

## Sommaire

### Analyse déjà effectuées dans le modèle :

- 1) Lister les numéros de contrats avec leur surface pour la commune de Caen..... 1
- 2) Lister les numéros de contrats avec le type de contrat et leur formule pour les maisons du département de la Saône et Loire ..... 1
- 3) Lister les nom des régions de France..... 3

### Analyse restante à faire :

- 4) Combien existe-t-il de contrats sur les résidences principales ..... 4
- 5) Quelle est la surface moyenne des logements avec un contrat à Paris ..... 4
- 6) Quels sont les 5 contrats qui ont les surfaces les plus élevées ..... 5
- 7) Quel est le prix moyen de la cotisation mensuelle ..... 6
- 8) Nombre de contrats pour chaque catégorie de prix de la valeur déclarée des biens ..... 6
- 9) Classement des 10 département ou le prix moyen de la cotisation est le plus élevé ..... 7
- 10) Quel est le nombre de contrats avec des formules “intégral” pour la région Pays de la Loire..... 8
- 11) Liste des communes ayant eu au moins 150 contrats..... 9
- 12) Quel est le nombre de contrats pour chaque région ..... 10

## 1) Lister les numéros de contrats et les surface pour la commune de Caen

Requêtes SQL dans SQLiteStudio

```
select contrat_id as Numéro_de_contrat,surface
from contrat
where commune = 'CAEN'
```

### Explication requête :

La requête montre grâce a fonction **SELECT** que l'on va utiliser tout au long des différentes requêtes mais de la fonction **FROM** qui sert renseigner dans quelle table les données sont récupérées.

Les informations que l'on recherche pour cette requête sont les numéros de contrat « Contrat\_ID » et les surface « Surface » concernant la commune de « CAEN », pour filtrer pour la commune qui nous intéresse on utilise la fonction **WHERE** qui permet de mettre une condition qui dans notre cas est *commune = 'CAEN'*

### Résultat obtenu :

Numéro de contrat	surface
103791	35
103792	99
103793	40
103794	20

On peut constater aussi qu'il n'y a que 4 contrats pour la ville de Caen sur plus de 30 000 contrats totaux, on peut aussi

## 2) Lister les contrats, type de contrat, formule pour les maisons du département de la Saône-et-Loire 71

Requetes SQL dans SQLiteStudio

```
select contrat_id as numéro_de_contrat, type_contrat as type_contrat, formule
from contrat
where code_departement = 71
and type_local = 'Maison'
```

### Explication requête :

La requête montre les contrats, le type de contrat et la formule pour les maison du département 71 Saône-et-Loire, les 2 restrictions sont élaborer grâce en premier temps le **WHERE** et en second temps **AND** qui permet de passer une deuxième conditions pour filtrer sur les maisons uniquement.

Résultat obtenu :

numéro de contrat	type contrat	formule
114768	Residence principale	Integral
114782	Residence principale	Classique
114812	Residence principale	Integral
114779	Residence principale	Classique

On peut constater que tous les contrats sont exclusivement des résidences principales ce qui veut dire qu'aucune maison n'est en location pour dans le département de la Saône-et-Loire

### 3) *Lister le nom des régions de France*

*Requetes SQL dans SQLiteStudio*

```
select distinct(reg_nom) as nom_région  
from region |
```

Explication requête ;

Dans cette requête rien de spécial mise à part l'attribut ***DISTINCT***, qui permet de prendre en compte les valeurs différentes d'une sélection dans notre cas cela ce rapporte au nom des régions car si on ne mettaient pas cette attribut la requête nous montrerais les régions de chaque contrats dans quel cas ont auraient des doublons.

Résultat obtenu :

Nom Région
Auvergne-Rhône-Alpes
Hauts-de-France
Provence-Alpes-Côte d'Azur
Grand Est
Occitanie
Normandie
Nouvelle-Aquitaine
Centre-Val de Loire
Corse
Bourgogne-Franche-Comté

On peut donc voir les 19 différentes régions de France ou l'entreprise possède des contrats

#### 4) Combien existe-t-il de contrats sur les résidences principales

Requetes SQL dans SQLiteStudio

```
select count(contrat_id) as nombre_de_contrat
from contrat
where type_contrat = 'Residence principale'
```

##### Explication requête :

La requête montre le nombre de contrats pour les résidences principales et c'est grâce à l'attribut 'Count' qui permet de compter une nombre de lignes d'une sélection, et on précise l'élément que l'on veut compter qui est dans notre cas Contrat\_ID renommé en Numéro de contrat

##### Résultat obtenu :

```
Nombre de contrat
25620
```

On obtient 25 620 contrats concernant des résidence principales, cela montre aussi que la majorité des contrats que possède **l'entreprise** est majoritairement des contrat pour des résidences principales

#### 5) Quelles est la surface moyenne des logements avec un contrat à Paris

Requetes SQL dans SQLiteStudio

```
select avg(surface) as Moyenne_surface
from contrat
where code_departement = 75|
```

##### Explication requête :

La fonction **Avg** signifie average qui permet de faire une moyenne d'une sélection qui dans nôtres cas fait la moyenne des surface pour les contrat de Paris, cependant il est nécessaire de prendre la valeur 75 qui correspond au code du département de Paris car il existe dans le fichier de données plusieurs arrondissement à paris donc aucune commune est égale à « PARIS »

### Résultat obtenu :

Moyenne surface
51.76954988591579

On peut voir donc que la moyenne de la surface pour les contrats de Paris est d'environ **52m<sup>2</sup>**, en faisant la même requête mais pour tous les contrats en enlevant le **WHERE**, on constate que la moyenne des contrats de paris est plus faibles que dans tout les contrats (**58m<sup>2</sup>**) laissant penser que les contrats sur Paris on des surfaces plus faibles.

### **6) Quels sont les 5 contrats qui ont les surfaces les plus élevées**

*Requetes SQL dans SQLiteStudio*

```
select contrat_id as Numéro_de_contrat, surface
from contrat
order by surface desc
limit 5|
```

### Explication requête :

Dans cette requête ce qui change c'est que l'on veut uniquement les 5 contrats avec les surfaces les plus élevées, on va donc utiliser la fonction **ORDER BY** qui permet de classer dans un ordre soit décroissant **DESC** dans nôtres cas les surfaces soit ascendant **ASC**, et la fonction **LIMIT** permet de limiter de limiter cette liste a un certain nombre qui dans notre cas est 5, ce qui nous donne dans le résultat une liste décroissantes des 5 plus grosses surfaces de tous les contrats

### Résultat obtenu :

Numéro de contrat	surface
104211	815
105463	742
130878	595
100822	570
109872	559

On constate que le plus gros contrats est de 815 m<sup>2</sup> ce qui est largement supérieur aux autres car la moyenne de tout les contrats sont de 58m<sup>2</sup> tout comme le 5<sup>ème</sup> qui est de 559m<sup>2</sup>, cela montre aussi que les clients de ces contrats doivent être des entreprise pour leur activité

## 7) Quel est le prix moyen de la cotisation mensuelle

Requetes SQL dans SQLiteStudio

```
select avg(prix_cotisation_mensuel) as Moyenne_prix_cotisation_mensuel  
from contrat
```

### Explication requête :

La requête ci-contre ne change pas des autres car on utilise les même fonction (select,from), et l'attribut pour calculer une moyenne « avg »

### Résultat obtenu :

Moyenne prix cotisation mensuel
19.32869622548212

On constate donc que le prix moyen de la cotisation mensuel est d'environ 19.33 € , pour tous les contrats

## 8) Nombre de contrats pour chaque catégorie de prix de la valeur déclarée des biens

Requetes SQL dans SQLiteStudio

```
select count(contrat_id)as Nombre_de_contrat,Valeur_declaree_biens  
from contrat  
group by Valeur_declaree_biens
```

### Explication requête :

Rien ne change des autres requêtes mise à part le **GROUP BY** qui permet agréger les données c'est à dire les regrouper suivant un filtre, dans cette requête le filtre étant la valeur déclarée des biens

### Résultat obtenu :

Nombre de contrat	Valeur declaree biens
39914	NULL
22720	0-25000
104	100000+
6815	25000-50000
696	50000-100000

Le résultat de cette requête est le suivant on peut voir qu'on a le nombre de contrat suivant la catégorie de valeur déclarée de biens, et avec cela on constate que la majorité des contrats ont des valeurs déclarées entre 0 et 25 000

### **9) Classement des 10 départements où le prix moyen de la cotisation est le plus élevé**

*Requetes SQL dans SQLiteStudio*

```
select dep_nom, avg(prix_cotisation_mensuel) as prix_moyen_cotisation_mensuel
from contrat, region
where contrat.code_postal = region.code_postal
group by dep_nom
order by prix_moyen_cotisation_mensuel desc
limit 10|
```

### Explication requête :

Dans cette requête on sélectionne la moyenne du prix de la cotisation mensuelle des contrats, et des départements, en suite on doit bien lier la table contrat avec la table région, puis on doit grouper par département pour donner la moyenne par département et on doit préciser à la requête d'effectuer un ordre décroissant pour avoir les 10 valeurs les plus élevées en précisant notamment la limite que l'on veut en l'occurrence 10

### Résultat obtenu :

dep_nom	prix_moyen_cotisation_mensuel
Paris	36.40203193033382
Hauts-de-Seine	26.25027685492802
Val-de-Marne	19.82416396979504
Yvelines	18.880187416332
Rhône	18.45681818181818
Ain	18.23888888888889
Alpes-Maritimes	18.13762287756926
Charente-Maritime	17.31730769230769
Haute-Savoie	17.1583850931677
Corse-du-Sud	17.07438016528926

Donc on peut voir le résultat des 10 moyennes du prix des cotisations les plus élevées par départements, on peut aussi constater que Paris a la moyenne la plus élevée là où il y avait la majorité des contrats en Îles de

France, on peut aussi dire que la différence entre le département de Paris et celui de Hauts de Seine et plus conséquent qu'entre le 3ème département et le dernier soir de **19 € à 17€**

## **10 ) Quel est le nombre de contrats avec des formules “intégral” pour la région Pays de la Loire**

*Requetes SQL dans SQLiteStudio*

```
select count(contrat_id) as nombre_contrat
from contrat,region
where contrat.Code_postal = region.code_postal
and formule = 'Integral '
and reg_nom = 'Pays de la Loire'
```

### **Explication requête :**

Dans cette requête on viens juste faire une jointure entre les deux tables *Contrat* et *Region* pour quel les lignes s'associent et qu'on puisse apres fitlrer pour les formules 'integral' et pour la région du Pays de la Loire avec les “AND”.

### **Résultat obtenu :**

nombre contrat
590

On peut donc voir le nombre de contrats pour les formules intégrale pour le Pays de la Loire qui sont aux nombre de 590 contrats

## **11) Liste des communes ayant eu au moins 150 contrats**

*Requetes SQL dans SQLiteStudio*

```
select commune,count(contrat_id) as nombre_de_contrat
from contrat
group by commune
having nombre_de_contrat > 150
```



### Explication requête :

Sur la requête on peut voir que l'on effectue un *count* sur les nombres de contrats avoir le nombre de contrats par département, pour avoir le résultat par département on effectue donc un **GROUP BY** par commune et par la suite on utilise la fonction **HAVING** pour n'afficher que les valeurs au dessus de 150 contrats

### Résultat obtenu :

commune	nombre_de_contrat
BORDEAUX	302
COURBEVOIE	163
GRENOBLE	220
LILLE	161
NANTES	291
NICE	387
PARIS 03	159
PARIS 09	204
PARIS 10	263
PARIS 11	381
PARIS 12	252

Le résultat obtenue de la requête sont les différentes communes pour lequel le nombre de contrats est supérieur a 150, on peut aussi constater que pour la commune de Paris, cette dernière est cinder en plusieurs arrondissement ce qui nous donne comme première valeur de la liste Bordeaux avec 302 contrats

## **12) Quel est le nombre de contrats pour chaque région**

*Requetes SQL dans SQLiteStudio*

```
select count(contrat_id) as nombre_de_contrat, reg_nom as region
from contrat,region
where contrat.Code_postal = region.code_postal
group by reg_nom|
```

### Explication requête :

Dans cette requête on viens juste selectionner le nombre de contats et le nom des région pour reconnaitre sur quel région il y a tel nombre de contrats, ensuite on viens faire la jointure entre les deux tables et pour finir on les groupés par région avec **GROUP BY** .

### Résultat obtenu :

nombre de contrat	region
3056	Auvergne-Rhône-Alpes
287	Bourgogne-Franche-Comté
947	Bretagne
598	Centre-Val de Loire
247	Corse
768	Grand Est
37	Guyane
1182	Hauts-de-France
17	La Réunion
73	Martinique
822	Normandie
2046	Nouvelle-Aquitaine
1601	Occitanie
1196	Pays de la Loire
3279	Provence-Alpes-Côte d'Azur
14179	Île-de-France

On peut donc voir le nombre de contrats par région ce qui nous donne un aperçu d'où l'activité est le plus présent qui est en Îles-de-france ou il y a le plus de contrats, d'ailleurs le nombre de contrats dans cette région est bien supérieur au autre régions nous montrant que l'activité n'est pas si diversifier géographiquement que pourrait l'être, ou une autre raison serait que la région et plus peuplés que les autres ou que les opportunités sont soit meilleur ou plus nombreuse las bas