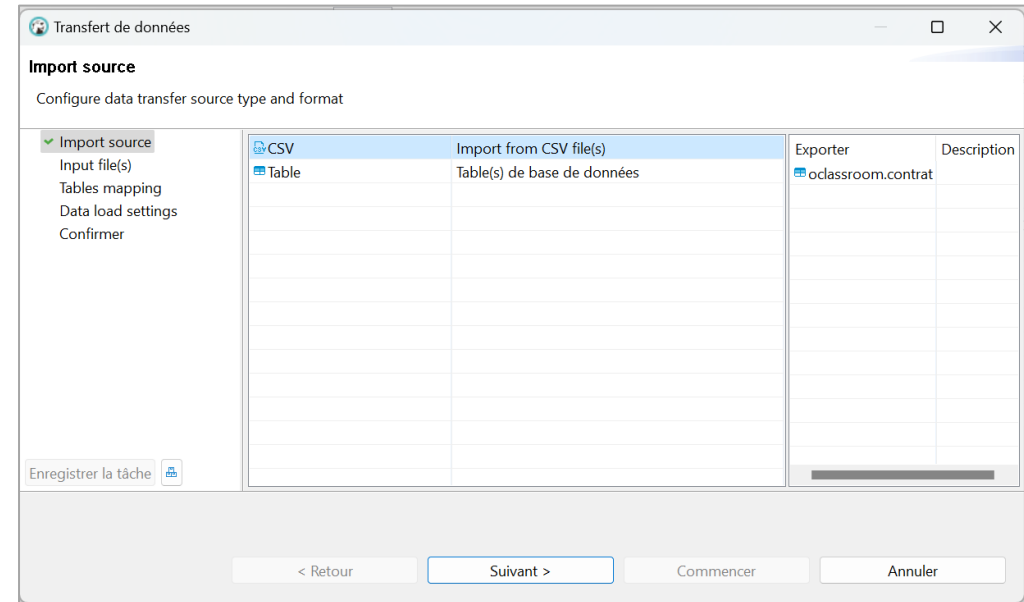




## Etape 03 : créer et charger la base de données 3/4



J'ai ensuite utilisé l'utilitaire d'importation de données de [Dbeaver](#) pour charger chacun des fichiers CSV, créés à l'étape 01, dans chacune des tables.



## Etape 03 : créer et charger la base de données 4/4

Une fois les données chargées, on obtient bien le nombre de lignes attendues dans chacune des tables.

**contrat**

Propriétés Données ER Diagram localhost Bases de données oclassroom Tables contrat

contrat Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	Contrat_ID	No_voie	B_T_Q	Type_de_voie	Voie	Lien_contrat_region	Cc
1	100601	190	A	RUE	CENTRALE	01350	
2	100602	347		RUE	DU CHATEAU	01103	
3	100603	58		AV	DU MONT BLANC	01143	
4	100604	140		RUE	DE L'ABBE JOLIVET	01288	
5	100605	39		RUE	BUFFON	01033	
6	100606	8		RUE	DE GENEVE	01354	
7	100607	2		RUE	DU RECULET	01354	
8	100608	1403		RUE	JEAN DE GINGINS	01143	
9	100609	226		ALL	DES CAPUCINES	01354	
10	100610	276		RTE	DE POUYNY	01288	
11	100611	79		CRS	DE VERDUN	01283	
12	100612	240		RUE	DE PRE BAILLY	01173	
13	100613	3		RUE	TURENNE	01033	
14	100614	44		ALL	DU SQUARE DE LAUSANNE	01143	
15	100615	59		RUE	ALEXANDRE BERARD	01004	
16	100616	282		CHE	DES LONGES RAYES	01071	
17	100617	54		GR	GRANDE RUE	01396	
18	100618	1		RUE	DE GEX	01160	
19	100619	17		LOT	LES JARDINS DE CHEVRY	01103	
20	100620	329		RUE	DES CARPIERES	01399	
21	100621	369		AV	CHARLES DE GAULLE	01202	
22	100622	6		RUE	DES USINIERS	01173	
23	100623	2		RUE	DU JURA	01354	
24	100624	54		RUE	MARCEL DEMIA	01004	
25	100625	269		CHE	DE CHARIGNIN	01034	
26	100626	322		RUE	DE BRETAGNY	01281	

Régénérer Save Cancel Exporter les résultats ... 200 30335

200 row(s) fetched - 0,001s, on 2024-03-02 at 09:36:07

**region**

Propriétés Données ER Diagram localhost Bases de données oclassroom Tables region

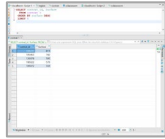
region Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	Code_dep_code_commune	reg_code	reg_nom	aca_nom	dep_nom	com_nom_maj_cc
1	01001	84	Auvergne-Rhône-Alpes	Lyon	Ain	L ABERGEMENT CLE
2	01002	84	Auvergne-Rhône-Alpes	Lyon	Ain	L ABERGEMENT DE
3	01003	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMAREINS
4	01004	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBERIEU EN BUGE
5	01005	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBERIEUX EN DOI
6	01006	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBLEON
7	01007	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBRONAY
8	01008	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBUTRIX
9	01009	84	Auvergne-Rhône-Alpes	Lyon	Ain	ANDERT ET CONDO
10	01010	84	Auvergne-Rhône-Alpes	Lyon	Ain	ANGLEFORT
11	01011	84	Auvergne-Rhône-Alpes	Lyon	Ain	APREMONT
12	01012	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARANC
13	01013	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARANDAS
14	01014	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBENT
15	01015	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBIGNIEU
16	01016	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBIGNY
17	01017	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARGIS
18	01018	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARLOD
19	01019	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARMIX
20	01020	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARNANS
21	01021	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARS SUR FORMANS
22	01022	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARTEMARE
23	01023	84	Auvergne-Rhône-Alpes	Lyon	Ain	ASNIERES SUR SAO
24	01024	84	Auvergne-Rhône-Alpes	Lyon	Ain	ATTIGNAT
25	01025	84	Auvergne-Rhône-Alpes	Lyon	Ain	BAGE LA VILLE
26	01026	84	Auvergne-Rhône-Alpes	Lyon	Ain	BAGE LE CHATEL

Régénérer Save Cancel Exporter les résultats ... 200 38916

200 row(s) fetched - 0,002s, on 2024-03-02 at 09:33:54

## Etape 04 et 05 : créer les requêtes SQL



Une fois ma Base créée, et mes tables chargées , j'ai exécuté chacune des requêtes dans l'éditeur SQL de [Dbeaver](#). Le détail se trouve en annexe.

01

```
select contrat_ID, Surface from contrat c
where Commune = 'Caen'
```

02

```
select contrat_ID, Type_contrat, Formule from contrat
c
where Type_local = 'Maison'
and Code_departement = 71
order by Formule
```

03

```
select distinct reg_nom
from region r
```

04

```
select count(*) from contrat c
where Type_contrat = 'Residence principale'
```

05

```
select AVG(Surface) from contrat c
where Code_departement = 75
```

06

```
SELECT contrat_id, Surface
FROM contrat c
ORDER BY surface DESC
LIMIT 5
```

07

```
select AVG(Prix_cotisation_mensuel) from contrat c
```

08

```
select Valeur_declaree_biens , count(*) as nb_contrats
from contrat c
group by Valeur_declaree_biens
order by 2 desc
```

09

```
select Code_departement , AVG(Prix_cotisation_mensuel)
from contrat c
group by Code_departement
order by AVG(Prix_cotisation_mensuel) desc
limit 10
```

10

```
select count(*) as nb_contrats
from contrat c , region r
where Lien_contrat_region = r.Code_dep_code_commune
and c.Formule = 'Integral'
and r.reg_nom = 'Pays de la Loire'
```

11

```
select r.com_nom_maj_court
from contrat c , region r
where Lien_contrat_region = r.Code_dep_code_commune
group by 1
having count(*) >= 150
```

12

```
select r.reg_nom , count(c.Contrat_ID) as
Nb_contrat_total
from contrat c , region r
where Lien_contrat_region = r.Code_dep_code_commune
group by 1
Order by 2 desc
```





# ANNEXES

**Requête 01** : Lister les numéros de contrats (contrat\_ID) avec leur surface pour la commune de Caen.

```
select contrat_ID, Surface from contrat c
where Commune = 'Caen'
```

	contrat_ID	Surface
1	103 791	35
2	103 792	99
3	103 793	40
4	103 794	20

**Requête 02 :** Lister les numéros de contrats avec le type de contrat et leur formule pour les maisons du département 71.

```
select contrat_ID, Type_contrat, Formule from contrat c
where Type_local = 'Maison'
and Code_departement = 71
order by Formule
```

	contrat_ID	Type_contrat	Formule
1	114779	Residence principale	Classique
2	114782	Residence principale	Classique
3	114768	Residence principale	Integral
4	114812	Residence principale	Integral



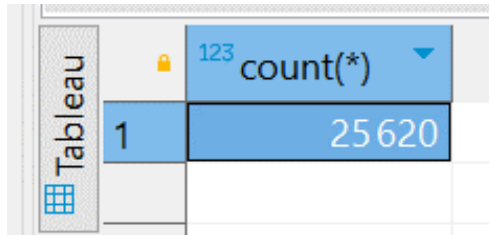
## Requête 03 : Lister le nom des régions de France.

```
select distinct reg_nom  
from region r
```

record	Tableau	Texte	ABC reg_nom
	1		Auvergne-Rhône-Alpes
	2		Hauts-de-France
	3		Provence-Alpes-Côte d'Azur
	4		Grand Est
	5		Occitanie
	6		Normandie
	7		Nouvelle-Aquitaine
	8		Centre-Val de Loire
	9		Bourgogne-Franche-Comté
	10		Bretagne
	11		Corse
	12		Pays de la Loire
	13		Ile-de-France
	14		Guadeloupe
	15		Martinique
	16		Guyane
	17		La Réunion
	18		Collectivités d'outre-mer
	19		Mayotte

## Requête 04 : Combien existe-t-il de contrats sur les résidences principales ?

```
select count(*) from contrat c  
where Type_contrat = 'Residence principale'
```



The image shows a Tableau interface. On the left, a vertical sidebar labeled 'Tableau' contains a grid icon. The main area displays a table with two columns. The first column has a single row with the value '1'. The second column has a header row with 'count(\*)' and a dropdown arrow, and a data row with the value '25 620'. A small yellow lock icon is visible above the header row.

	count(*)
1	25 620

## Requête 05 : Quelle est la surface moyenne des logements avec un contrat à Paris ?

```
select AVG(Surface) from contrat c  
where Code_departement = 75
```

select AVG(Surface) from contrat c

123	AVG(Surface) ▼
	51,7695



## Requête 06 : Quels sont les 5 contrats qui ont les surfaces les plus élevées ?

```

SELECT contrat_id, Surface
  FROM contrat c
 ORDER BY surface DESC
LIMIT 5

```

		<sup>123</sup> contrat_id ▼	<sup>123</sup> Surface ▼
<div>Tableau</div> <div>Texte</div>	1	104211	815
	2	105463	742
	3	130878	595
	4	100822	570
	5	109872	559

## Requête 07 : Quel est le prix moyen de la cotisation mensuelle ?

```
select AVG(Prix_cotisation_mensuel) from contrat c
```

Tableau		123 AVG(Prix_cotisation_mensuel) ▼
	1	19,3287

**Requête 08 :** Quel est le nombre de contrats pour chaque catégorie de prix de la valeur déclarée des biens ?

```
select Valeur_declaree_biens , count(*) as nb_contrats
from contrat c
group by 1
order by 2 desc
```

		ABC Valeur_declaree_biens ▼	123 nb_contrats ▼
Tableau	1	0-25000	22720
	2	25000-50000	6815
	3	50000-100000	696
Texte	4	100000+	104

**Requête 09 :** Quel est le classement des 10 départements où le prix moyen de la cotisation est le plus élevé ?

```
select Code_departement , AVG(Prix_cotisation_mensuel)
from contrat c
group by Code_departement
order by AVG(Prix_cotisation_mensuel) desc
limit 10
```

	123 Code_departement	123 AVG(Prix_cotisation_mensuel)
1	75	36,3991
2	92	26,2694
3	94	19,8242
4	78	18,8802
5	69	18,4568
6	1	18,2389
7	6	18,1376
8	17	17,3173
9	74	17,1584
10	20	17,0283



**Requête 10 :** Quel est le nombre de contrats avec des formules "Intégral" pour la région Pays de la Loire ?

```
select count(*) as nb_contrats
from contrat c , region r
where Lien_contrat_region = r.Code_dep_code_commune
and c.Formule = 'Integral'
and r.reg_nom = 'Pays de la Loire'
```

Tableau	1	nb_contrats
		561

## Requête 11 : Quelle est la liste des communes ayant au moins 150 contrats ?

```

select r.com_nom_maj_court, count(*) as nb_contrat
from contrat c, region r
where Lien_contrat_region= r.Code_dep_code_commune
group by 1
having count(*) >= 150
order by 2 desc

```

	com_nom_maj_court	nb_contrat
1	PARIS 18	515
2	PARIS 17	468
3	PARIS 15	407
4	PARIS 16	394
5	NICE	387
6	PARIS 11	381
7	BORDEAUX	302
8	PARIS 20	302
9	NANTES	291
10	PARIS 19	266
11	PARIS 10	263
12	PARIS 12	252
13	PARIS 14	222
14	GRENOBLE	220
15	PARIS 9	204
16	TOULOUSE	187
17	TOULON	170
18	COURBEVOIE	163
19	LILLE	161
20	PARIS 3	159

## Requête 12 : Quel est le nombre de contrats pour chaque région ?

```

select r.reg_nom , count(c.Contrat_ID) as Nb_contrat_total
from contrat c , region r
where Lien_contrat_region = r.Code_dep_code_commune
group by 1
Order by 2 desc
  
```

	ABC reg_nom	123 Nb_contrat_total
1	Ile-de-France	13474
2	Provence-Alpes-Côte d'Azur	3287
3	Auvergne-Rhône-Alpes	2972
4	Nouvelle-Aquitaine	2097
5	Occitanie	1837
6	Hauts-de-France	1327
7	Pays de la Loire	1138
8	Bretagne	945
9	Normandie	898
10	Grand Est	806
11	Centre-Val de Loire	804
12	Bourgogne-Franche-Comté	402
13	Corse	247
14	Martinique	60
15	Guyane	37
16	La Réunion	4