



RESUME

Dans le but de réaliser le livrable du projet 3 vous trouverez dans ce documents technique l'exploration des données, leurs dictionnaires ainsi que le schéma relationnel et la création de la base de données avec les requêtes SQL.

Yann RES

Projet 3

DOCUMENT TECHNIQUE

Requêtez une base de données avec SQL

Table des matières

Aucune entrée de table des matières n'a été trouvée.

1. Contexte.

L'objectif de ce projet est de construire une base de données fiable et exploitable de deux CSV contrat et région en passant par plusieurs étapes et en utilisant différents outils. Tout d'abord l'exploration des fichiers CSV, puis complétion du dictionnaire des données et du schéma relationnel. Puis création de la base de données et importation des données depuis les CSV. Jusqu'à obtenir une base de données exploitable via des requêtes SQL.

Pour ce qui est des outils j'ai utilisé Excel avec PowerQuery, SQL Power Architect puis MySQL WorkBench. Je me suis aussi aidé de NotePad++ pour un problème d'encodage UTF-8 avec BOM.

But final avoir les deux tables contrat et région dans la base de données avec respectivement 30 335 et 38 916 lignes.

2. Découverte des données brutes.

Ouverture des CSV via import de données de CSV dans un seul et même fichier Excel. Ce qui nous permet de découvrir le type de données et explorer leurs contraintes via des filtres mais aussi de nettoyer les données pour un import propre dans la BDD.

Ci-dessous un exemple des données à corriger dans « voie ». ou la présence de cellule vide pour les null.

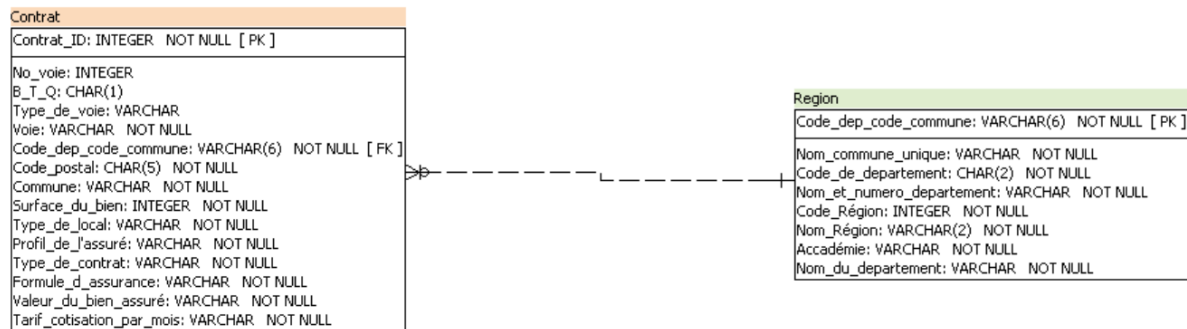
Contrat ID	No. voie	B T Q	Type de voie	Voie	Code dep. code commune	Code postal	Surface	Type local	Occupation	Type contrat	Formule	Valeur declaree biens	Prix cotisation mensuel
100601	190	A	RUE		01350	01370	50	Appartement	Locataire	Residence principale	Integral	0-25000	25
100602	347		RUE		01103	01170	48	Appartement	Locataire	Residence principale	Classique	0-25000	30
100603	58	AV			01143	01220	131	Appartement	Proprietaire	Residence principale	Integral	25000-50000	57
100604	140		RUE		01288	01630	109	Maison	Locataire	Residence principale	Integral	25000-50000	43
100605	39		RUE		01033	01200	109	Appartement	Locataire	Residence principale	Classique	0-25000	33
100606	8		RUE		01354	01630	53	Appartement	Proprietaire	Residence principale	Classique	0-25000	19
100607	2		RUE		01354	01630	59	Appartement	Proprietaire	Residence principale	Integral	0-25000	15
100608	1403		RUE		01143	01220	93	Maison	Proprietaire	Mise en location	Integral	25000-50000	34
100609	226	ALL			01354	01630	117	Maison	Proprietaire	Residence principale	Classique	25000-50000	32
100610	276	RTE			01288	01630	36	Appartement	Proprietaire	Residence principale	Integral	25000-50000	22
100611	79	CRS			01283	01100	138	Appartement	Proprietaire	Residence secondaire	Classique	0-25000	11
100612	240		RUE		01173	01170	45	Appartement	Locataire	Residence principale	Classique	0-25000	16
100613	3		RUE		01033	01200	83	Appartement	Locataire	Residence principale	Classique	0-25000	14
100614	44	ALL			01143	01220	88	Appartement	Locataire	Residence principale	Integral	25000-50000	34
100615	59		RUE		01004	01500	165	Appartement	Locataire	Residence principale	Classique	25000-50000	24
100616	282	CHE			01071	01170	42	Appartement	Proprietaire	Residence principale	Classique	0-25000	17
100617	54	GR			01396	01150	68	Appartement	Proprietaire	Residence secondaire	Classique	0-25000	10
100618	1		RUE		01160	01210	83	Appartement	Proprietaire	Residence principale	Classique	0-25000	20
100619	17	LOT			01103	01170	30	Appartement	Proprietaire	Residence secondaire	Classique	0-25000	12
100620	329		RUE		01399	01170	25	Appartement	Locataire	Residence principale	Integral	0-25000	11
100621	369	AV			01202	01150	53	Appartement	Locataire	Residence principale	Integral	0-25000	12
100622	6		RUE		01173	01170	46	Appartement	Proprietaire	Residence secondaire	Classique	0-25000	12
100623	2		RUE		01354	01630	84	Appartement	Proprietaire	Residence principale	Classique	0-25000	20
100624	54		RUE		01004	01500	68	Appartement	Locataire	Residence principale	Integral	0-25000	13
100625	269	CHE			01034	01300	62	Appartement	Proprietaire	Residence principale	Classique	0-25000	11
100626	322		RUE		01281	01210	113	Maison	Proprietaire	Residence principale	Integral	25000-50000	32
100627	132	ALL			01354	01630	132	Appartement	Proprietaire	Residence principale	Classique	25000-50000	30
100628	80		RUE		01160	01210	64	Appartement	Proprietaire	Residence principale	Integral	0-25000	15
100629	4		RUE		01138	01350	36	Appartement	Proprietaire	Residence principale	Integral	0-25000	12
100630	79	GR			01202	01150	70	Appartement	Locataire	Residence principale	Classique	0-25000	14

3. Le dictionnaire des données.

Quelques règles sur certains des champs comme les codes postaux qui doivent être à 5 caractères strict d'où le CHAR 5 ou B_T_Q qui est un caractère unique CHAR 5. Pour ce qui est de Code_dep_code_commune en VARCHAR 6 qui est entre 5 et 6 caractères (6 pour les DOM TOM).

Pour reg_code et dep_code en CHAR 2 et 3 pour ne pas perdre le 0 pour les code allant de 1 à 9 avec INT.

4. Le schémas relationnel et export DDL.



Ici rien de compliquer on reprend le dictionnaire des données et on ajoute les contrainte null et not null que l'on a découvert dans l'exploration des csv. Après l'export du DDL pour MySQL il restait quelques petits détails à corriger comme l'obligation d'avoir des tailles définies pour VARCHAR. Ce qui nous donne le code qui suit.

```

CREATE TABLE Region (
  Code_dep_code_commune VARCHAR(6) NOT NULL
    CHECK (CHAR_LENGTH(TRIM(Code_dep_code_commune)) BETWEEN 5 AND
6),
  com_nom_maj_court VARCHAR(60) NOT NULL,
  dep_code VARCHAR(3) NOT NULL,
    CHECK (CHAR_LENGTH(TRIM(dep_code)) BETWEEN 2 AND 3),
  dep_nom_num VARCHAR(50) NOT NULL,
  reg_code INT NOT NULL,
  reg_nom VARCHAR(40) NOT NULL,
  aca_nom VARCHAR(50) NOT NULL,
  dep_nom VARCHAR(50) NOT NULL,
  PRIMARY KEY (Code_dep_code_commune)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

CREATE TABLE Contrat (
  Contrat_ID INT NOT NULL AUTO_INCREMENT,
  No_voie INT,
  B_T_Q CHAR(1) NULL,
  Type_de_voie VARCHAR(50) NOT NULL,
  Voie VARCHAR(255) NOT NULL,
  Code_dep_code_commune VARCHAR(6) NOT NULL
    CHECK (CHAR_LENGTH(TRIM(Code_dep_code_commune)) BETWEEN 5 AND
6),
  Code_postal CHAR(5) NOT NULL,
  Commune VARCHAR(100) NOT NULL,
  Surface INT UNSIGNED NOT NULL,
  Type_local VARCHAR(50) NOT NULL,
  Occupation VARCHAR(50) NOT NULL,
  Type_contrat VARCHAR(50) NOT NULL,
  Formule VARCHAR(50) NOT NULL,

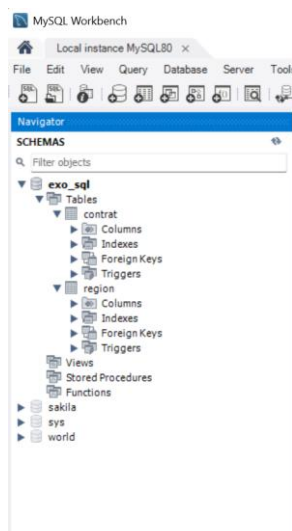
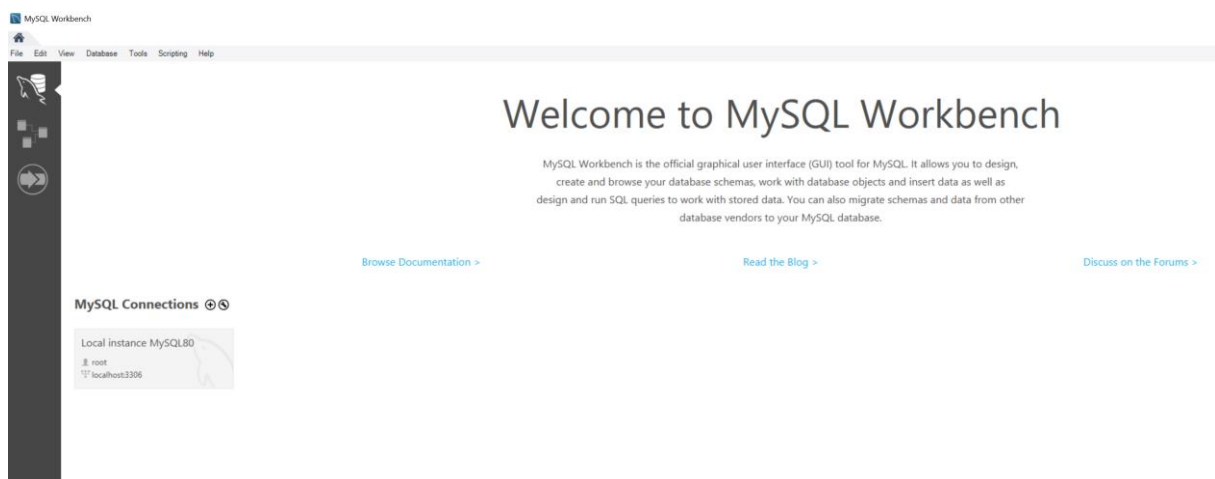
```

```
Valeur_declaree_biens    VARCHAR(30) NOT NULL,
Prix_cotisation_mensuel  INT UNSIGNED NOT NULL,
PRIMARY KEY (Contrat_ID),
CONSTRAINT region_contrat_fk
    FOREIGN KEY (Code_dep_code_commune)
    REFERENCES Region (Code_dep_code_commune)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

CREATE INDEX ix_contrat_code_dep_comm ON Contrat
(Code_dep_code_commune);
```

5. Création de la base de données.

Après installation de MySQL Workbench et configuration de la base de données Workbench on crée les tables et les contraintes grâce au code généré précédemment avec Architect.



6. Import des données dans la base.

Deux soit s'offre à nous l'import via l'outils dédié ou via un script SQL. J'ai rencontré des problèmes de lenteur anormale et de la casse sur l'encodage avec l'outils (ci-dessous) alors je suis passé via un script SQL, code que vous retrouverez plus bas.

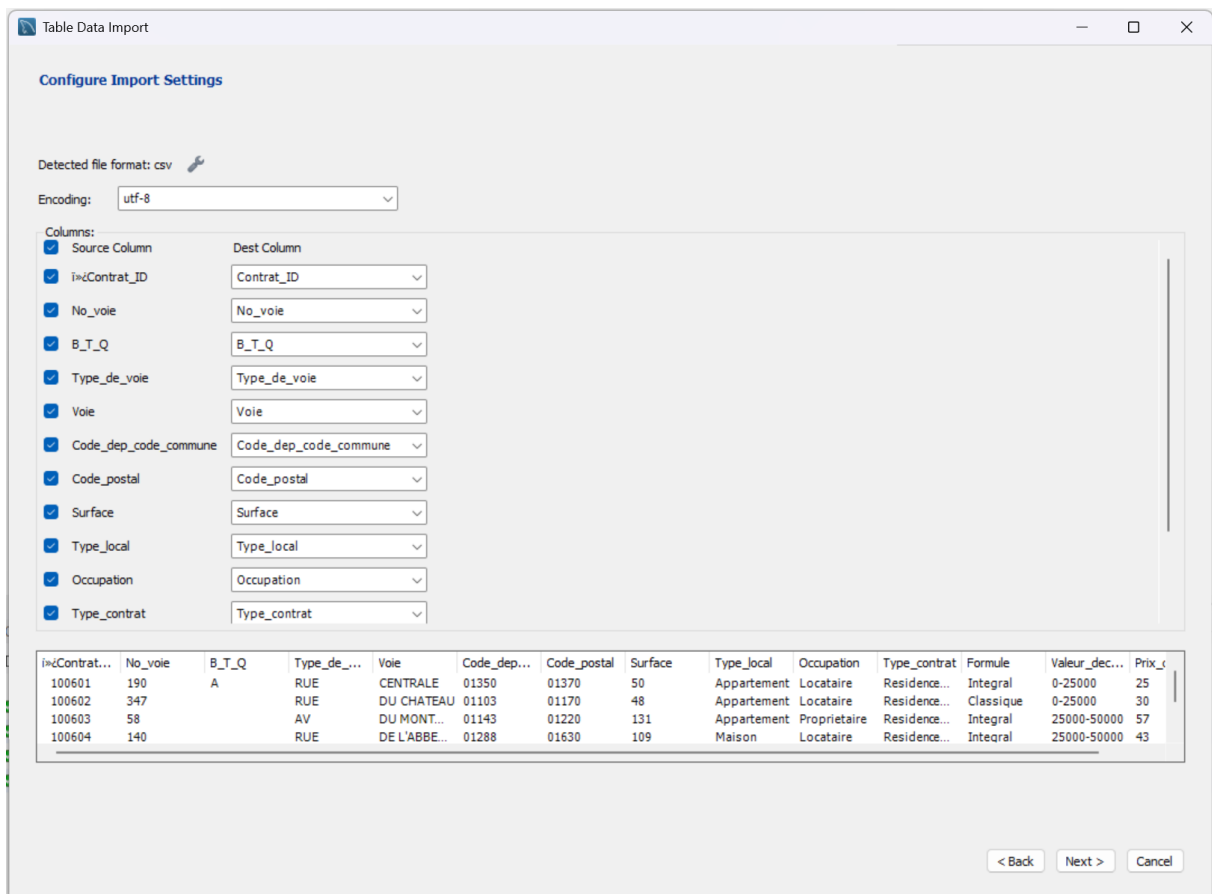


Table Data Import

Configure Import Settings

Detected file format: csv

Encoding: utf-8

Columns:

Source Column	Dest Column
☒ InContrat_ID	Contrat_ID
☒ No_voie	No_voie
☒ B_T_Q	B_T_Q
☒ Type_de_voie	Type_de_voie
☒ Voie	Voie
☒ Code_dep_code_commune	Code_dep_code_commune
☒ Code_postal	Code_postal
☒ Surface	Surface
☒ Type_local	Type_local
☒ Occupation	Occupation
☒ Type_contrat	Type_contrat

InContrat...	No_voie	B_T_Q	Type_de...	Voie	Code_dep...	Code_postal	Surface	Type_local	Occupation	Type_contrat	Formule	Valeur_dec...	Prix_c
100601	190	A	RUE	CENTRALE	01350	01370	50	Appartement	Locataire	Residence...	Integral	0-25000	25
100602	347		RUE	DU CHATEAU	01103	01170	48	Appartement	Locataire	Residence...	Classique	0-25000	30
100603	58		AV	DU MONT...	01143	01220	131	Appartement	Proprietaire	Residence...	Integral	25000-50000	57
100604	140		RUE	DE L'ABBE...	01288	01630	109	Maison	Locataire	Residence...	Integral	25000-50000	43

< Back Next > Cancel

Import dans région :

```
USE exo_sql;

LOAD DATA LOCAL INFILE
'C:/Users/Yann/Documents/Projet3BI/region_v2_bomless.csv'
INTO TABLE Region
CHARACTER SET utf8mb4
FIELDS TERMINATED BY ';'
ENCLOSED BY '"'
ESCAPED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES
(@code_comm, @reg_code, @reg_nom, @aca_nom, @dep_nom,
@com_nom_maj_court, @dep_code, @dep_nom_num)
SET
  Code_dep_code_commune = TRIM(@code_comm),
```

```

com_nom_maj_court      = TRIM(@com_nom_maj_court),
dep_code               = TRIM(@dep_code),
dep_nom_num            = TRIM(@dep_nom_num),
reg_code               = NULLIF(TRIM(@reg_code), ''),
reg_nom                = TRIM(@reg_nom),
aca_nom                = TRIM(@aca_nom),
dep_nom                = TRIM(@dep_nom);

```

Import dans contrat :

```

LOAD DATA LOCAL INFILE
'C:/Users/Yann/Documents/Projet3BI/Contrat_v2_bomless.csv'
INTO TABLE Contrat
CHARACTER SET utf8mb4
FIELDS TERMINATED BY ';'
ENCLOSED BY '"'
ESCAPED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES
(@contrat_id, @no_voie, @btq, @type_voie, @voie, @code_comm, @cp, @surface,
 @type_local, @occupation, @type_contrat, @formule, @valeur_decl,
 @prix_mensuel)
SET
  Contrat_ID = CAST(NULLIF(TRIM(@contrat_id), '') AS UNSIGNED),
  No_voie = CASE
    WHEN TRIM(@no_voie) = '' THEN NULL
    ELSE CAST(NULLIF(REGEXP_REPLACE(TRIM(@no_voie), '^[^0-9]',
    ''), '' ) AS UNSIGNED)
  END,
  Surface = CAST(NULLIF(REGEXP_REPLACE(TRIM(@surface), '^[^0-9]', ''), '' )
AS UNSIGNED),
  Prix_cotisation_mensuel = CAST(NULLIF(REGEXP_REPLACE(TRIM(@prix_mensuel),
'^[^0-9]', ''), '' ) AS UNSIGNED),

  Code_dep_code_commune = (
    CASE
      WHEN REGEXP_LIKE(REPLACE(REPLACE(REPLACE(TRIM(@code_comm), ' ', ''), ' ', ''), '- ', '' ), '^[0-9]{4}$')
      THEN LPAD(REPLACE(REPLACE(REPLACE(TRIM(@code_comm), ' ', ''), ' ', ''), '- ', '' ), 5, '0')
      ELSE UPPER(REPLACE(REPLACE(REPLACE(TRIM(@code_comm), ' ', ''), ' ', ''), '- ', '' ))
    END
  ),

  Code_postal = LPAD(REGEXP_REPLACE(TRIM(@cp), '^[^0-9]', ''), 5, '0'),

  B_T_Q          = NULLIF(TRIM(@btq), ''),
  Type_de_voie   = TRIM(@type_voie),
  Voie           = TRIM(@voie),
  Type_local     = TRIM(@type_local),
  Occupation     = TRIM(@occupation),

```

```
Type_contrat      = TRIM(@type_contrat),  
Formule           = TRIM(@formule),  
Valeur_declaree_biens = TRIM(@valeur_decl);
```

7. Vérification de la base chargée.

Voici les captures d'écran avec les base chargé via un count(*) dans chacune des tables.

