DAT’Assur Habitation

Document technique

Analyse de la base de données de Dat'Assur Habitation

André Mamdouba Bangoura
04/08/2024

Table des matières

1. Base de données chargées DAT'Assur.....2

1.1 Première table CONTRATS 2

1.2 Deuxième table REGIONS 2

2. EXPLORATION TYPES DE DONNEES 3

2.1 TABLE CONTRATS 3

2.1.1 Longueur maximale des valeurs de chaque colonne de la table CONTRATS 3

2.1.2 Vérification du contenu alphanumérique des champs de la table CONTRATS ..5

2.2 TABLE REGIONS 7

2.2.1 Longueur maximale des valeurs de chaque colonne de la table REGIONS :.....7

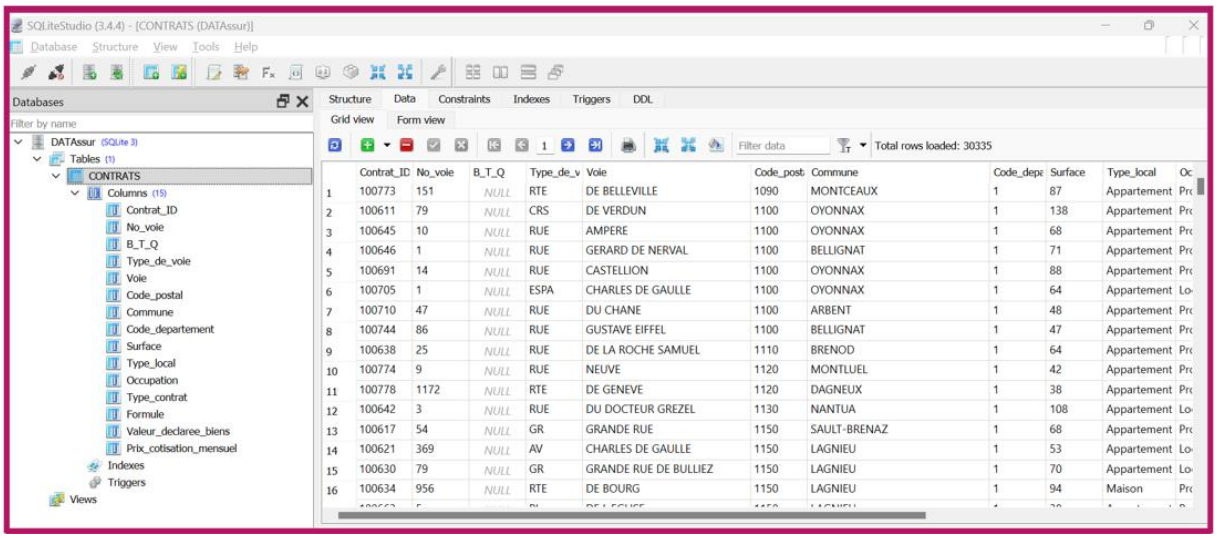
2.2.2 Vérification du contenu alphanumérique des champs de la table REGIONS 8

3. Dictionnaire de données.....9

4. Modèle d'aide à la création de requêtes 9

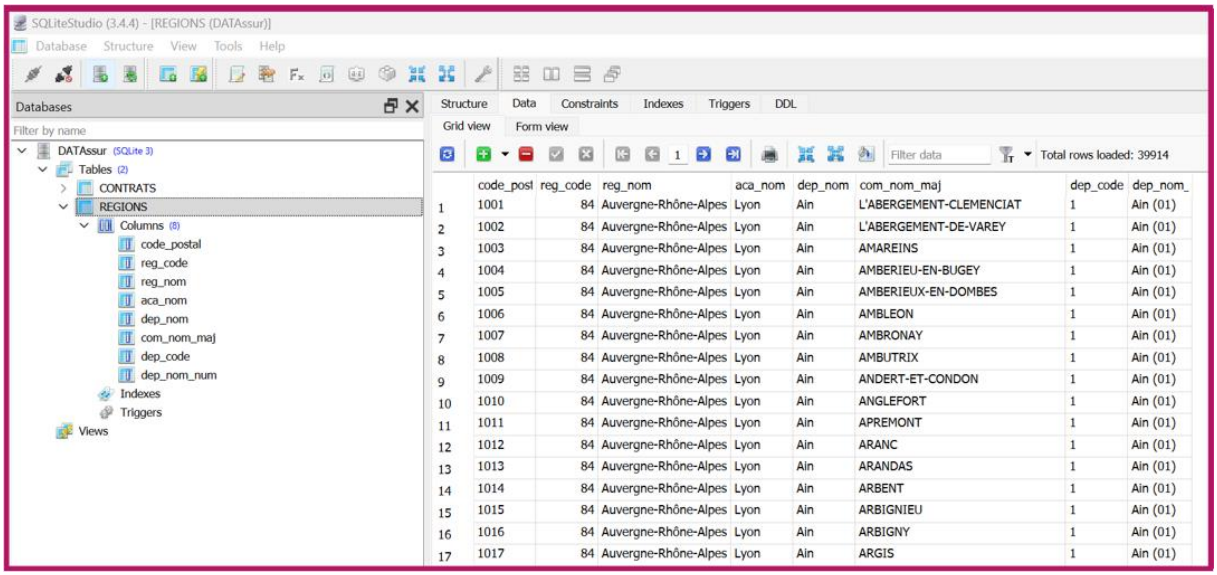
1. Base de données chargées DAT'Assur

1.1 Première table CONTRATS :



	Contrat_ID	No_voie	B_T_Q	Type_de_v	Voie	Code_post	Commune	Code_depa	Surface	Type_local	Oc
1	100773	151	NULL	RTE	DE BELLEVILLE	1090	MONTCEAUX	1	87	Appartement	Pr
2	100611	79	NULL	CRS	DE VERDUN	1100	OYONNAX	1	138	Appartement	Pr
3	100645	10	NULL	RUE	AMPERE	1100	OYONNAX	1	68	Appartement	Pr
4	100646	1	NULL	RUE	GERARD DE NERVAL	1100	BELLIGNAT	1	71	Appartement	Pr
5	100691	14	NULL	RUE	CASTELLION	1100	OYONNAX	1	88	Appartement	Pr
6	100705	1	NULL	ESPA	CHARLES DE GAULLE	1100	OYONNAX	1	64	Appartement	Lo
7	100710	47	NULL	RUE	DU CHANE	1100	ARBENT	1	48	Appartement	Pr
8	100744	86	NULL	RUE	GUSTAVE EIFFEL	1100	BELLIGNAT	1	47	Appartement	Pr
9	100638	25	NULL	RUE	DE LA ROCHE SAMUEL	1110	BRENOD	1	64	Appartement	Pr
10	100774	9	NULL	RUE	NEUVE	1120	MONTLUEL	1	42	Appartement	Pr
11	100778	1172	NULL	RTE	DE GENEVE	1120	DAGNEUX	1	38	Appartement	Pr
12	100642	3	NULL	RUE	DU DOCTEUR GREZEL	1130	NANTUA	1	108	Appartement	Lo
13	100617	54	NULL	GR	GRANDE RUE	1150	SAULT-BRENAZ	1	68	Appartement	Pr
14	100621	369	NULL	AV	CHARLES DE GAULLE	1150	LAGNIEU	1	53	Appartement	Lo
15	100630	79	NULL	GR	GRANDE RUE DE BULLIEZ	1150	LAGNIEU	1	70	Appartement	Lo
16	100634	956	NULL	RTE	DE BOURG	1150	LAGNIEU	1	94	Maison	Pr

1.2 Deuxième table REGIONS :



	code_post	reg_code	reg_nom	aca_nom	dep_nom	com_nom_maj	dep_code	dep_nom_
1	1001	84	Auvergne-Rhône-Alpes	Lyon	Ain	L'ABERGEMENT-CLEMENCIAT	1	Ain (01)
2	1002	84	Auvergne-Rhône-Alpes	Lyon	Ain	L'ABERGEMENT-DE-VAREY	1	Ain (01)
3	1003	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMAREINS	1	Ain (01)
4	1004	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBERIEU-EN-BUGEY	1	Ain (01)
5	1005	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBERIEUX-EN-DOMBES	1	Ain (01)
6	1006	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBLEON	1	Ain (01)
7	1007	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBRONAY	1	Ain (01)
8	1008	84	Auvergne-Rhône-Alpes	Lyon	Ain	AMBUTRIX	1	Ain (01)
9	1009	84	Auvergne-Rhône-Alpes	Lyon	Ain	ANDERT-ET-CONDON	1	Ain (01)
10	1010	84	Auvergne-Rhône-Alpes	Lyon	Ain	ANGLEFORT	1	Ain (01)
11	1011	84	Auvergne-Rhône-Alpes	Lyon	Ain	APREMONT	1	Ain (01)
12	1012	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARANC	1	Ain (01)
13	1013	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARANDAS	1	Ain (01)
14	1014	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBENT	1	Ain (01)
15	1015	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBIGNIEU	1	Ain (01)
16	1016	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARBIGNY	1	Ain (01)
17	1017	84	Auvergne-Rhône-Alpes	Lyon	Ain	ARGIS	1	Ain (01)

2. EXPLORATION TYPES DE DONNEES :

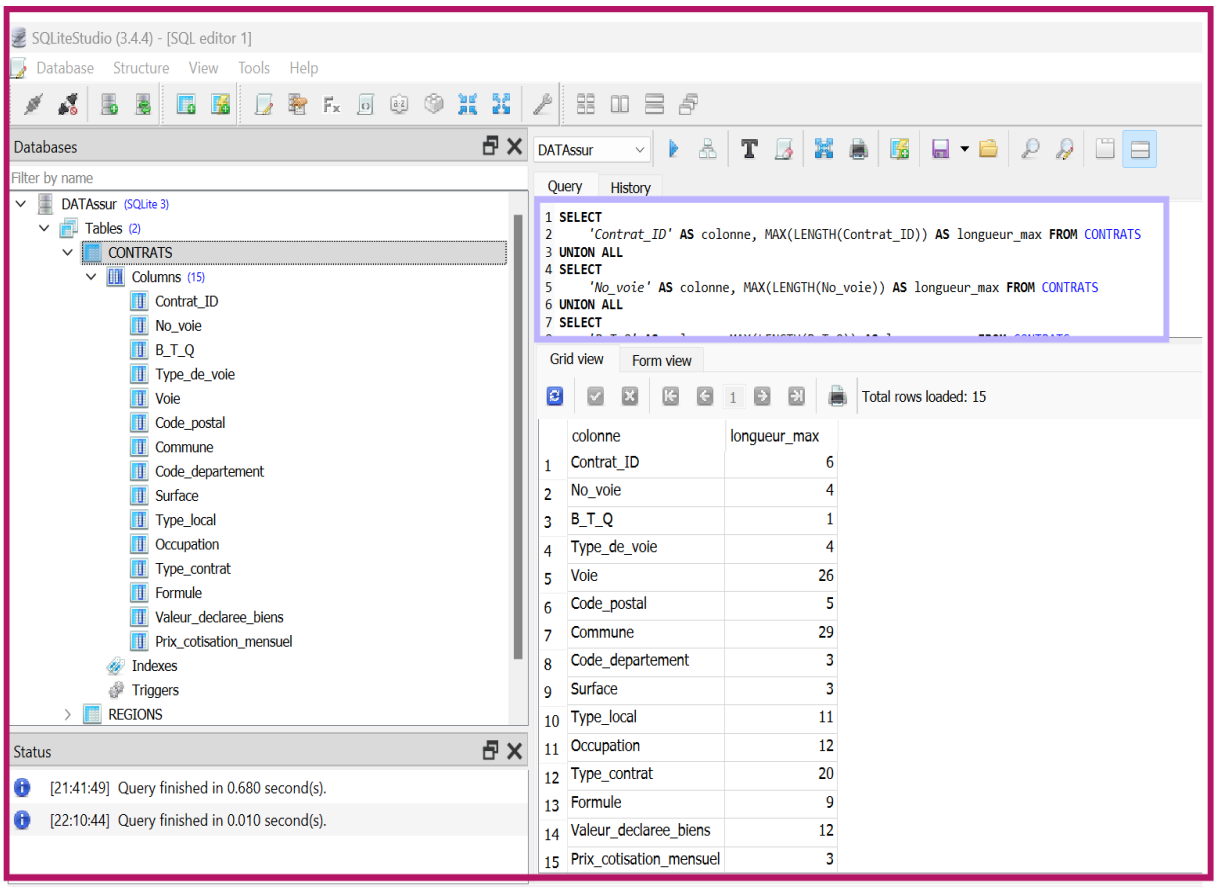
2.1 TABLE CONTRATS :

2.1.1 Longueur maximale des valeurs de chaque colonne de la table CONTRATS :

Longueur maximale des valeurs de chaque colonne de la table CONTRATS :

```sql
SELECT
    'Contrat_ID' AS colonne, MAX(LENGTH(Contrat_ID)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'No_voie' AS colonne, MAX(LENGTH(No_voie)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'B_T_Q' AS colonne, MAX(LENGTH(B_T_Q)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Type_de_voie' AS colonne, MAX(LENGTH(Type_de_voie)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Voie' AS colonne, MAX(LENGTH(Voie)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Code_postal' AS colonne, MAX(LENGTH(Code_postal)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Commune' AS colonne, MAX(LENGTH(Commune)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Code_departement' AS colonne, MAX(LENGTH(Code_departement)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Surface' AS colonne, MAX(LENGTH(Surface)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Type_local' AS colonne, MAX(LENGTH(Type_local)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Occupation' AS colonne, MAX(LENGTH(Occupation)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Type_contrat' AS colonne, MAX(LENGTH(Type_contrat)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Formule' AS colonne, MAX(LENGTH(Formule)) AS longueur_max FROM CONTRATS
UNION ALL
SELECT
    'Valeur_declaree_biens' AS colonne, MAX(LENGTH(Valeur_declaree_biens)) AS longueur_max
FROM CONTRATS
UNION ALL
SELECT
    'Prix_cotisation_mensuel' AS colonne, MAX(LENGTH(Prix_cotisation_mensuel)) AS longueur_max
FROM CONTRATS;
```

RESULTATS :

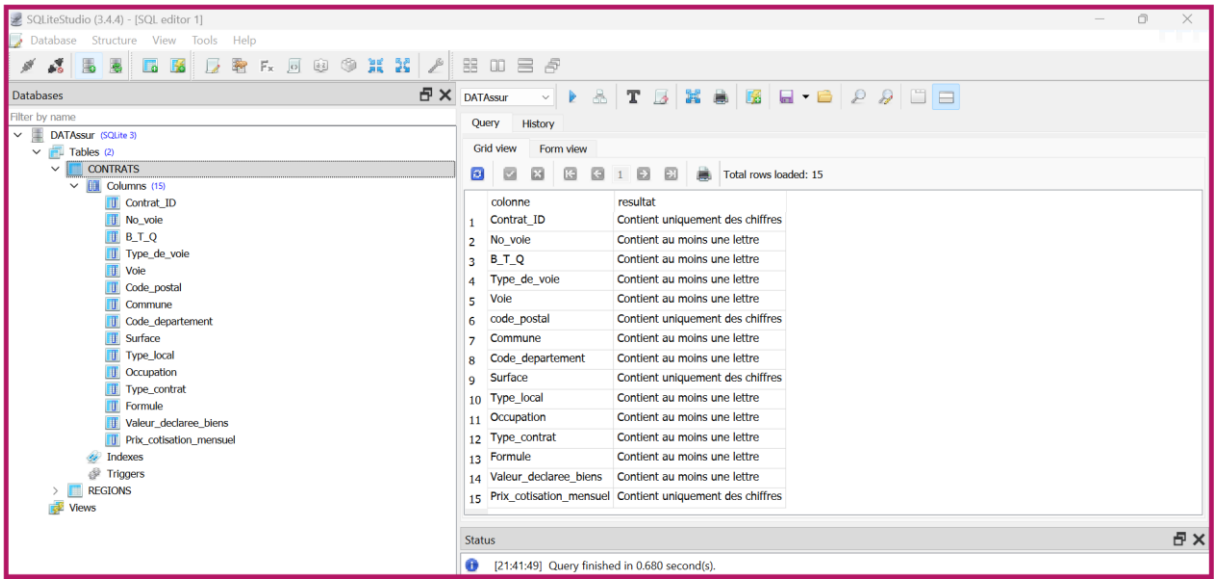

SQLiteStudio (3.4.4) - [SQL editor 1]
Database Structure View Tools Help
Databases
Filter by name
DATAssur (SQLite 3)
Tables (2)
CONTRATS
Columns (15)
Contrat_ID
No_voie
B_T_Q
Type_de_voie
Voie
Code_postal
Commune
Code_departement
Surface
Type_local
Occupation
Type_contrat
Formule
Valeur_declaree_biens
Prix_cotisation_mensuel
Indexes
Triggers
REGIONS
Status
[21:41:49] Query finished in 0.680 second(s).
[22:10:44] Query finished in 0.010 second(s).
DATAssur
Query
History
1 SELECT
2 'Contrat_ID' AS colonne, MAX(LENGTH(Contrat_ID)) AS longueur_max FROM CONTRATS
3 UNION ALL
4 SELECT
5 'No_voie' AS colonne, MAX(LENGTH(No_voie)) AS longueur_max FROM CONTRATS
6 UNION ALL
7 SELECT
Grid view
Form view
Total rows loaded: 15
colonne
longueur_max
1 Contrat_ID 6
2 No_voie 4
3 B_T_Q 1
4 Type_de_voie 4
5 Voie 26
6 Code_postal 5
7 Commune 29
8 Code_departement 3
9 Surface 3
10 Type_local 11
11 Occupation 12
12 Type_contrat 20
13 Formule 9
14 Valeur_declaree_biens 12
15 Prix_cotisation_mensuel 3

2.1.2 Vérification du contenu alphanumérique des champs de la table CONTRATS :

Vérification du contenu alphanumérique des champs de la table CONTRATS :

```
SELECT
    'Contrat_ID' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Contrat_ID NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'No_voie' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE No_voie NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'B_T_Q' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE B_T_Q NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
    UNION ALL
SELECT
    'Type_de_voie' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Type_de_voie NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Voie' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Voie NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'code_postal' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE code_postal NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Commune' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Commune NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Code_departement' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Code_departement NOT REGEXP '^[0-9]+$')
THEN 'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Surface' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Surface NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Type_local' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Type_local NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Occupation' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Occupation NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Type_contrat' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE type_contrat NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Formule' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Formule NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Valeur_declaree_biens' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Valeur_declaree_biens NOT REGEXP '^[0-
9]+$') THEN 'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'Prix_cotisation_mensuel' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM CONTRATS WHERE Prix_cotisation_mensuel NOT REGEXP '^[0-
9]+$') THEN 'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat;
```

RESULTAT :


SQLiteStudio (3.4.4) - [SQL editor 1]
Database Structure View Tools Help
Databases
Filter by name
DATAssur (SQLite 3)
Tables (2)
CONTRATS
Columns (15)
Contrat_ID
No_voie
B_T_Q
Type_de_voie
Voie
Code_postal
Commune
Code_departement
Surface
Type_local
Occupation
Type_contrat
Formule
Valeur_declaree_biens
Prix_cotisation_mensuel
Indexes
Triggers
REGIONS
Views
DATAssur
Query
History
Grid view
Form view
Total rows loaded: 15
colonne
resultat
1 Contrat_ID Contient uniquement des chiffres
2 No_voie Contient au moins une lettre
3 B_T_Q Contient au moins une lettre
4 Type_de_voie Contient au moins une lettre
5 Voie Contient au moins une lettre
6 code_postal Contient uniquement des chiffres
7 Commune Contient au moins une lettre
8 Code_departement Contient au moins une lettre
9 Surface Contient uniquement des chiffres
10 Type_local Contient au moins une lettre
11 Occupation Contient au moins une lettre
12 Type_contrat Contient au moins une lettre
13 Formule Contient au moins une lettre
14 Valeur_declaree_biens Contient au moins une lettre
15 Prix_cotisation_mensuel Contient uniquement des chiffres
Status
[21:41:49] Query finished in 0.680 second(s).

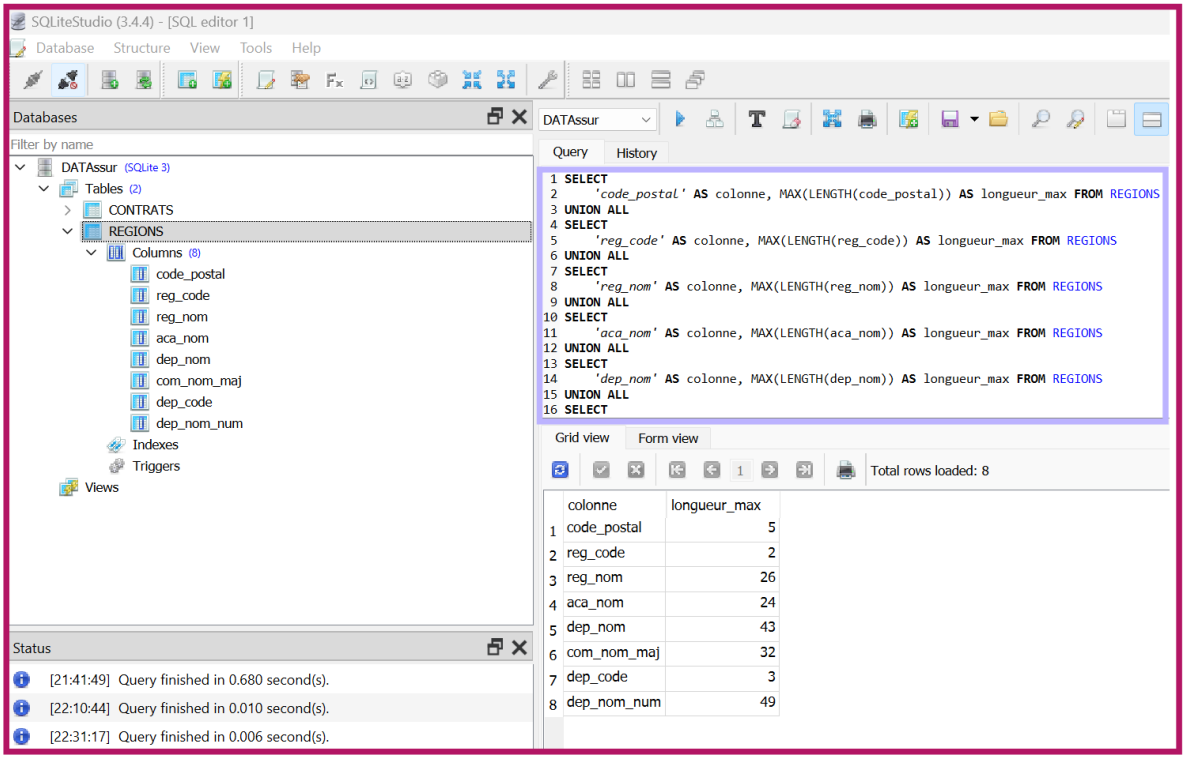
2.2 TABLE REGIONS :

2.2.1 Longueur maximale des valeurs de chaque colonne de la table REGIONS :

Longueur maximale des valeurs de chaque colonne de la table REGIONS :

```sql
SELECT
    'code_postal' AS colonne, MAX(LENGTH(code_postal)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'reg_code' AS colonne, MAX(LENGTH(reg_code)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'reg_nom' AS colonne, MAX(LENGTH(reg_nom)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'aca_nom' AS colonne, MAX(LENGTH(aca_nom)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'dep_nom' AS colonne, MAX(LENGTH(dep_nom)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'com_nom_maj' AS colonne, MAX(LENGTH(com_nom_maj)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'dep_code' AS colonne, MAX(LENGTH(dep_code)) AS longueur_max FROM REGIONS
UNION ALL
SELECT
    'dep_nom_num' AS colonne, MAX(LENGTH(dep_nom_num)) AS longueur_max FROM REGIONS;
```

RESULTATS :

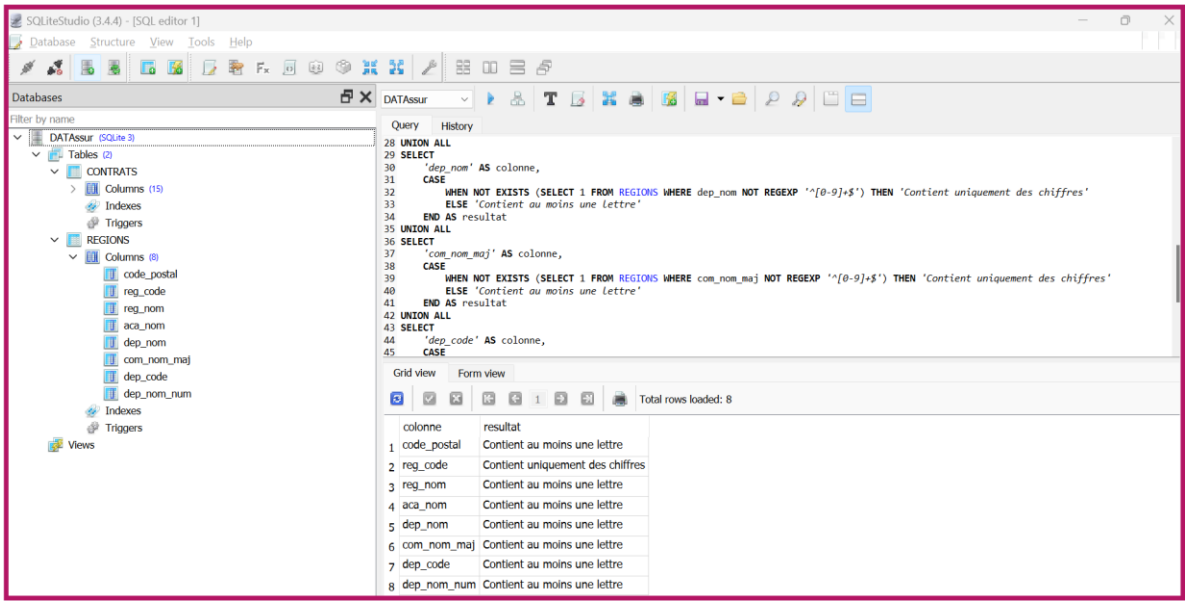


2.2.2 Vérification du contenu alphanumérique des champs de la table REGIONS :

Vérification du contenu alphanumérique des champs de la table REGIONS :

```sql
SELECT
    'code_postal' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE code_postal NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'reg_code' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE reg_code NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'reg_nom' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE reg_nom NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
    UNION ALL
SELECT
    'aca_nom' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE aca_nom NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'dep_nom' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE dep_nom NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'com_nom_maj' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE com_nom_maj NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'dep_code' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE dep_code NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat
UNION ALL
SELECT
    'dep_nom_num' AS colonne,
    CASE
        WHEN NOT EXISTS (SELECT 1 FROM REGIONS WHERE dep_nom_num NOT REGEXP '^[0-9]+$') THEN
'Contient uniquement des chiffres'
        ELSE 'Contient au moins une lettre'
    END AS resultat;
```

RESULTATS :



3. Dictionnaire de données :

Nom du champs	Type de données	Taille	Contrainte	Description	Table
Contrat_ID	INT		Clé primaire	Id unique pour les contrats	Contrat
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	
Type_de_voie	CHAR	4		Type de voie pour l'adresse du logement assuré: rue, av (Avenue), rte (Route), ...	
Voie	CHAR	26		Libellé de la voie pour l'adresse du logement assuré	
Code_postal	CHAR	5		Code postal pour l'adresse du logement assuré	
Commune	CHAR	29		Libellé de la commune de l'adresse du logement	
Code_departement	INT			Code du departement pour l'adresse du logement	
Surface	INT			Surface du logement	
Type_local	CHAR	11		Type de logement (Maison, Appartement)	
Occupation	CHAR	12		Type d'occupation (Locataire ou Propriétaire)	
Type_contrat	CHAR	20		Type de contrat (Residence principale, secondaire ou mise en location)	
Formule	CHAR	9		Type de formule choisi par l'assuré (Integral ou Classique)	
Valeur_declare_mobilier	CHAR	12		Montant déclaré par l'assuré. Cette valeur sert de base à l'indemnisation en cas de sinistre.	
Prix_cotisation_mensuel	INT			La cotisation d'assurance correspond à la somme que doit payer l'assuré chaque mois,	
code_postal	CHAR	5	Clé primaire	code commune (referentiel-geographique-francais, source www.data.gouv.fr)	Region
reg_code	INT			Code région	
reg_nom	CHAR	26		Libellé région	
aca_nom	CHAR	24		Libellé de l'académie	
dep_nom	CHAR	43		Libellé du département	
com_nom_maj	CHAR	32		Libellé de la commune en majuscule	
dep_code	CHAR	3		Code département	
dep_nom_num	CHAR	49		Libellé du département et code	

4. Modèle d'aide à la création de requêtes :

	Besoin	Informations recherchées	Clauses	Requete
1	Quels sont les contrats avec le prix de la cotisation et leur surface pour les appartements ?	Contrats, prix de la cotisation, surface	appartements	Select contrat_id, prix_cotisation_mensuel, surface from contrat c where type_local = 'Appartement'
2	Lister les numéros de contrats (contrat_ID) avec leur surface pour la commune de Caen	Contrats, surface	CAEN	Select contrat_id, surface from CONTRATS where Commune = "CAEN"
3	Lister les numéros de contrats (contrat_ID), avec le type de contrat et leur formule pour les maisons du département de la Saône-et-Loire (Département 71)	Contrats, type de contrat, formule,	Maison, 71	SELECT Contrat_ID, Type_contrat, Formule FROM CONTRATS WHERE Type_local = 'maison' AND Code_departement = '71';
4	Lister le nom des régions de France	Nom des regions	reg_nom	SELECT DISTINCT reg_nom FROM REGIONS