

ANALYSE DES REQUÊTES

1/ Lister les numéros de contrats (contrat_ID) avec leur surface pour la commune de Caen.

- **Informations recherchées** : numéros de contrats (contrats_ID) ; surface ; commune Caen
- **Clauses** : select ; from ; where
- **Requête** : **Select** contrat_ID, surface, commune
From contrats_clients
Where commune='CAEN'

The screenshot shows the SQLiteStudio 3.4.4 application window. The left sidebar displays the database structure for 'panamapapers (SQLite 3)', including tables 'contrats' and 'contrats_clients'. The main editor window shows the following SQL query:

```
1 select contrat_id,surface,commune from contrats_clients
2 where commune = 'CAEN'
```

Below the query editor, the results are displayed in a table format. The status bar indicates 'Nombre de lignes chargées : 4'.

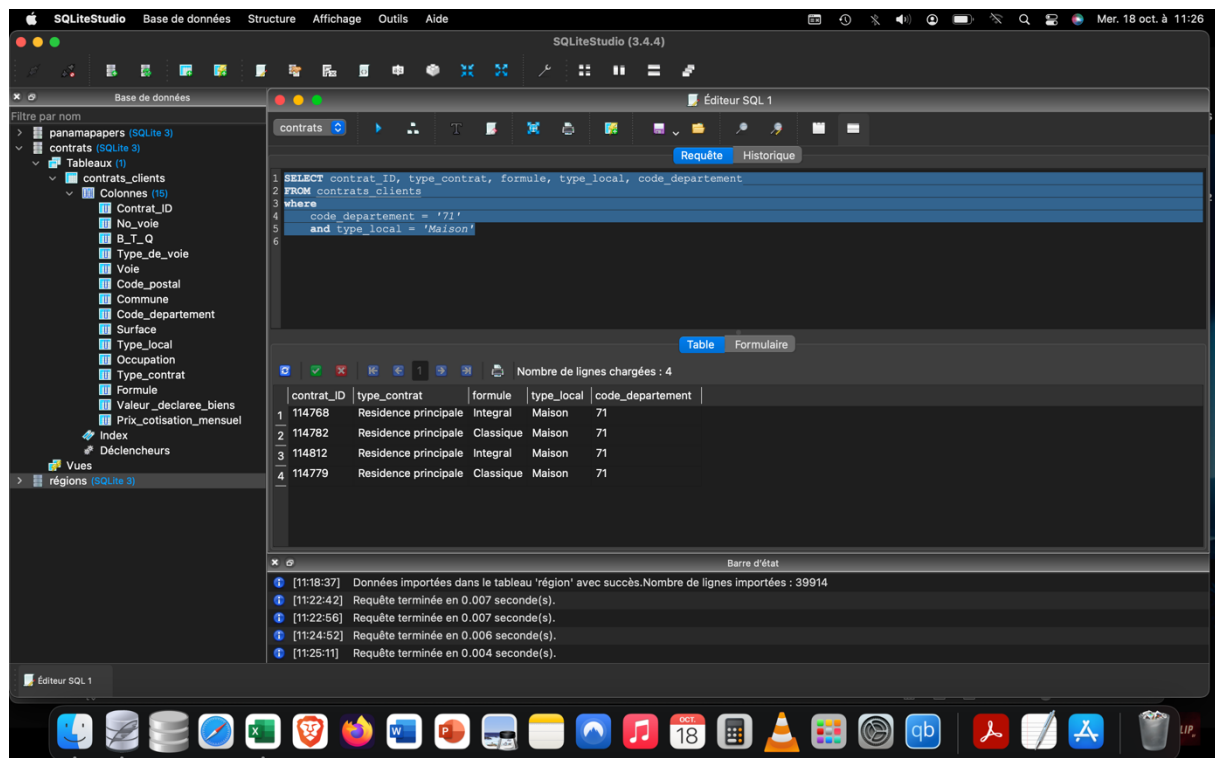
	contrat_id	surface	commune
1	103791	35	CAEN
2	103792	99	CAEN
3	103793	40	CAEN
4	103794	20	CAEN

The bottom status bar shows a log of queries and their execution times:

- [10:21:54] Requête terminée en 0.001 seconde(s).
- [10:23:35] Erreur pendant l'exécution de la requête sur la base de données « contrats » : no such column: contrat_id
- [10:23:52] Requête terminée en 0.001 seconde(s).
- [10:24:38] Requête terminée en 0.001 seconde(s).
- [10:24:47] Requête terminée en 0.002 seconde(s).

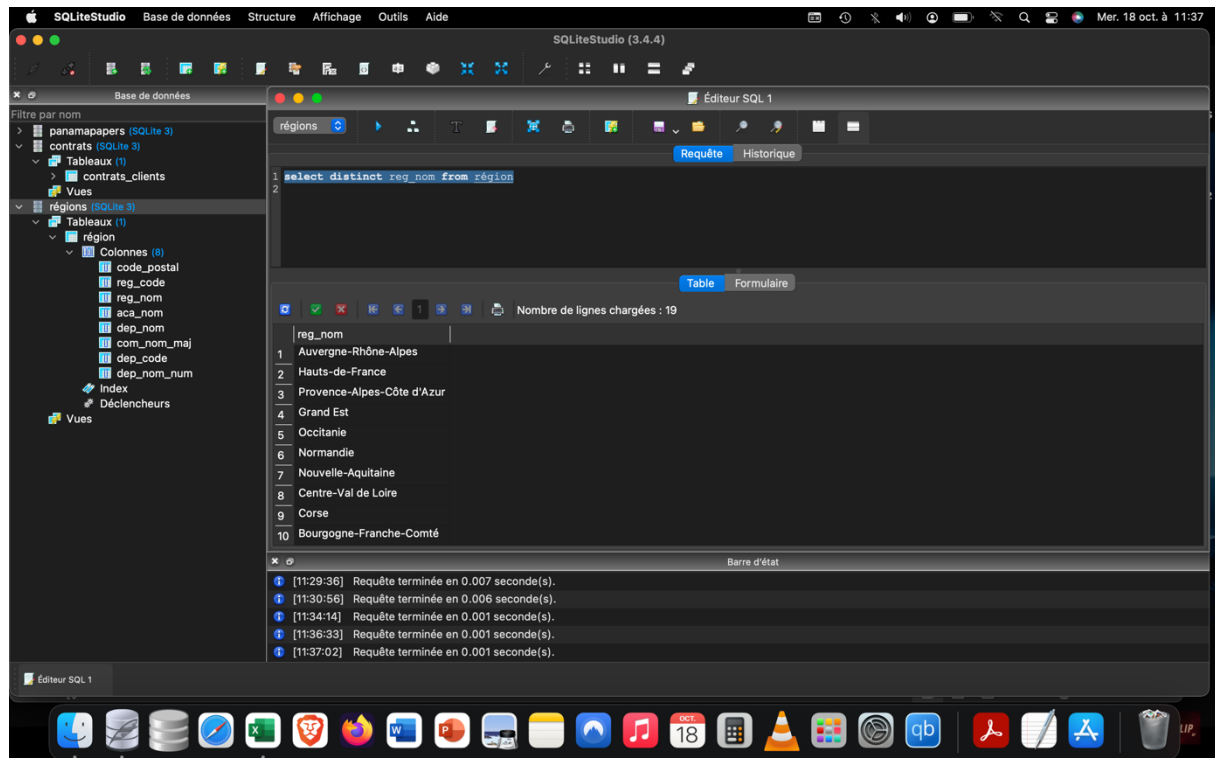
2/ Lister les numéros de contrats, avec le type de contrat et leur formule pour les maisons du département de la Saône-et-Loire.

- **Informations recherchées** : numéros de contrats (contrat_ID) ; type de contrat ; formule ; type local (maison) ; code département 71
- **Clauses** : select ; from; where ; and
- **Requête** : **Select** contrat_ID, type_contrat, formule, type_local, code_departement
From contrats_clients
Where code_departement = '71' **and** type_local = 'Maison'



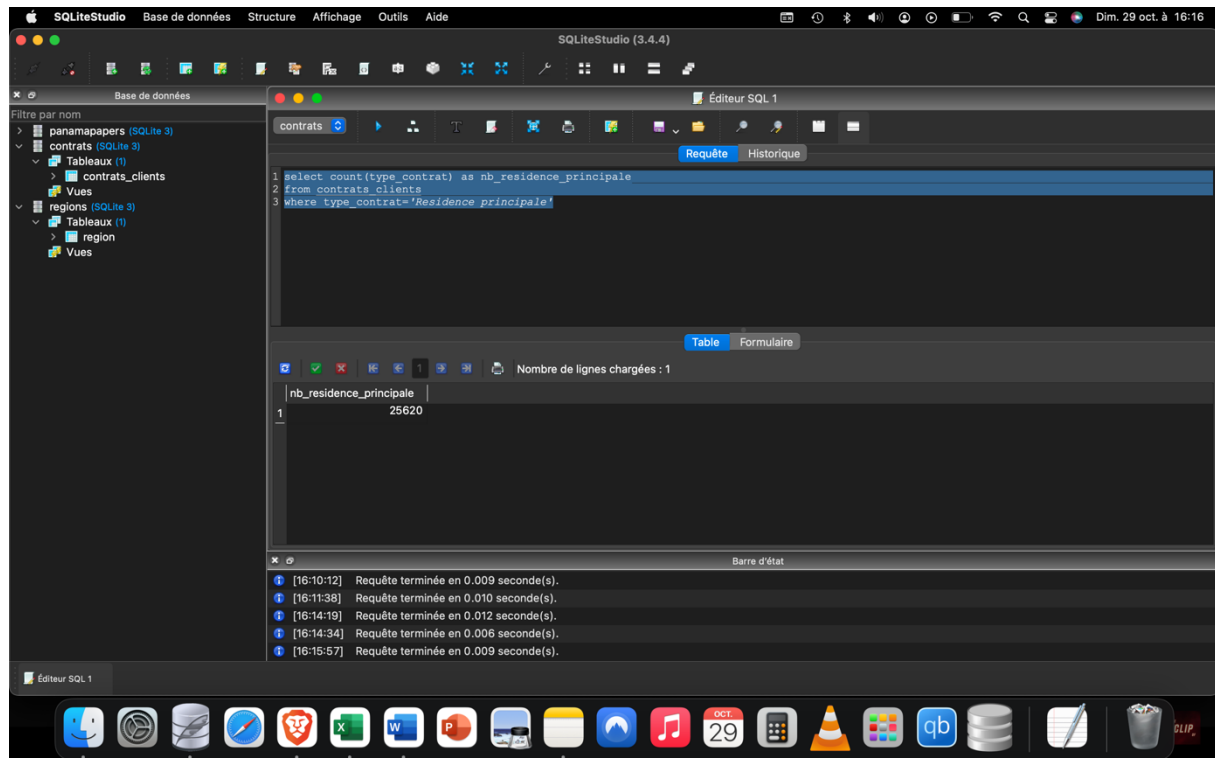
3/ Lister les noms des régions de France.

- **Informations recherchées** : nom des régions de France (reg_nom)
- **Clauses** : select ; distinct
- **Requête** : **Select distinct reg_nom**
From region



4/ Combien existe-t-il de contrats sur les résidences principales ?

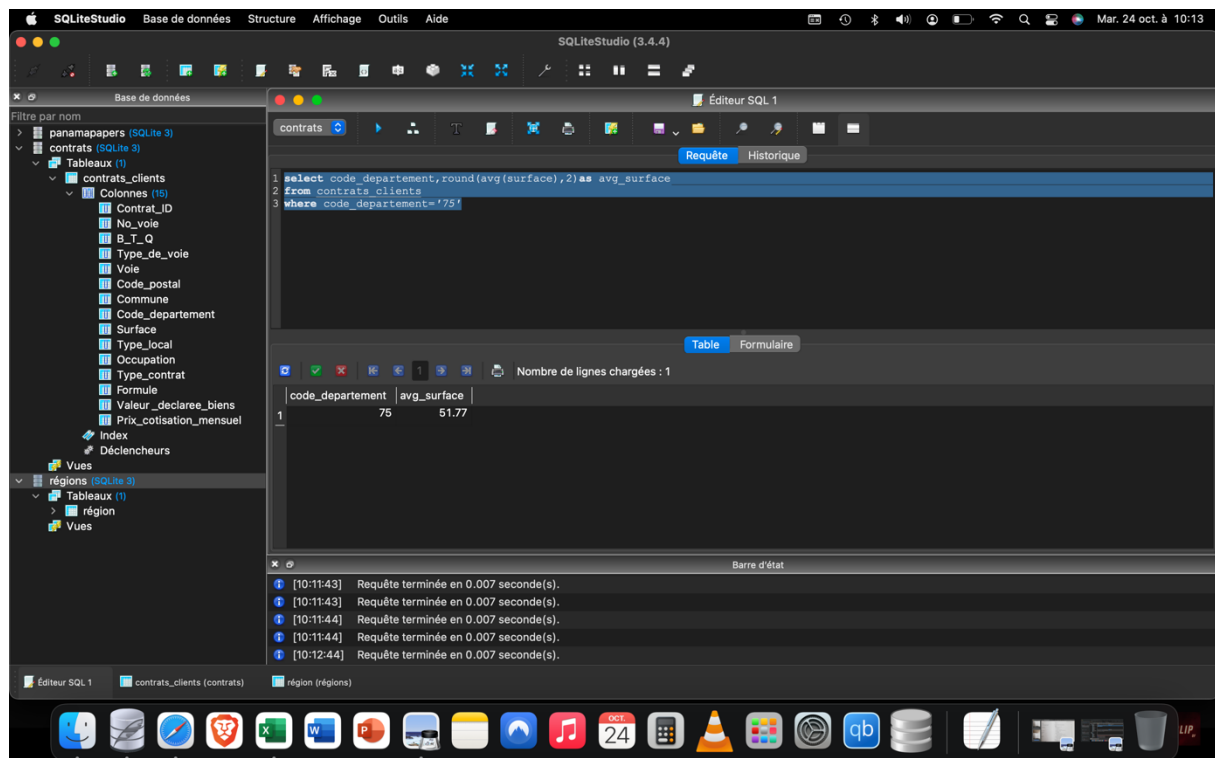
- **Informations recherchées** : nombre de contrats ; résidences principales
- **Clauses** : select ; count() ; from ; as ; where
- **Requête** : **select count(type_contrat) as nb_residence_principale**
From contrats_clients
Where type_contrat='Residence principale'



5/ Quelle est la surface moyenne des logements avec un contrat à Paris ?

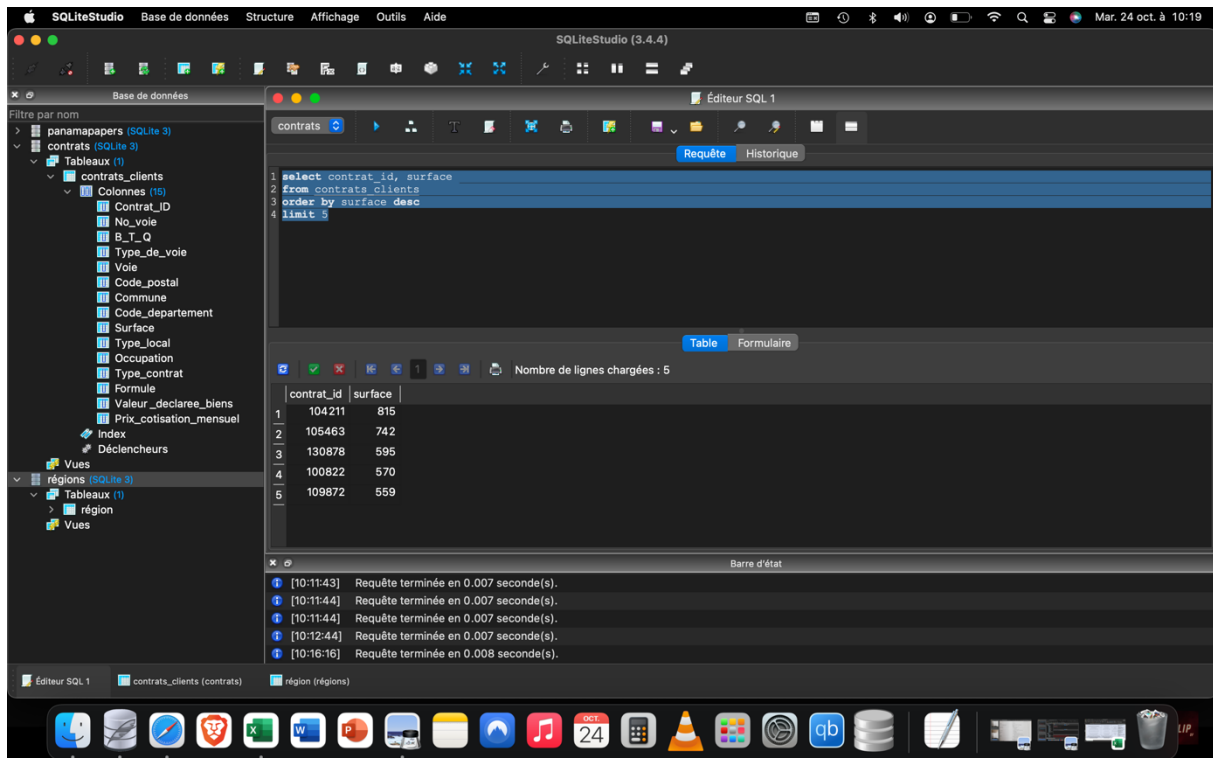
- **Informations recherchées** : surface moyenne ; contrat Paris
- **Clauses** : select ; from ; round AVG() ; as ; where
- **Requête** :

```
Select code_departement,  
Round (AVG(surface),2) as avg_surface  
From contrats_clients  
Where code_departement = '75'
```



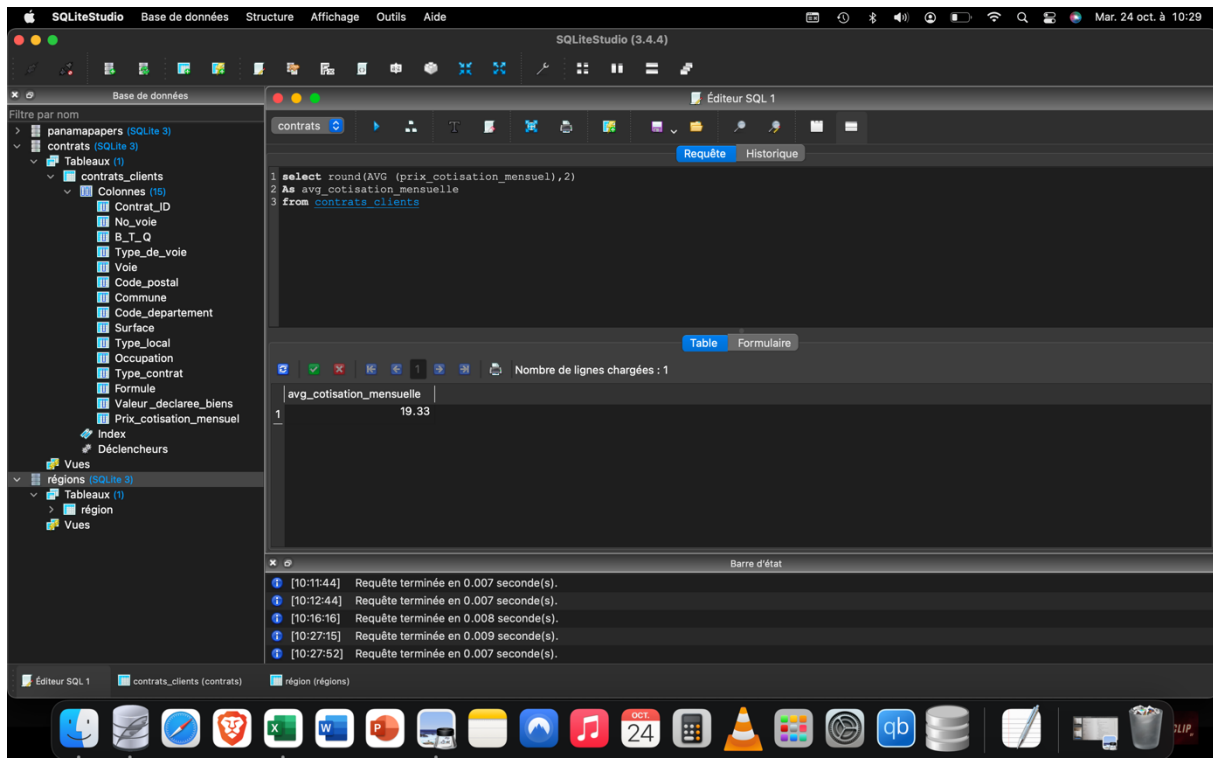
6/ Quels sont les 5 contrats qui ont les surfaces les plus élevées ?

- **Informations recherchées** : les 5 contrats ayant les surfaces les plus élevées
- **Clauses** : select ; from ; order by desc ; limit
- **Requête** : **Select** contrat_ID, surface
From contrats_clients
Order by surface **desc**
Limit 5



7/ Quel est le prix moyen de la cotisation mensuelle ?

- **Informations recherchées** : prix moyen ; cotisation mensuelle
- **Clauses** : select round(AVG) as from
- **Requête** : **Select round(AVG(prix_cotisation_mensuel),2)**
As avg_cotisation_mensuelle
From contrats_clients



8/ Nombre de contrats pour chaque catégorie de prix de la valeur déclarée des biens.

- **Informations recherchées** : nombre de contrats ; catégorie valeur déclarée des biens
- **Clauses** : select ; count() ; from ; group by
- **Requête** : **Select** valeur_declaree_biens, **count(*)**
From contrats_clients
Group by valeur_declaree_biens

The screenshot shows the SQLiteStudio 3.4.4 interface. The left sidebar displays the database structure for 'panamapapiers (SQLite 3)', including the 'contrats' table and its columns. The main editor window shows the following SQL query:

```
1 select valeur_declaree_biens,
2 count(*) from Contrats_Clients
3 group by valeur_declaree_biens
```

Below the query editor, the results are displayed in a table format. The table has two columns: 'valeur_declaree_biens' and 'count(*)'. The results are as follows:

valeur_declaree_biens	count(*)
0-25000	22720
100000+	104
25000-50000	6815
50000-100000	696

The bottom status bar indicates that the query was terminated successfully in 0.022 seconds.

9/ Classement des 10 départements où le prix moyen de la cotisation est le plus élevé ?

- **Informations recherchées** : 10 départements où le prix moyen de la cotisation est le plus élevé.
- **Clauses** : select ; as ; round() ; avg(); from ; group by ; order by desc ; limit
- **Requête** : **Select** code_departement as dep, **Round(AVG(prix_cotisation_mensuel),2)**
As avg_cotisation
From contrats_clients
Group by dep
Order by avg_cotisation desc
Limit 10

The screenshot shows the SQLiteStudio 3.4.4 interface. On the left, the 'Base de données' (Database) pane shows a project named 'panamapapers' containing a database 'contrats' with a table 'contrats_clients'. The table structure is visible, listing columns like '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', 'Index', and 'Déclencheurs'. The main 'Éditeur SQL 1' pane contains the following SQL query:

```
1 select code_departement as dep, round(avg(prix_cotisation_mensuel),2) as avg_cotisation
2 from contrats_clients
3 group by dep
4 order by avg_cotisation desc
5 limit 10
6
7
```

Below the query editor, the 'Table' tab displays the results of the query in a table with 10 rows and 2 columns: 'dep' and 'avg_cotisation'.

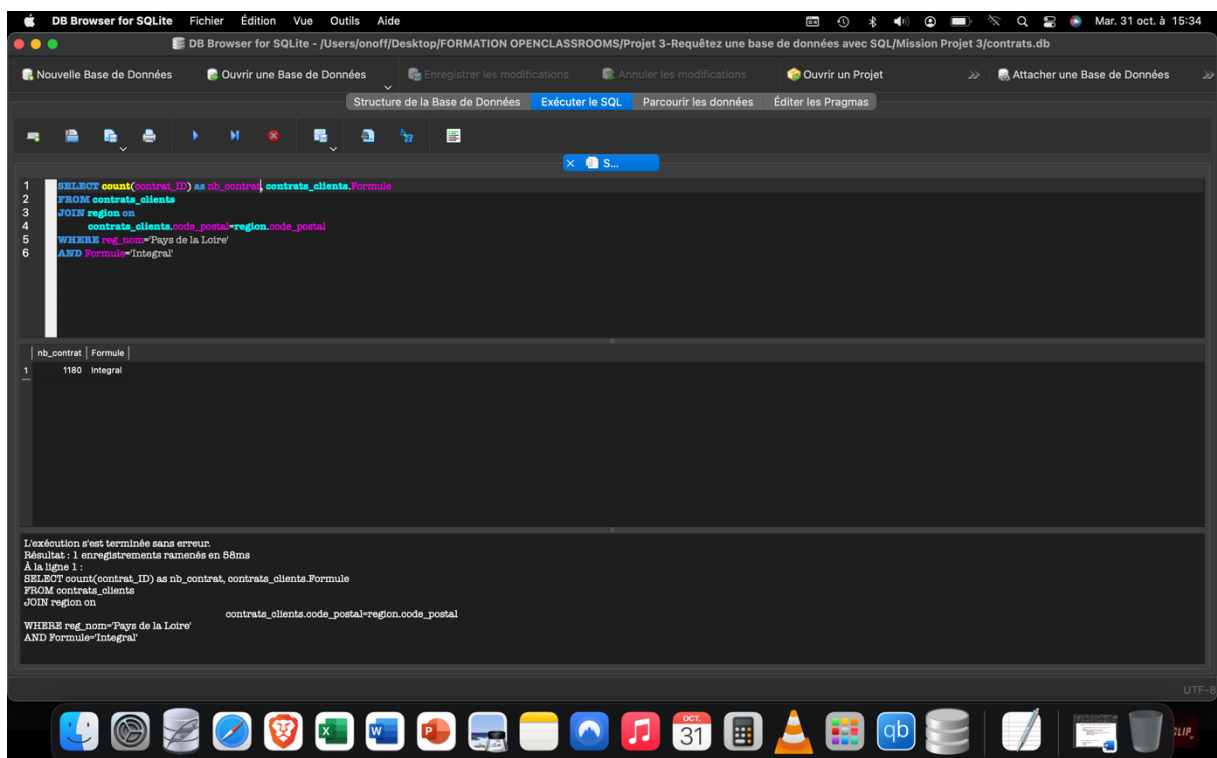
	dep	avg_cotisation
1	75	36.4
2	92	26.27
3	94	19.82
4	78	18.88
5	69	18.46
6	1	18.24
7	6	18.14

The 'Barre d'état' (Status Bar) at the bottom shows the execution history:

- [10:32:59] Requête terminée en 0.021 seconde(s).
- [10:33:16] Requête terminée en 0.022 seconde(s).
- [10:48:13] Erreur pendant l'exécution de la requête sur la base de données « contrats » : near "as": syntax error
- [10:48:16] Erreur pendant l'exécution de la requête sur la base de données « contrats » : near "as": syntax error
- [10:48:38] Requête terminée en 0.022 seconde(s).

10/ Quel est le nombre de contrats avec des formules « Integral » pour la région Pays de la Loire ?

- **Informations recherchées** : nombre de contrats ; formule « Integral » ; Pays de la Loire
- **Clauses** : select ; count() ; from ; left join on ; where ; and
- **Requête** : **Select count(contrat_ID) As nb_contrat, contrats_clients.Formule**
From contrats_clients
Join region **on** contrats_clients.code_postal = region.code_postal
Where reg_nom='Pays de la Loire'
And Formule='Integral'



The screenshot shows the DB Browser for SQLite application. The SQL editor contains the following query:

```
1 SELECT count(contrat_ID) as nb_contrat, contrats_clients.Formule
2 FROM contrats_clients
3 JOIN region on
4     contrats_clients.code_postal=region.code_postal
5 WHERE reg_nom='Pays de la Loire'
6 AND Formule='Integral'
```

The results pane shows a single row:

nb_contrat	Formule
1180	Integral

The status bar at the bottom indicates: "L'exécution s'est terminée sans erreur. Résultat: 1 enregistrements renvoyés en 58ms. À la ligne 1: SELECT count(contrat_ID) as nb_contrat, contrats_clients.Formule FROM contrats_clients JOIN region on contrats_clients.code_postal=region.code_postal WHERE reg_nom='Pays de la Loire' AND Formule='Integral'".

11/ Liste des communes ayant eu au moins 150 contrats ?

- **Informations recherchées** : communes avec au moins 150 contrats
- **Clauses** : select ; count() ; from ; group by ; having ; order by desc
- **Requête** : **Select commune, Count(*)**
From contrats_clients
Group by commune
Having count(*)>=150
Order by count(*) desc

The screenshot shows the SQLiteStudio 3.4.4 interface. The left sidebar displays the database structure with 'contrats_clients' selected. The main editor shows the following SQL query:

```
1 select Commune, count(*)
2 from contrats_clients
3 group by Commune
4 having count(*)>=150
5 order by count(*) desc
6
```

Below the query editor, the results are displayed in a table with 20 rows. The first 7 rows are visible:

	Commune	count(*)
1	PARIS 18	515
2	PARIS 17	468
3	PARIS 15	407
4	PARIS 16	394
5	NICE	387
6	PARIS 11	381
7	PARIS 20	302

The bottom status bar shows the query execution log with several messages, including syntax errors and successful execution times.

12/ Quel est le nombre de contrats pour chaque région ?

- **Informations recherchées** : nombre de contrats pour chaque région
- **Clauses** : select ; count() ; from ; join on ; group by
- **Requête** : **Select** region.reg_nom, **Count**(contrat_ID)
From contrats_clients
Join region **on** contrats_clients.code_postal=region.code_postal
Group by region.reg_nom

