Requêtes base de données immobilières

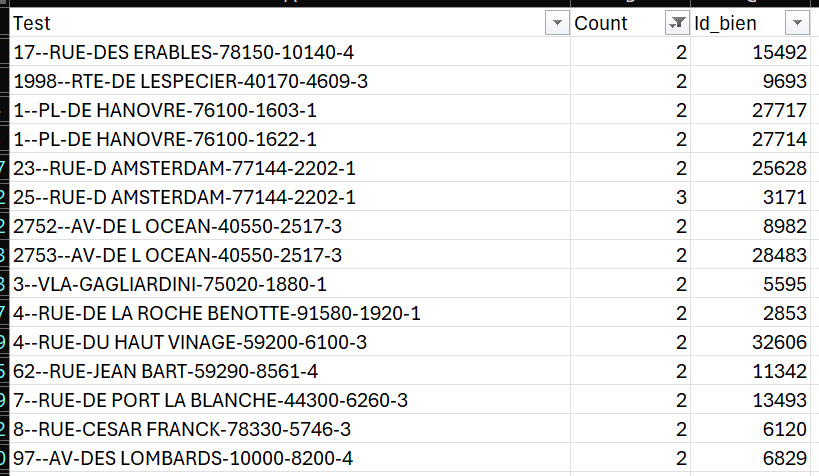
**Remarque sur les doublons parmi les biens** :

en réalité, il manque une donnée « numéro d’appartement » dans les fichiers source, car en effet la plupart des doublons sont des biens de type Appartement, or il n’est pas rare que des appartements sis dans un même bâtiment (et donc à la même adresse) aient exactement les mêmes attributs, à l’exception du numéro d’appartement.

Cela revient à considérer qu’une ligne dans le fichier Biens.csv n’est pas tout à fait un bien immobilier physique, mais un profil de bien. 2 lignes identiques dans ce fichier désignent en réalité un logement similaire et pas forcément identique. On dénombre ainsi 16 lignes apparaissant au moins en double : 1 profil de bien apparaît à 3 reprises, les 14 autres apparaissent à 2 reprises. Parmi ces profils de bien, un seul est une maison, les autres sont des appartements.

Cela définit donc notre marge d’erreur (15) quand on requête notre base de données en espérant y trouver des biens physiques, alors qu’elle ne donne que des profils de bien.

Le champ Test ci-dessous est obtenu en concaténant l’un après l’autre tous les attributs d’un bien.



**Requête 1** : Nombre total d’appartements vendus au 1er semestre 2020

2 versions possibles selon qu’on prend en compte les doublons ou non :

Avec doublons (on considère que chaque ligne de Valeurs-foncières.csv désigne un appartement physique différent, ce qui est invérifiable) :

**SELECT** Type\_local, sum(Nb\_Ventes) **AS** Nombre\_Biens

**FROM** (

**SELECT** \*, count(Bien.Id\_bien) **AS** Nb\_ventes

**FROM** Vente **JOIN** Bien

**ON** Vente.Id\_bien = Bien.Id\_bien

**WHERE** Date\_vente **BETWEEN** '2020-01-01' **AND** '2020-06-30'

**GROUP BY** Bien.Id\_bien

**HAVING** Bien.Type\_local = 'Appartement'

);

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

Sans les doublons (remplacer sum() par count()):

**SELECT** Type\_local, count(Nb\_Ventes) **AS** Nombre\_Biens

**FROM** (

**SELECT** \*, count(Bien.Id\_bien) **AS** Nb\_ventes

**FROM** Vente **JOIN** Bien

**ON** Vente.Id\_bien = Bien.Id\_bien

**WHERE** Date\_vente **BETWEEN** '2020-01-01' **AND** '2020-06-30'

**GROUP BY** Bien.Id\_bien

**HAVING** Bien.Type\_local = 'Appartement'

);

Une image contenant texte, Police, ligne, capture d’écran

Le contenu généré par l’IA peut être incorrect.

**Requête 2** : Nombre de ventes d’appartement par région pour le 1er semestre 2020

**SELECT** Reg\_nom, Type\_local, count(Id\_vente) **AS** Nombre\_ventes

**FROM** Region R **LEFT** **JOIN** (

**SELECT** \*

**FROM** Departement D **JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

**WHERE** Date\_vente **BETWEEN** ‘2020-01-01’ **AND** ‘2020-06-30’

**AND** B.Type\_local = 'Appartement'

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

) **AS** Jcv **ON** D.Cod\_dep = Jcv.Cod\_dep

) **AS** Jdv **ON** R.Cod\_reg = Jdv.Cod\_reg

**GROUP BY** Reg\_nom;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

**Requête 3** : Proportion des ventes d’appartements selon le nombre de pièces

À noter qu’aucune date n’est précisée dans cette requête, n’y apparaît donc aucun filtre de date.

**WITH** NbApp\_byPieces **AS** (

**SELECT** Type\_local, Nb\_pieces,

count(Id\_vente) **AS** Nombre\_Ventes

**FROM** Vente V **JOIN** Bien B

**ON** V.Id\_bien = B.Id\_bien

**WHERE** Type\_local = 'Appartement'

**GROUP BY** Nb\_pieces

),

Total\_Ventes\_App **AS** (

**SELECT** sum(Nombre\_Ventes)

**FROM** NbApp\_byPieces

)

**SELECT** \*,

round(100.0\*Nombre\_Ventes / (**SELECT** \*

**FROM** Total\_Ventes\_App),2) **AS** "Proportion (%)"

**FROM** NbApp\_byPieces;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

**Requête 4** : Liste des 10 départements où le prix du mètre carré est le plus élevé

Sous-entendu mètre carré habitable, c’est donc la surface au sens de la loi Carrez qui va nous intéresser ici, mais faire la distinction entre appartement et maison aurait été pertinent ; la demande suggère de considérer de manière indistincte les biens de type maison et les biens de type appartement, alors qu’il est bien connu que toutes autres choses égales par ailleurs, les biens de type maison sont plus chers au m² (pas de parties communes à respecter).

On pense bien à filtrer les ventes pour lesquelles la valeur foncière n’a pas été renseignée, d’où le filtre : **WHERE** Valeur > 0.

**SELECT** D.Cod\_dep, Dep\_nom,

round(sum(Valeur)/sum(Surface\_Carrez),0) **AS** Prix\_m2

**FROM** Departement D **LEFT JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

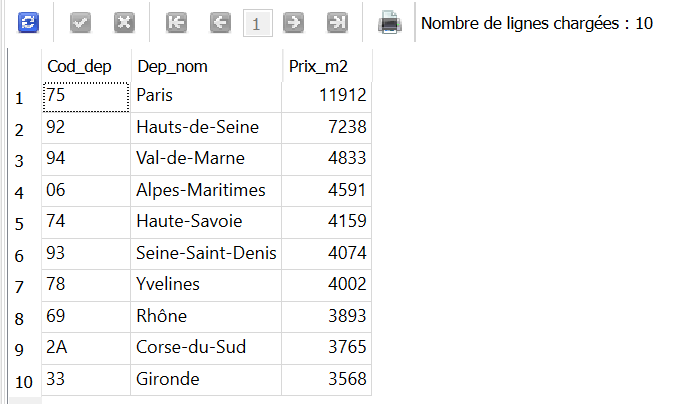
**WHERE** Valeur > 0

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

) **AS** Jcv **ON** D.Cod\_dep = Jcv.Cod\_dep

**GROUP BY** Dep\_nom

**ORDER BY** Prix\_m2 **DESC** **LIMIT** 10;



**Requête 5** : Prix moyen du mètre carré d’une maison en Île-de-France

**SELECT** R.Cod\_reg, Reg\_nom, Type\_local,

round(sum(Valeur)/sum(Surface\_Carrez)) **AS** Prix\_moyen\_m2

**FROM** Region R **LEFT JOIN** (

**SELECT** \*

**FROM** Departement D **JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

**WHERE** V.Valeur > 0

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

) **AS** Jcv **ON** D.Cod\_dep = Jcv.Cod\_dep

) **AS** Jdv **ON** R.Cod\_reg = Jdv.Cod\_reg

**GROUP BY** R.Cod\_reg, Type\_local

**HAVING** R.Cod\_reg = 11 **AND** Type\_local = 'Maison';

Une image contenant texte, Police, capture d’écran, ligne

Le contenu généré par l’IA peut être incorrect.

**Requête 6** : Liste des 10 appartements les plus chers avec la région et le nombre

de mètres carrés.

**SELECT** V.Id\_bien, Type\_local, Valeur, Reg\_nom, Lib\_com, Surface\_Carrez

**FROM** Vente V **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Departement D **JOIN** Region R

**ON** D.Cod\_reg = R.Cod\_reg

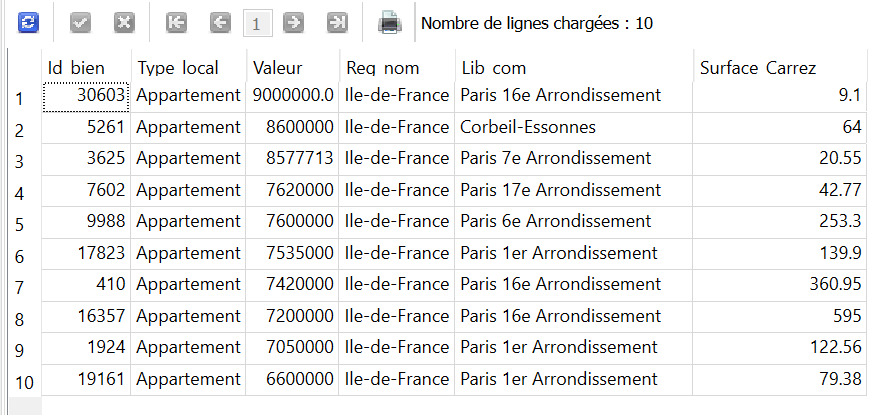
) **AS** Jdr **ON** C.Cod\_dep = Jdr.Cod\_dep

) **AS** Jcr **ON** B.Cod\_depcom = Jcr.Cod\_depcom

) **AS** Jbr **ON** V.Id\_bien = Jbr.Id\_bien

**WHERE** Valeur > 0

**ORDER BY** Valeur **DESC LIMIT** 10;



Certains résultats sont ahurissants, en réalité les données ont subi de mauvais traitements, si bien que certaines lignes importantes ont disparu des fichiers de données source. Par exemple, 9 millions d’euros pour un appartement de 9,1 m², c’est absurde : la transaction réelle associée concerne un ensemble situé à la même adresse avec comme disposition principale la vente d’un appartement de 433.7 m². De même, 8,6 millions d’euros pour un appartement de 64 m² à Corbeil-Essonnes, c’est absurde. La transaction réelle correspondante concerne tout un lot immobilier de quelques immeubles avec une vingtaine d’appartements chacun.

**Requête 7** : Taux d’évolution du nombre de ventes entre le premier et le second trimestre de 2020

**WITH** Nb\_ventes\_T1 **AS** (

**SELECT** count(\*)

**FROM** Vente

**WHERE** Date\_vente **BETWEEN** '2020-01-01' **AND** '2020-03-31'

),

Nb\_ventes\_T2 **AS** (

**SELECT** count(\*)

**FROM** Vente

**WHERE** Date\_vente **BETWEEN** '2020-04-01' **AND** '2020-06-30'

)

**SELECT** round(100.0 \* ((**SELECT** \*

**FROM** Nb\_ventes\_T2)

/ (1.0 \* (**SELECT** \*

**FROM** Nb\_ventes\_T1)) - 1.0),2)

**AS** "Evolution\_Nombre\_Ventes\_2020\_T1-T2 (%)";

Une image contenant texte, capture d’écran, Police, ligne

Le contenu généré par l’IA peut être incorrect.

**Requête 8** : Le classement des régions par rapport au prix au mètre carré des appartements de plus de 4 pièces

**SELECT** Reg\_nom,

round(sum(Valeur)/sum(Surface\_Carrez)) **AS** Prix\_m2

**FROM** Region R **LEFT JOIN** (

**SELECT** \*

**FROM** Departement D **JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

**WHERE** Type\_local = 'Appartement'

**AND** Nb\_pieces > 4

**AND** Valeur > 0

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

) **AS** Jcv **ON** D.Cod\_dep = Jcv.Cod\_dep

) **AS** Jdv **ON** R.Cod\_reg = JDv.Cod\_reg

**GROUP BY** Reg\_nom

**ORDER BY** Prix\_m2 **DESC**;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

**Requête 9** : Liste des communes ayant eu au moins 50 ventes au 1er trimestre

**SELECT** C.Cod\_depcom, Lib\_com,

count(Id\_vente) **AS** Nombre\_Ventes\_T1

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

**WHERE** V.Date\_vente **BETWEEN** '2020-01-01' **AND** '2020-03-31'

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

**GROUP BY** C.Cod\_depcom

**HAVING** Nombre\_Ventes\_T1 >= 50

**ORDER BY** Nombre\_Ventes\_T1 **DESC**;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

**Requête 10** : Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces

On crée la vue des prix moyens au m² des appartements selon leur nombre de pièces :

**WITH** Prix\_m2\_byNbPieces\_App **AS** (

**SELECT** Nb\_pieces, Type\_local,

round(sum(Valeur)/sum(Surface\_Carrez)) **AS** Prix\_m2

**FROM** Vente V **JOIN** Bien B

**ON** B.Id\_bien = V.Id\_bien

**WHERE** Type\_local = 'Appartement'

**AND** Valeur > 0

**GROUP BY** Nb\_pieces

)

**SELECT** round(100.0 \* (

**(SELECT** Prix\_m2

**FROM** Prix\_m2\_byNbPieces\_App

**WHERE** Nb\_pieces = 3)

/ (**SELECT** Prix\_m2

**FROM** Prix\_m2\_byNbPieces\_App

**WHERE** Nb\_pieces = 2)

- 1.0 ),2)

**AS** "Diff\_Prix\_m2\_T2-T3 (%)";

Une image contenant texte, capture d’écran, Police, conception

Le contenu généré par l’IA peut être incorrect.

**Requête 11** : Les moyennes de valeurs foncières pour le top 3 des communes de chacun des départements 6, 13, 33, 59 et 69

**WITH** Valeurs\_moyennes\_Com **AS** (

**SELECT** Cod\_depcom, Lib\_com, D.Cod\_dep, Dep\_nom, Valeur\_moyenne

**FROM** Departement D **JOIN** (

**SELECT** \*

**FROM** Commune C **JOIN** (

**SELECT** \*, round(avg(Valeur)) **AS** Valeur\_moyenne

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

**WHERE** Valeur > 0

**GROUP BY** B.Cod\_depcom

) **AS** Jbv **ON** C.Cod\_depcom = Jbv.Cod\_depcom

) **AS** Jcv **ON** D.Cod\_dep = Jcv.Cod\_dep

)

**SELECT** \*

**FROM** (**SELECT** \*, row\_number() **OVER** (

**PARTITION BY** Cod\_dep **ORDER BY** Valeur\_moyenne **DESC**) Rang\_dep

**FROM** Valeurs\_moyennes\_Com

**WHERE** Cod\_dep **IN** ('06','13','33','59','69')

) **WHERE** Rang\_dep <= 3;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

**Requête 12** : Les 20 communes avec le plus de transactions pour 1000 habitants pour les communes qui dépassent les 10 000 habitants.

**SELECT** Lib\_com, C.Cod\_depcom, Pop\_tot,

round(1000.0\*count(Id\_vente)/Pop\_tot,2) **AS** Nb\_Ventes\_1000hab

**FROM** Commune C **JOIN** (

**SELECT** \*

**FROM** Bien B **JOIN** Vente V

**ON** B.Id\_bien = V.Id\_bien

) **AS** Jbv **ON** C.cod\_depcom = Jbv.Cod\_depcom

**WHERE** Pop\_tot > 10000

**GROUP BY** C.Cod\_depcom

**ORDER BY** Nb\_Ventes\_1000hab **DESC** **LIMIT** 20;

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.