

PROJET 3

Requêtez une base de
données avec SQL

PLAN

1

- Installation de SQLite Studio

2

- Dictionnaire des données

3

- Création de la base de données

4

- Importation des données

5

- Définir les structures des données

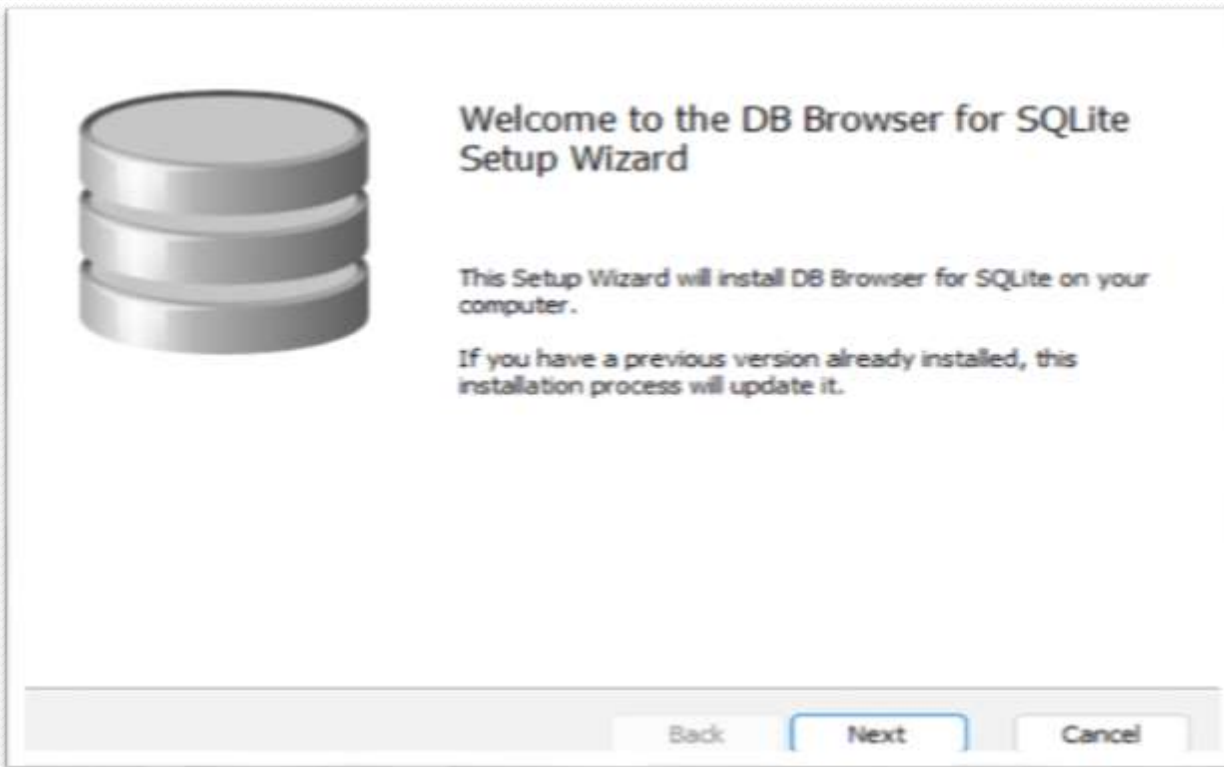
6

- Modèle d'aide à la création de requêtes

7

- Requêtes SQL

Etape 1 : Installation de SQLite Studio



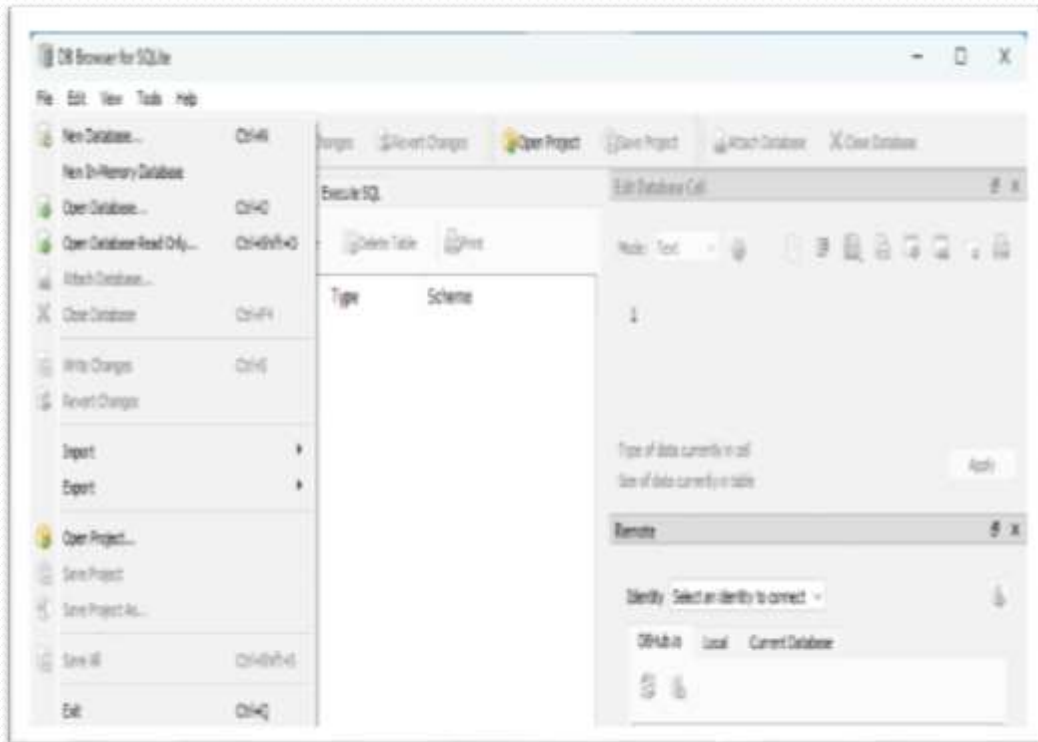
v3.4.4

Etape 2: Dictionnaire des données

dictionnaire_données_etudiant - Microsoft Excel

	A	B	C	D	E	F	G
1							
2		Nom du champs	type de donné	Taille	Contrainte	Description	Table
3		Contrat_ID	INTEGER		Clé primaire	Id unique pour les contrats	Contrat
4		No_voie	INTEGER			Numéro dans la voie pour l'adresse du logement assuré	
5		B_T_Q	TEXT	1		Indicateur éventuel de répétition pour l'adresse du logement assuré	
6		Type_de_voie	TEXT	4		Type de voie pour l'adresse du logement assuré: rue, av (Avenue), rte (Route), ...	
7		Voie	TEXT	30		Libellé de la voie pour l'adresse du logement assuré	
8		Code_postal	INTEGER		NOT NULL	Code postal pour l'adresse du logement assuré	
9		Commune	TEXT	30	NOT NULL	Libellé de la commune de l'adresse du logement	
10		Code_département	INTEGER		NOT NULL	Code du département pour l'adresse du logement	
11		Surface	INTEGER			Surface du logement	
12		Type_local	TEXT	15		Type de logement (Maison, Appartement)	
13		Occupation	TEXT	15		Type d'occupation (Locataire ou Propriétaire)	
14		Type_contrat	TEXT	20	NOT NULL	Type de contrat (Résidence principale, secondaire ou mise en location)	
15		Formule	TEXT	10		Type de formule choisi par l'assuré (Integral ou Classique)	
16		Valeur_declare_mobiliaire	INTEGER			Montant déclaré par l'assuré. Cette valeur sert de base à l'indemnisation en cas de sinistre.	
17		Prix_cotisation_mensuelle	INTEGER			La cotisation d'assurance correspond à la somme que doit payer l'assuré chaque mois.	
18		code_postal	TEXT	5	Clé primaire	code commune (referentiel-geographique-francais, source www.data.gouv.fr)	Region
19		reg_code	INTEGER		NOT NULL	Code région	
20		reg_nom	TEXT	30	NOT NULL	Libellé région	
21		aca_nom	TEXT	30		Libellé de l'académie	
22		dep_nom	TEXT	50	NOT NULL	Libellé du département	
23		com_nom_maj	TEXT	50		Libellé de la commune en majuscule	
24		dep_code	INTEGER		NOT NULL	Code département	
25		dep_nom_num	TEXT	50	NOT NULL	Libellé du département et code	

Etape 3 :Création d'une base de données



- Nom du base de données;

Etape 4.1 : Importation de données

« contrat »

Structure Données Contraintes Index Déclencheurs DDL

Table Formulaire








Etape 4.2 : Importation de données

« region »

Structure Données Contraintes Index Déclencheurs DDL								
Table Formulaire								
Filtre de données Nombre de lignes chargées : 39914								
	code_postal	reg_code	reg_nom	aca_nom	dep_nom	com_nom_maj	dep_code	dep_nom_num
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-DOBES	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	ARCI	1	Ain (01)

Etape 5.1: Structure des données

« contrat »

Structure										
Données										
Contraintes										
Index										
Déclencheurs										
DDL										
my_project_3_c		Nom de la table : contrat		<input type="checkbox"/> WITHOUT ROWID <input type="checkbox"/> STRICT						
	Nom	Type de données	Clé primaire	Clé étrangère	Unique	Contrôle	Non NULL	Collecter	Généré	
1	contrat_ID	INTEGER								NULL
2	NO_voie	INTEGER								NULL
3	B_T_Q	TEXT (1)								NULL
4	Type_voie	TEXT (4)								NULL
5	voie	TEXT (30)								NULL
6	code_postale	INTEGER								NULL
7	commune	TEXT (30)								NULL
8	code_departement	INTEGER								NULL
9	surface	INTEGER								NULL
10	Type_local	TEXT (15)								NULL
11	occupation	TEXT (15)								NULL
12	type_contrat	TEXT (20)								NULL
13	formule	TEXT (10)								NULL
14	Valeur_declaree_biens	INTEGER								NULL

- Types des données;
- Tailles des données;
- Contraintes :
primary key,
foreign key, Not
null

Etape 5.2: Structure des données

« region »

Structure

Données

Contraintes

Index

Déclencheurs

DDL

🔄

✓

✕

📊

📋

📈

⬆️

⬇️

🔗

🌐

🔗

🔗

🔗

🔗

📄

🔄

my_project_3_c

Nom de la table :

☐ WITHOUT ROWID

☐ STRICT

	Nom	Type de données	Clé primaire	Clé étrangère	Unique	Contrôle	Non NULL	Collecter	Généré
1	code_postale	TEXT (5)	🔑						NULL
2	reg_code	INTEGER					🚫		NULL
3	reg_nom	TEXT (30)					🚫		NULL
4	aca_nom	TEXT (30)							NULL
5	dep_nom	TEXT (50)					🚫		NULL
6	com_nom_maj	TEXT (50)							NULL
7	dep_code	INTEGER					🚫		NULL
8	dep_nom_num	TEXT (50)					🚫		NULL






Requête

Historique



```
1 select max(length(code_postale)) as code_postale_max_length,
2       max(length(reg_nom)) as reg_nom_max_length |
3 from region;
```

Table

Formulaire



1

Nombre de lignes chargées : 1

	code postale max length	5	reg nom max length	26
1				

Etape 5.3 : Structure des données (Contraintes)

Structure

Données

Contraintes

Index

Déclencheurs

DDL

🔄

✓

✗

📊

📋

📁

⬆

⬇

🔍

🌐

🔗

🔗

🔗

📄

🔄

my_project_3_c

Nom de la table : region

☐ WITHOUT ROWID
☐ STRICT

	Nom	Type de données	Clé primaire	Clé étrangère	Unique	Contrôle	Non NULL	Collecter	Généré
1	code_postale	TEXT (5)	🔑						
2	reg_code	INTEGER					🚫		
3	reg_nom	TEXT (30)					🚫		
4	aca_nom	TEXT (30)							
5	dep_nom	TEXT (50)					🚫		
6	com_nom_maj	TEXT (50)							
7	dep_code	INTEGER					🚫		
8	dep_nom_num	TEXT (50)					🚫		

Structure
















Données

Contraintes

Index

Déclencheurs

DDL










my_project_3_c

Nom de la table :




☐ WITHOUT ROWID

☐ STRICT





	Nom	Type de données	Clé primaire	Clé étrangère	Unique	Contrôle	Non NULL	Collecter	Généré
1	contrat_ID	INTEGER							
2	NO_voie	INTEGER							
3	B_T_Q	TEXT (1)							
4	Type_voie	TEXT (4)							
5	voie	TEXT (30)							
6	code_postale	INTEGER							
7	commune	TEXT (30)							
8	code_departement	INTEGER							
9	surface	INTEGER							
10	Type_local	TEXT (15)							
11	occupation	TEXT (15)							
12	type contrat	TEXT (20)							

Etape 5.4 : Structure des données (Indexes)

Structure Données Contraintes Index Déclencheurs DDL

Nom	Unique	
primary_index	<input checked="" type="checkbox"/>	contrat_ID COLLATE BINARY

Structure	Données	Contraintes	Index	Déclencheurs	DDL
   					
Nom		Unique			
sqlite_autoindex_region_1		<input checked="" type="checkbox"/>	code_postale COLLATE BINARY		

Etape 6: Modèle d'aide à la création de requêtes

Template_aide_redaction_requete - Microsoft Excel

	B	C	D	E
1	Besoin	Informations recherchées	Clauses	Requete
2	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'
3	Lister les numéros de contrats (contrat_ID) avec leur surface pour la commune de Caen	contrat_ID,surface,commune ='Caen'	commune	select contrat_id, surface from contrat where upper(commune) = 'CAEN';
4	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)	contrat_ID,type_contrat,formule	Maisons	select contrat_id, type_contrat, Formule from contrat where type_local = 'Maison' and code_departement = 71;
5	Lister le nom des régions de France	reg_nom	Regions	select distinct reg_nom from region;

Etape 7.1: Requêtes SQL

The screenshot shows a database query interface. At the top, there are two tabs: 'Requête' (selected) and 'Historique'. Below the tabs, the SQL query is displayed in a text area:

```
1 select contrat_id, surface
2 from contrat
3 where upper(commune) = 'CAEN';
```

Below the query area, there are two tabs: 'Table' (selected) and 'Formulaire'. Below the tabs, there is a toolbar with various icons (refresh, check, close, back, forward, first, last, print) and a status bar that reads 'Nombre de lignes chargées : 4'. Below the toolbar, the query results are displayed in a table:

	contrat id	surface
1	103791	35
2	103792	99
3	103793	40
4	103794	20

- Numéros de contrats avec leur surface pour la commune de Caen;
- Utilisation de la fonction « **UPPER** »

Etape 7.2: Requêtes SQL


Requête


Historique


1 select contrat_ID, type_contrat, Formule
2 from contrat where type_local = 'Maison' and code_departement = 71;


Table


Formulaire














1















Nombre de lignes chargées : 4

	contrat ID	type contrat	Formule
1	114768	Residence principale	Integral
2	114779	Residence principale	Classique
3	114782	Residence principale	Classique
4	114812	Residence principale	Integral

➤ Numéros de contrats avec le type de contrat et leur formule pour les maisons du département de la « Saône-et-Loire » (Département 71)

Etape 7.3: Requêtes SQL

Requête		Historique
<pre>1 select distinct reg_nom as 'Noms des régions de France' 2 from region;</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 19
1	Noms des régions de France	
2	Auvergne-Rhône-Alpes	
3	Hauts-de-France	
4	Provence-Alpes-Côte d'Azur	
5	Grand Est	
6	Occitanie	
7	Normandie	
8	Nouvelle-Aquitaine	
9	Centre-Val de Loire	
10	Corse	
11	Bourgogne-Franche-Comté	
12	Bretagne	
13	Pays de la Loire	
14	Île-de-France	
15	Guadeloupe	









➤ Noms des régions de France

➤ Utilisation de :

➤ **DISTINCT**

➤ **ALIAS**

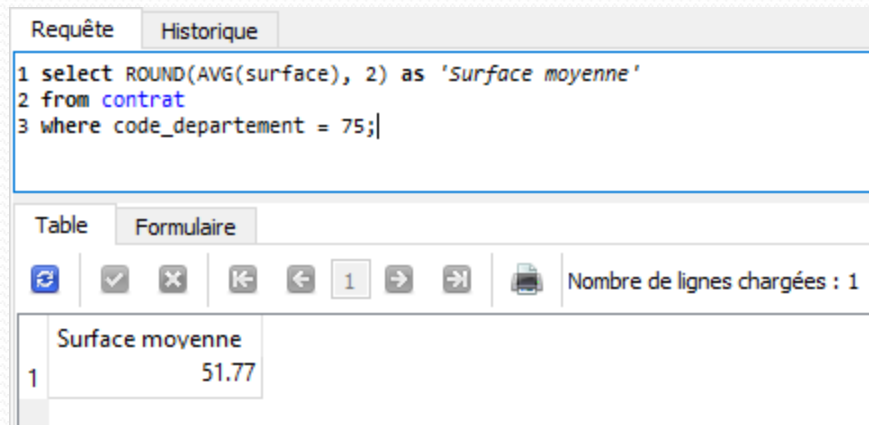
Etape 7.4: Requêtes SQL

Requête		Historique
<pre>1 select COUNT(contrat_ID) as 'Nombre de contrats' 2 from contrat 3 where type_contrat = 'Residence principale';</pre>		
Table		Formulaire
     1  		 Nombre de lignes chargées : 1
1	Nombre de contrats	25620

➤ Nombre de contrats sur les résidences principales

➤ ALIAS

Etape 7.5: Requêtes SQL



The screenshot shows a database application interface. At the top, there are two tabs: 'Requête' (selected) and 'Historique'. Below the tabs, the SQL query is displayed in a text area:









```
1 select ROUND(AVG(surface), 2) as 'Surface moyenne'
2 from contrat
3 where code_departement = 75;
```

Below the query, there are two more tabs: 'Table' (selected) and 'Formulaire'. Under the 'Table' tab, there is a toolbar with various icons (refresh, check, close, back, forward, first, last, print) and a status bar that reads 'Nombre de lignes chargées : 1'. Below the toolbar, a table displays the results of the query:

	Surface moyenne
1	51.77









- Surface moyenne des logements avec un contrat à Paris;
- Utilisation des fonction:
- **ROUND**
- **AVG**

Etape 7.6: Requêtes SQL

Requête		Historique
<pre>1 select contrat_ID, surface 2 from contrat 3 order by surface desc 4 LIMIT 5</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 5
	contrat ID	surface
1	104211	815
2	105463	742
3	130878	595
4	100822	570
5	109872	559









- Les 5 contrats qui ont les surfaces les plus élevées;
- Utilisation de :
- **ORDER BY DESC**
- **LIMIT**

Etape 7.7: Requêtes SQL

Requête		Historique
<pre>1 select ROUND(AVG(prix_cotisation_mensuel),2) AS 'Prix moyen de cotisation mensuelle' 2 from contrat;</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 1
Prix moyen de cotisation mensuelle		
1	19.33	








- Prix moyen de cotisation mensuelle;
- **ALIAS**
- **ROUND**
- **AVG**

Etape 7.8: Requêtes SQL

Requête		Historique
<pre>1 select valeur_declaree_biens as 'Catégorie de prix', 2 count(contrat_ID) as 'Nombre des contrats' 3 from contrat 4 Group by valeur_declaree_biens; 5</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 4
	Catégorie de prix	Nombre des contrats
1	0-25000	22720
2	100000+	104
3	25000-50000	6815
4	50000-100000	696

- Nombre des contrats par catégorie de prix de la valeur déclarée des biens
- Utilisation
- **ALIAS**
- **GROUP BY**









Etape 7.9: Requêtes SQL

Requête		Historique
<pre>1 select region.dep_nom, ROUND(AVG(Prix_cotisation_mensuel),2) as 'Prix moyen de cotisation' 2 from contrat join region on region.dep_code = contrat.code_departement 3 group by code_departement 4 order by AVG(Prix_cotisation_mensuel) desc 5 limit 10;</pre>		
Table		Formulaire
     1  		Nombre de lignes chargées : 10
	dep nom	Prix moyen de cotisation
1	Paris	36.4
2	Hauts-de-Seine	26.27
3	Val-de-Marne	19.82
4	Yvelines	18.88
5	Rhône	18.46
6	Ain	18.24
7	Alpes-Maritimes	18.14
8	Charente-Maritime	17.32
9	Haute-Savoie	17.16
10	Corse-du-Sud	17.07

- ROUND
- AVG
- ALIAS
- JOIN
- GROUP BY
- ORDER BY
- DESC
- LIMIT









➤ Les 10 départements où le prix moyen de cotisation est le plus élevé

Etape 7.10: Requêtes SQL

Requête		Historique
<pre>1 select count(c.contrat_ID) as 'Nombre des contrats Integral région Pays de La Loire' 2 from contrat c join region r on r.dep_code = c.code_departement 3 where c.formule = 'Integral' and r.reg_nom = 'Pays de La Loire';</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 1
Nombre des contrats Integral région Pays de la Loire		
1	178215	

- Nombre de contrats avec des formules « Intégral » pour la région Pays de la Loire
- Utilisation de :
- **ALIAS**
- **COUNT**
- **JOIN**









Etape 7.11: Requêtes SQL

Requête		Historique
<pre>1 select commune, count(contrat_ID) as 'Nombre des contrats' 2 from contrat group by commune having count(contrat_ID) >= 150;</pre>		
Table		Formulaire
     1   		Nombre de lignes chargées : 20
	commune	Nombre des contrats
1	BORDEAUX	302
2	COURBEVOIE	163
3	GRENOBLE	220
4	LILLE	161
5	NANTES	291
6	NICE	387
7	PARIS 03	159
8	PARIS 09	204
9	PARIS 10	263
10	PARIS 11	381
11	PARIS 12	252
12	PARIS 14	222
13	PARIS 15	407
14	PARIS 16	394

- ALIAS
- GROUP BY
- HAVING

➤ Communes ayant eu au moins 150 contrats

Etape 7.12: Requêtes SQL

Requête		Historique
<pre>1 select r.reg_nom as 'Région', count(c.contrat_ID) as 'Nombre des contrats' 2 from region r join contrat c on c.code_departement = r.dep_code 3 group by r.reg_nom;</pre>		
Table		Formulaire
     1    Nombre de lignes chargées : 16		
	Région	Nombre des contrats
1	Auvergne-Rhône-Alpes	1222743
2	Bourgogne-Franche-Comté	159575
3	Bretagne	340449
4	Centre-Val de Loire	212794
5	Corse	38478
6	Grand Est	449865
7	Guyane	999
8	Hauts-de-France	869994
9	La Réunion	476
10	Martinique	2482
11	Normandie	613868
12	Nouvelle-Aquitaine	1042026
13	Occitanie	668557
14	Pays de la Loire	358841

- ALIAS
- JOIN
- GROUP BY

- Nombre des contrats pour chaque région

CONCLUSION

- ➔ Importer des fichiers CSV
- ➔ Déterminer les tailles des colonnes, contraintes (null or not null)
- ➔ ALIAS
- ➔ JOIN
- ➔ GROUP BY & ORDER BY (ASC/DESC) & HAVING
- ➔ LIMIT
- ➔ AVG / COUNT
- ➔ ROUND(colonne, précision)