

OPENCLASSROOMS



Parcours Data Analyst

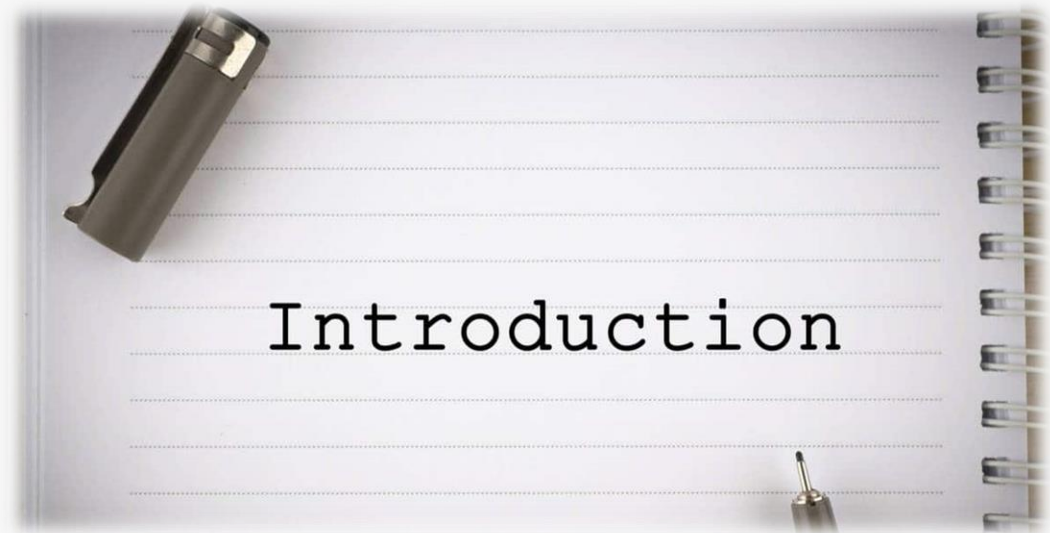
Projet 3



Projet DataImmo

Donia NEFIZ

<b>1</b>	<b>Introduction .....</b>	<b>03</b>
<b>2</b>	<b>Modélisation de la Base de Données DataImmo .....</b>	<b>04</b>
<b>3</b>	<b>Implémentation de la Base de Données DataImmo .....</b>	<b>08</b>
<b>4</b>	<b>Requêtes &amp; résultats .....</b>	<b>12</b>
<b>5</b>	<b>Bilan .....</b>	<b>22</b>



**Laplace Immo** est un réseau national  
d'agences immobilières

**Poste occupé** : Data Analyst

## PROJET DATAIMMO

La création d'un modèle à partir du data set  
open source "Demandes de valeurs foncières" publié par  
la Direction Générale des Finances Publiques.

**Objectif du projet** : soutenir les agents dans leur travail  
de conseiller clientèle en transaction immobilière

**Source des données** : [data.gouv.fr](https://data.gouv.fr)

**Données traitées** : premier semestre 2020 (échantillon  
pilote)



## Modélisation de la Base de données

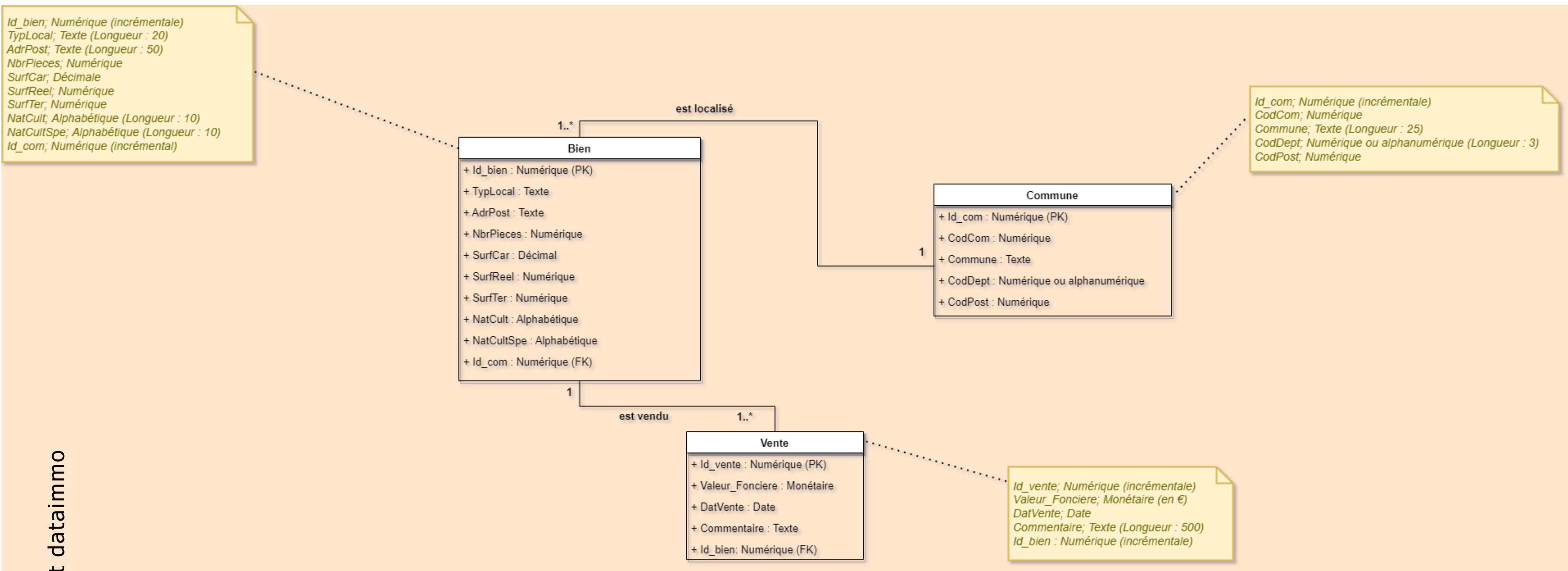
1. Dictionnaire des données
2. Modèle conceptuel de données  
(format diagramme de classes UML)
3. Schéma relationnel normalisé  
(sous SQL Power Architect)

# 1. DICTIONNAIRE DES DONNÉES \*

Numéro	Code propriété	Signification	Type	Observation
1	Id_bien	ID Bien	Numérique	PK (incrémentale)
2	Id_com	ID Commune	Numérique	PK (incrémentale)
3	Id_vente	ID Vente	Numérique	PK (incrémentale)
4	TypLocal	Type Local	Texte	Longueur : 20
6	NbrPieces	Nombre de pièces	Numérique	
7	SurfCar	Surface carré	Décimale	
8	SurfReel	Surface réelle	Numérique	
9	SurfTer	Surface terrain	Numérique	
10	NatCult	Nature culture	Texte	Longueur : 10
11	NatCultSpe	Nature culture spéciale	Texte	Longueur : 10
12	CodDept	Code Département	Alphanumérique	Longueur : 3
13	AdrPost	Adresse postale	Texte	Longueur : 50
14	CodPost	Code Postale	Numérique	
15	CodCom	Code Commune	Numérique	
16	Commune	Commune	Texte	Longueur : 25
17	Valeur_Fonciere	Valeur foncière	Monétaire	
18	DatVente	Date de la vente	Date	
19	Commentaire	Commentaire	Texte	Longueur : 500

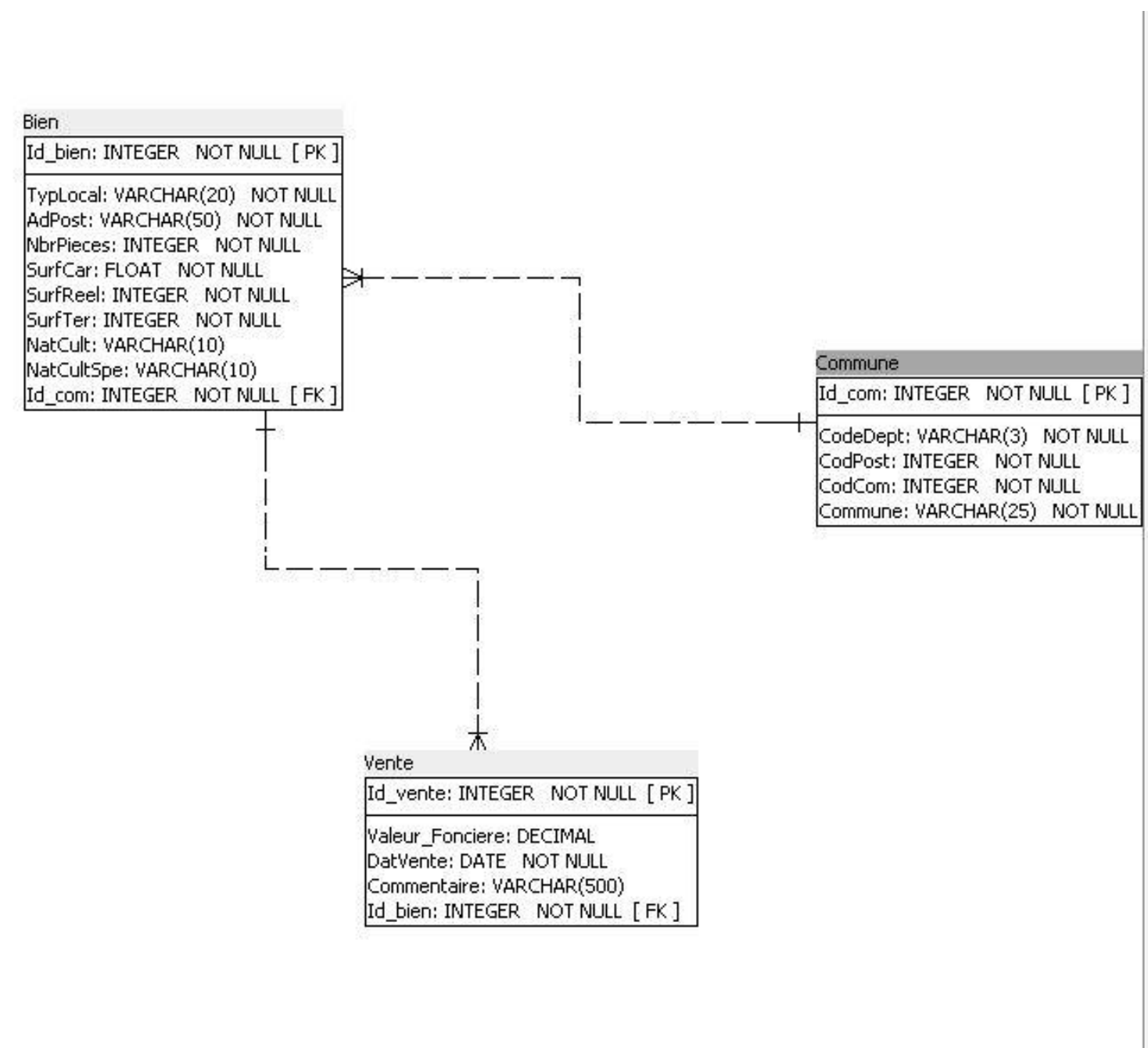
\* Les données ont été nettoyées et sélectionnées pour répondre aux besoins du projet

## 2. MODÈLE CONCEPTUEL DES DONNÉES \*



\* Diagramme de classes conçu avec app.diagrams.net (anciennement Draw.io)

### 3. SCHÉMA RELATIONNEL NORMALISÉ \*



\* Schéma relationnel conçu avec SQL Power Architect



## Implémentation de la base de données

1. Préparation des données avant import sur le SGBD MySQL
2. Création des tables sur MySQL et import des données



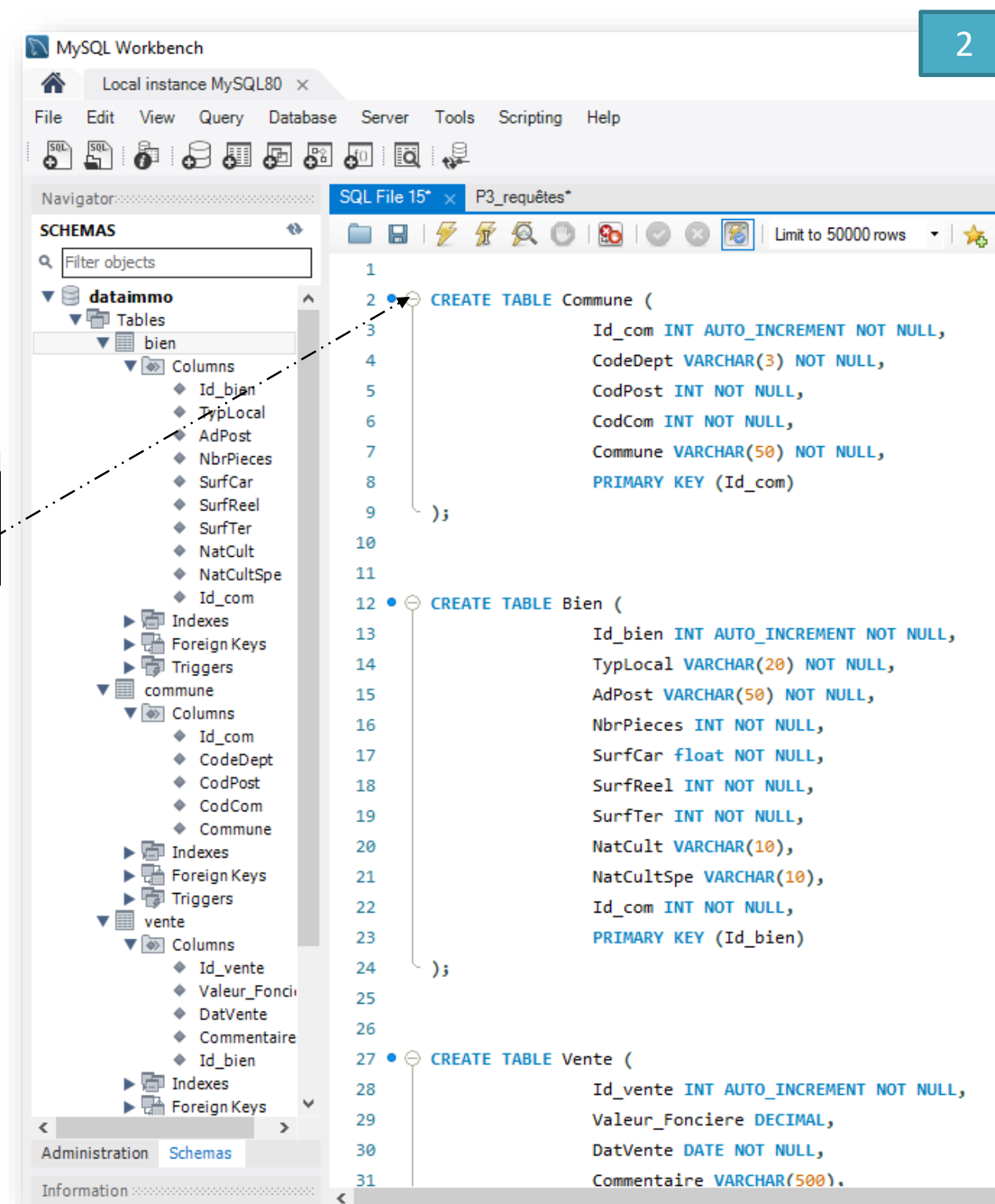
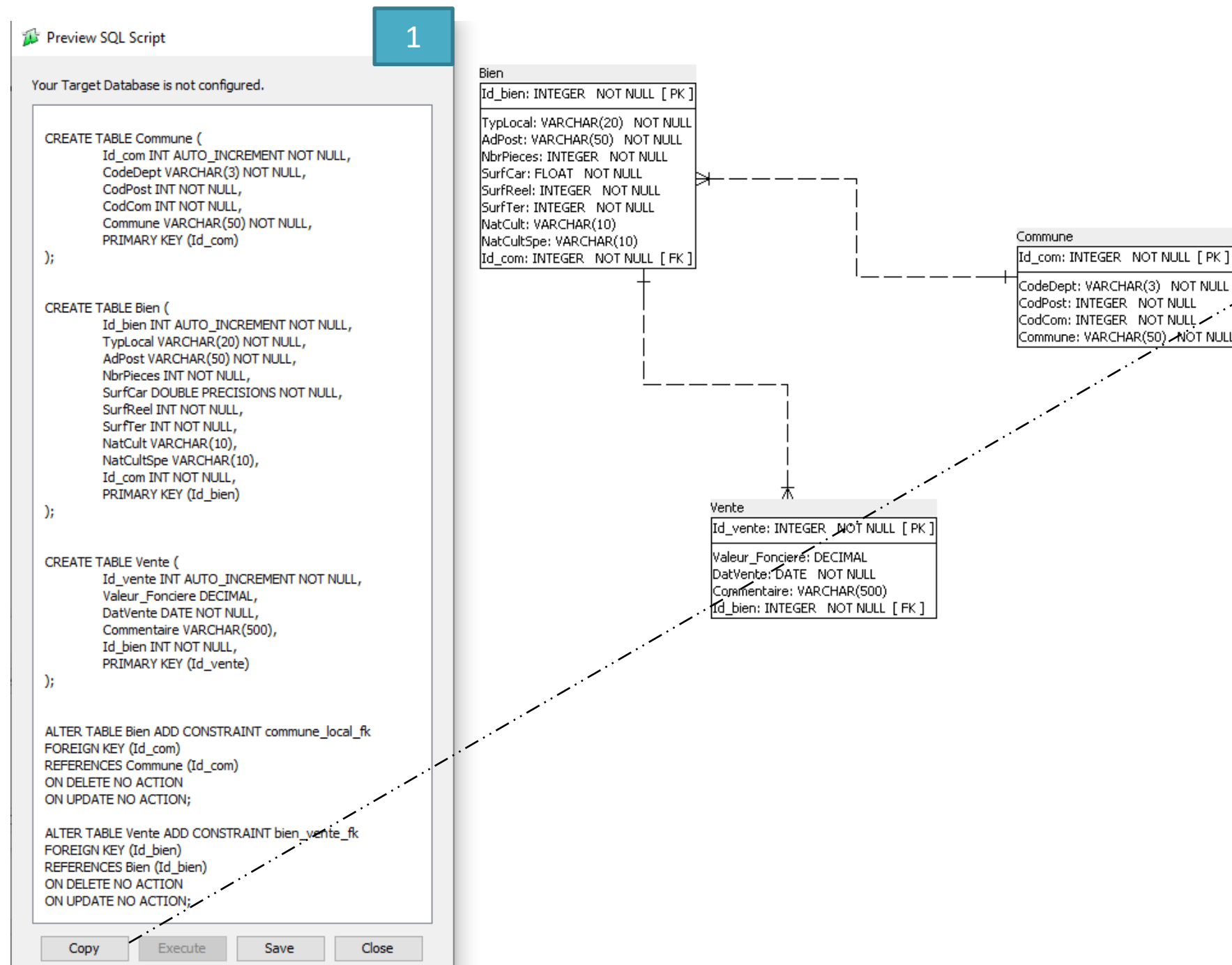
# 1. PRÉPARATION DES DONNÉES

Liaison des tables et création des clés étrangères (FK) à l'aide de Power Query sur notre base de données Excel pour répondre au schéma relationnel DataImmo :

1. Création des tables
2. Gestion des doublons sur les tables
3. Fusion des requêtes et ajout de colonnes d'index pour création des FK
4. Nettoyage de données
5. Chargement des données
6. Préparation des attributs (types et libellés) et complétion des données manquantes sur les colonnes NOT NULL
7. Exportation des tables au format CSV >> prêt pour import dans MySQL

# 2. CRÉATION DES TABLES SUR MySQL ET IMPORT DES DONNÉES

Projet dataimmo

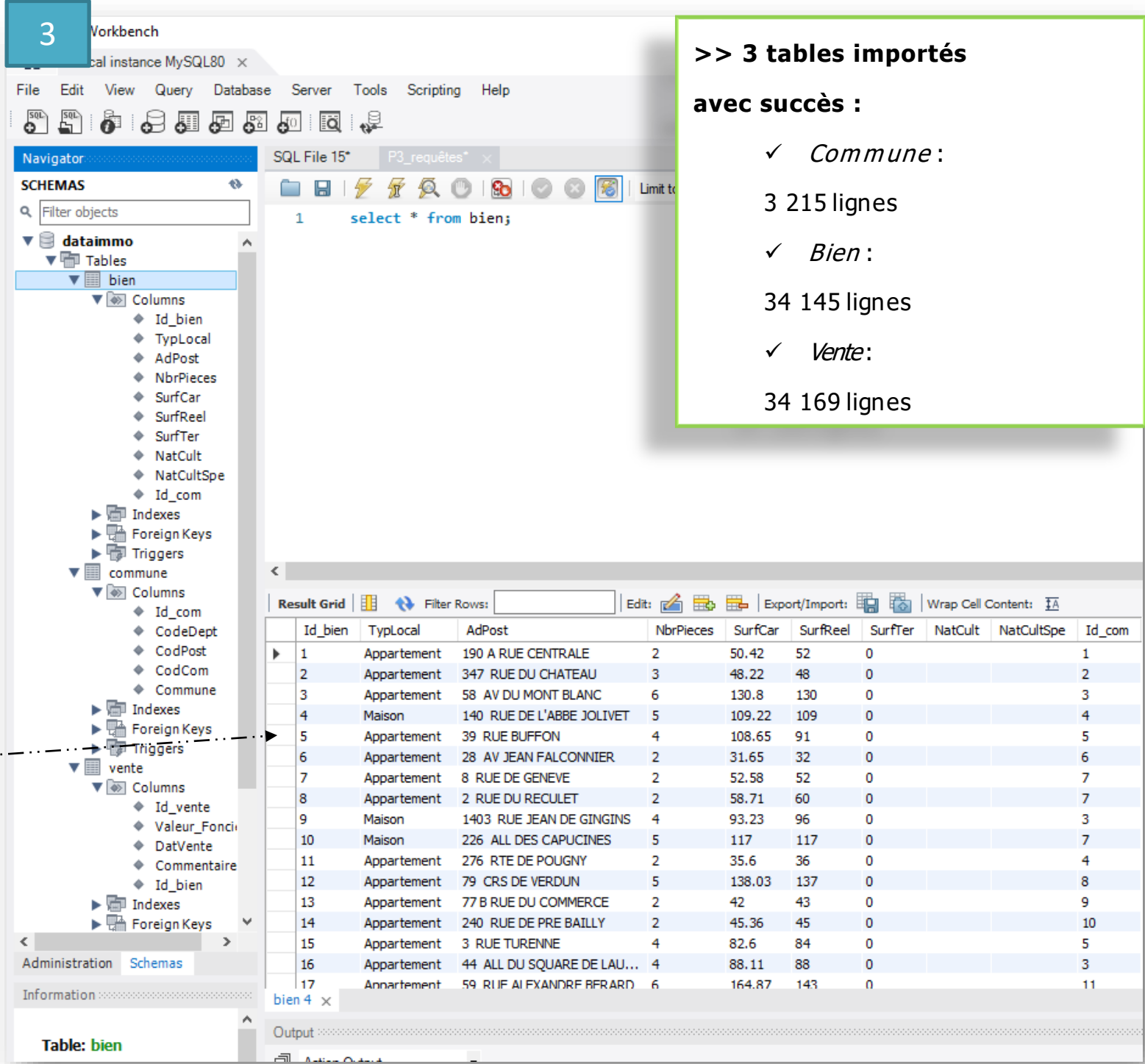
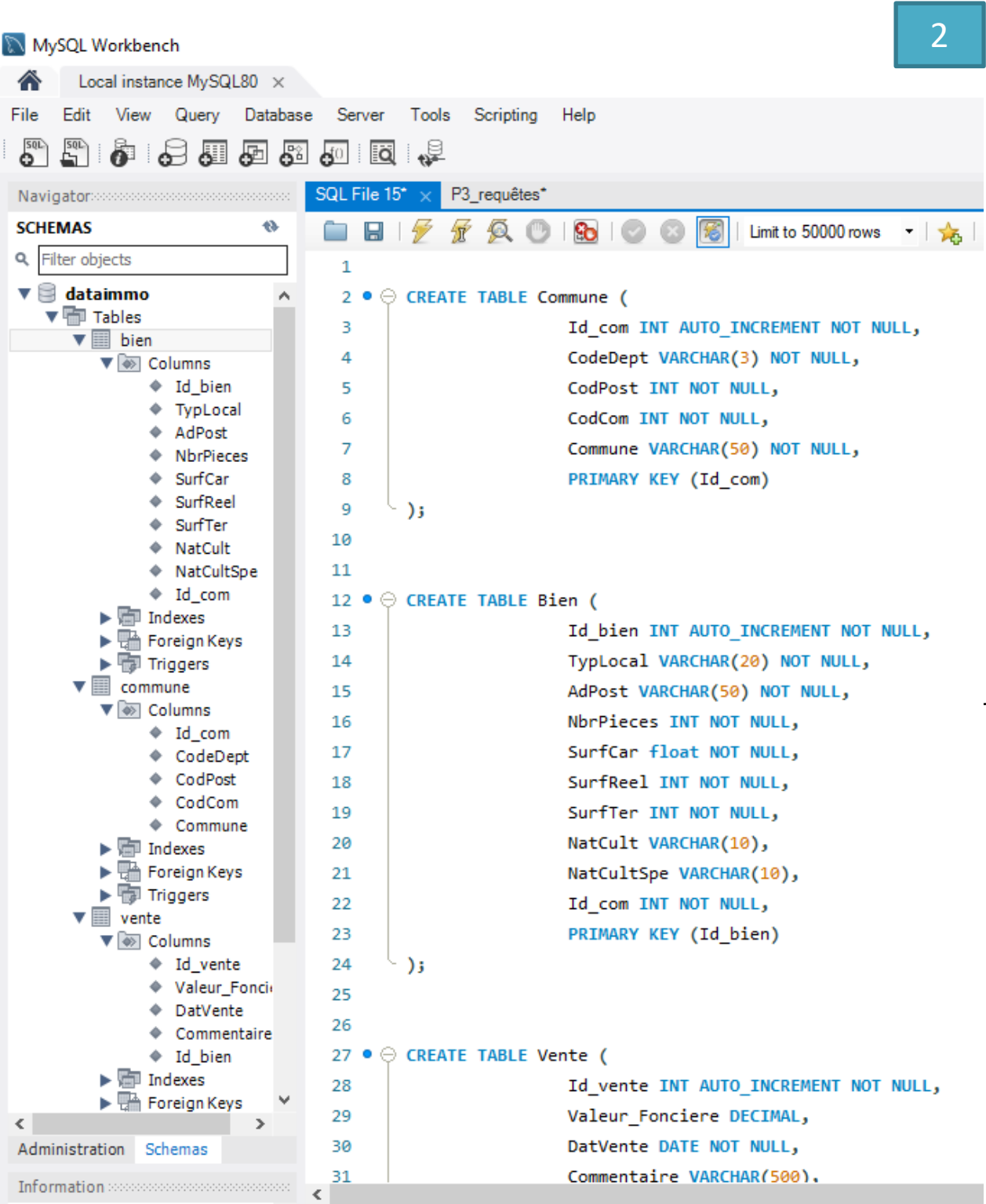


... et exécuté sur MySQL Workbench

Script généré sur Power Architect...

## 2. CRÉATION DES TABLES SUR MySQL ET IMPORT DES DONNÉES

Résultat :













REQUÊTES

&

RÉSULTATS

## 1. Nombre total d'appartements vendus au 1er semestre 2020 :

```
1  /*
2      1. Je compte le nombre de ventes en appliquant un COUNT sur l'Id_vente
3      2. Je joins la table vente à la table bien
4      3. Je filtre sur les appartements en appliquant un WHERE sur le TypLocal
5      4. J'ajoute un filtre sur le 1er semestre 2020 avec un AND DatVente BETWEEN "AAAA-MM-JJ" AND "AAAA-MM-JJ"
6  */
7
8  ●  SELECT COUNT(Id_vente) AS "Total des appartements vendus au 1er semestre 2020"
9      FROM vente
10     JOIN bien
11     ON vente.Id_bien=bien.Id_bien
12     WHERE TypLocal="Appartement"
13     AND DatVente BETWEEN "2020-01-01" and "2020-06-30";
```

Result Grid     Filter Rows: <input type="text"/>   Export:    Wrap Cell Content: 	
	Total des appartements vendus au 1er semestre 2020
31377	
31377	
semestre 2020	Total des appartements vendus au 1er
Result Grid     Filter Rows: <input type="text"/>   Export:    Wrap Cell Content: 	

## 2. Proportion des ventes d'appartement par le nombre de pièces :

```

1  /*
2      1. Je calcule la proportion des ventes
3      2. Je joins la table vente à la table bien
4      3. Je filtre sur les appartements en appliquant un WHERE sur le TypLocal
5      4. Je regroupe les valeurs par le nombre de pièces en appliquant un GROUP BY sur le NbrPieces
6      5. Je formate l'affichage des nombres décimaux en appliquant un ROUND
7  */
8
9  ●  SELECT (Nbrpieces),
10     ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM vente),2) AS "Proportion des ventes d'appartements par le nombre de pièces"
11     FROM vente
12     JOIN bien
13     ON vente.Id_bien=bien.Id_bien
14     WHERE TypLocal="Appartement"
15     GROUP BY NbrPieces;

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	Nbrpieces	Proportion des ventes d'appartements par le nombre de pièces			
0	0.09				
1	19.72				
2	28.63				
3	26.24				
4	13.05				
5	3.26				
6	0.60				
7	0.16				
8	0.05				
9	0.02				
10	0.01				
11	0.00				
12	0.00				
13	0.00				
14	0.00				

### 3. Liste des 10 départements où le prix du mètre carré est le plus élevé :

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

/\*

1. Je calcule le prix moyen du m² à l'aide de la colonne Valeur\_Fonciere et SurfReel...

2. ... à l'aide d'une jointure entre les tables commune, bien et vente

3. Je regroupe les valeurs par département en appliquant un GROUP BY sur le CodeDept

4. Je trie les valeurs par ordre décroissant en appliquant un ORDER BY sur le prix moyen du m² DESC

5. Je limite le nombre de valeurs à 10 lignes en appliquant un LIMIT 10

\*/

•

SELECT CodeDept AS "Liste des 10 départements où le prix du m² est le plus élevé",

(SUM(Valeur\_Fonciere) / SUM(SurfReel)) AS "Prix moyen du m²"

FROM commune

JOIN bien

ON commune.Id\_com=bien.Id\_com

JOIN vente

ON bien.Id\_bien=vente.Id\_bien

GROUP BY CodeDept

ORDER BY SUM(Valeur\_Fonciere) / SUM(SurfReel) DESC

LIMIT 10;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Fetch rows:

	Liste des 10 départements où le prix du m² est le plus élevé	Prix moyen du m²
▶	75	11969.9186
	92	7359.1114
	94	5036.8673
	6	4607.9430
	93	4120.5693
	74	4046.1626
	78	4042.6963
	69	3900.5257
	2A	3755.9116
	33	3635.3190
	33	3632.3180
	54	3122.8119
	68	3000.2521



#### 4. Prix moyen du mètre carré d'une maison en Île-de-France :

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

/\*

1. Je calcule le prix moyen du m² en appliquant un AVG sur la Valeur\_Fonciere divisé par la SurfReel...

2. ... à l'aide d'une jointure entre les tables commune, bien et vente

3. Je filtre sur les maisons en appliquant un WHERE sur le TypLocal

4. J'ajoute un filtre avec un AND sur les CodeDept 75, 77, 78, 91, 92, 93, 94, 95

5. Je formate l'affichage du résultat en lui appliquant un ROUND

\*/

•

SELECT ROUND(AVG(Valeur\_Fonciere/SurfReel),2) AS "Prix moyen du mètre carré d'une maison en Île-de-France"

FROM vente

JOIN bien

ON vente.Id\_bien=bien.Id\_bien

JOIN commune

ON commune.Id\_com=bien.Id\_com

WHERE TypLocal="Maison"

AND CodeDept IN (75, 77, 78, 91, 92, 93, 94, 95);

<

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Prix moyen du mètre carré d'une maison en Île-de-France

3997.71

3997.71

Île-de-France

Prix moyen du mètre carré d'une maison en

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

<



5. Liste des 10 appartements les plus chers  
avec le département et le nombre de mètres carrés :

```
1  /*
2  1. Je sélectionne l'Id_bien, la SurfReel, le CodeDept et la Valeur_Fonciere...
3  2. 2. ... à l'aide d'une jointure entre les tables commune, bien et vente
4  3. Je filtre sur les appartements en appliquant un WHERE sur le typLocal
5  4. Je trie les valeurs par ordre décroissant en appliquant un ORDER BY sur la Valeur_Fonciere
6  5. Je limite le nombre de valeurs à 10 lignes en appliquant un LIMIT 10
7  */
8
9  • SELECT bien.Id_bien AS "Liste des 10 appartements les plus chers (1er semestre 2020)",
10     SurfReel AS "Nombre de m2",
11     CodeDept AS "Département",
12     Valeur_Fonciere AS "Prix"
13 FROM bien
14 JOIN commune
15 ON commune.Id_com=bien.Id_com
16 JOIN vente
17 ON bien.Id_bien=vente.Id_bien
18 WHERE TypLocal="Appartement"
19 ORDER BY Valeur_Fonciere DESC
20 LIMIT 10;
```

	Liste des 10 appartements les plus chers (1er semestre 2020)	Nombre de m2	Département	Prix
▶	32251	10	75	9000000
	21816	62	91	8600000
	29777	289	75	8577713
	32409	42	75	7620000
	29828	200	75	7600000
	29500	143	75	7535000
	31949	357	75	7420000
	32111	241	75	7200000
	29331	310	75	7050000
	29491	76	75	6600000
	32401	12	12	6600000
	32331	370	12	1020000

## 6. Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020 :

```
1  /*
2  1. Je crée deux tables éphémères avec WITH :
3      a. Je calcule le nombre de ventes au 1er trimestre 2020 en appliquant un COUNT sur l'Id_vente
4          et en filtrant sur le 1er trimestre 2020 avec un WHERE DatVente BETWEEN "AAAA-MM-JJ" AND "AAAA-MM-JJ"
5      b. Je calcule le nombre de ventes au 2ème trimestre 2020 en appliquant un COUNT sur l'Id_vente
6          et en filtrant sur le 2ème trimestre 2020 avec un WHERE DatVente BETWEEN "AAAA-MM-JJ" AND "AAAA-MM-JJ"
7  2. Je calcule le taux d'évolution du nombre de ventes entre le 1er et le 2ème trimestre 2020
8  3. Je formate le résultat final en lui appliquant un ROUND
9  */
10
11 • WITH
12
13 Ventes_1er_trim AS (
14     SELECT
15         COUNT(Id_vente) AS "Nbre_ventes_1er_trim"
16     FROM dataimmo.vente
17     WHERE DatVente BETWEEN "2020-01-01" AND "2020-03-31"
18 ),
19 Ventes_2eme_trim AS (
20     SELECT
21         COUNT(Id_vente) AS "Nbre_ventes_2eme_trim"
22     FROM dataimmo.vente
23     WHERE DatVente BETWEEN "2020-04-01" AND "2020-06-30"
24 )
25 SELECT
26     ROUND((((Nbre_ventes_2eme_trim - Nbre_ventes_1er_trim)/Nbre_ventes_2eme_trim)*100),2) AS "Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020 (%)"
27 FROM Ventes_2eme_trim,
28 Ventes_1er_trim;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020 (%)
3.54
3.54
Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020 (%)

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

7. Liste des communes où le nombre de ventes a augmenté d'au moins 20% entre le 1er et le 2nd trimestre de 2020 :

```
1  /*
2  1. Je crée deux tables éphémères avec WITH :
3      a. Je calcule le nombre de ventes par commune au 1er trimestre 2020 en appliquant un COUNT sur l'Id_vente,
4         je filtre sur le 1er trimestre 2020 avec un WHERE DatVente BETWEEN "AAAA-MM-JJ" AND "AAAA-MM-JJ"
5         et je regroupe les valeurs par Id_com avec un GROUP BY Id_com
6      b. Je calcule le nombre de ventes par commune au 2ème trimestre 2020 en appliquant un COUNT sur l'Id_vente,
7         je filtre sur le 2ème trimestre 2020 avec un WHERE DatVente BETWEEN "AAAA-MM-JJ" AND "AAAA-MM-JJ"
8         et je regroupe les valeurs par Id_com avec un GROUP BY Id_com
9  2. Je calcule le taux d'évolution du nombre de ventes entre le 1er et le 2ème trimestre 2020
10 3. Je filtre sur les communes ayant un taux supérieur ou égal à 20%
11 4. Je regroupe les valeurs par Commune
12 5. Et je formate le résultat final en lui appliquant un ROUND
13 */
```

	Commune	Taux_evol_1er_2eme_trim_2020
►	TOULON	20.00
	PARIS 08	20.78
	SAINT-BRIEUC	20.93
	RUEIL-MALMAISON	20.93
	NEUILLY-SUR-SEINE	21.05
	PARIS 14	21.05
	PAU	21.79
	LYON 3EME	21.79
	AGDE	24.44
	LORIENT	24.44
	LE MANS	24.44
	LE PERREUX-SUR-MARNE	24.44
	PARIS 01	24.44

1400 row(s) returned

```
17  Ventes_1er_trim AS (
18      SELECT
19          commune.Id_com,
20          Commune,
21          COUNT(Id_vente) AS "Nbre_ventes_1er_trim"
22      FROM vente
23      JOIN bien
24          ON vente.Id_bien=bien.Id_bien
25      JOIN commune
26          ON bien.Id_com=commune.Id_com
27      WHERE DatVente BETWEEN "2020-01-01" AND "2020-03-31"
28      GROUP BY Id_com
29  ),
30  Ventes_2eme_trim AS (
31      SELECT
32          commune.Id_com,
33          Commune,
34          COUNT(Id_vente) AS "Nbre_ventes_2eme_trim"
35      FROM vente
36      JOIN bien
37          ON vente.Id_bien=bien.Id_bien
38      JOIN commune
39          ON bien.Id_com=commune.Id_com
40      WHERE DatVente BETWEEN "2020-04-01" AND "2020-06-30"
41      GROUP BY Id_com
42  )
43  SELECT
44      commune.Commune,
45      ROUND((((Nbre_ventes_2eme_trim - Nbre_ventes_1er_trim)/Nbre_ventes_2eme_trim)*100,2) AS "Taux_evol_1er_2eme_trim_2020"
46  FROM Ventes_1er_trim,
47       Ventes_2eme_trim
48  JOIN commune
49      ON Ventes_2eme_trim.Id_com=commune.Id_com
50  WHERE ((Nbre_ventes_2eme_trim - Nbre_ventes_1er_trim)/Nbre_ventes_2eme_trim)*100 >= 20
51  GROUP BY Commune;
52
53  SELECT
54      commune.Commune,
55      ROUND((((Nbre_ventes_2eme_trim - Nbre_ventes_1er_trim)/Nbre_ventes_2eme_trim)*100,2) AS "Taux_evol_1er_2eme_trim_2020"
56  FROM Ventes_1er_trim,
57       Ventes_2eme_trim
58  JOIN commune
59      ON Ventes_2eme_trim.Id_com=commune.Id_com
60  WHERE ((Nbre_ventes_2eme_trim - Nbre_ventes_1er_trim)/Nbre_ventes_2eme_trim)*100 >= 20
61  GROUP BY Commune;
```

### 8. Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces :

```
1  /*
2  1. Je crée deux tables éphémères avec WITH :
3      a. Je calcule le prix du m² pour un appartement 2 pièces en appliquant un AVG de la Valeur_Foncière divisé par la SurfReel,
4          je filtre sur les appartements avec un WHERE sur le TypLocal AND Nbrpieces = 2
5      b. Je calcule le prix du m² pour un appartement 3 pièces en appliquant un AVG de la Valeur_Foncière divisé par la SurfReel,
6          je filtre sur les appartements avec un WHERE sur le TypLocal AND Nbrpieces = 3
7  2. Je calcule la différence en pourcentage du prix du m² pour un appartement de 2 pièces et un appartement de 3 pièces
8  3. Et je formate le résultat final en lui appliquant un ROUND
9  */
```

	Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces
▶	-13.03

```
1  WITH
2
3  Prix_m2_2p AS(
4      SELECT AVG((Valeur_Fonciere) / (SurfReel)) AS "2p"
5          FROM vente
6          JOIN bien
7              ON vente.Id_bien=bien.Id_bien
8          JOIN commune
9              ON commune.Id_com=bien.Id_com
10         WHERE TypLocal="Appartement"
11         AND NbrPieces=2
12     ),
13
14     Prix_m2_3p AS(
15         SELECT AVG((Valeur_Fonciere) / (SurfReel)) AS "3p"
16             FROM vente
17             JOIN bien
18                 ON vente.Id_bien=bien.Id_bien
19             JOIN commune
20                 ON commune.Id_com=bien.Id_com
21             WHERE TypLocal="Appartement"
22             AND NbrPieces=3
23     )
24     SELECT
25         ROUND(((3p-2p)/2p)*100,2) AS "Différence en pourcentage du prix
26         au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces"
27     FROM Prix_m2_3p,
28         Prix_m2_2p
```

Result Grid

Filter Rows:

Exports:

Wrap Cell Content:

Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces
-13.03
-13.03



9. Les moyennes de valeurs foncières pour le top 3 des communes des départements 6, 13, 33, 59 et 69 :

```
1  /*
2  1. Je crée une table éphémère avec WITH :
3      a. Je calcule la moyenne des Valeur_Foncière par Commune et CodeDept
4      b. Je réalise un top 3 des communes ayant les Valeurs_Foncière moyennes
5      les plus élevées pour les départements 6, 13, 33, 59, 69
6  2. Je sélectionne la valeur_moyenne de mon top 3 calculé dans ma table éphémère,
7  les CodeDept (6, 13, 33, 59, 69), les Commune associées ainsi que leur classement
8  3. Et je formate la valeur_moyenne de ma table rankvaleurcommune en lui appliquant un ROUND
9  */
```

	Département	Commune	Prix moyen
►	13	GIGNAC-LA-NERTHE	330000.0
	13	SAINT SAVOURNIN	314425.0
	13	CASSIS	313416.9
	33	LEGE-CAP-FERRET	549500.6
	33	VAYRES	335000.0
	33	ARCACHON	307435.9
	59	BERSEE	433202.0
	59	CYSOING	408550.0
	59	HALLUIN	322250.0
	6	SAINT-JEAN-CAP-FERRAT	968750.0
	6	EZE	655000.0
	6	MOUANS-SARTOUX	476898.1
	69	VILLE SUR JARNIOUX	485300.0
	69	LYON 2EME	455217.3
	69	LYON 6EME	426968.3

```
45  WITH
46
47  valeur_par_ville AS (
48
49      SELECT
50          CodeDept,
51          Commune,
52          avg(Valeur_Fonciere) as valeur
53      FROM vente
54      JOIN bien
55      ON bien.Id_bien=vente.Id_bien
56      JOIN commune
57      ON commune.Id_com=bien.Id_com
58      WHERE CodeDept IN (6,13,33,59,69)
59      GROUP BY Commune)
60
61
62      SELECT
63          CodeDept AS "Département",
64          Commune AS "Commune",
65          round(valeur,1) AS "Prix moyen"
66  FROM (
67      SELECT CodeDept, Commune, valeur,
68      rank() OVER (PARTITION BY CodeDept ORDER BY valeur DESC) AS rang
69      FROM valeur_par_ville) AS result
70      WHERE rang <= 3
71
72  WHERE rang <= 3
73
74  FROM valeur_par_ville AS result
75
76  SELECT CodeDept, Commune, valeur,
77  rank() OVER (PARTITION BY CodeDept ORDER BY valeur DESC) AS rang
78  FROM valeur_par_ville
79  WHERE rang <= 3
```

# BILAN

## DIFFICULTÉS RENCONTRÉES

- ⚠ Découverte de la modélisation d'une BDD et de son implémentation
- ⚠ Liaison des tables et création de la FK « Id\_bien » dans la table vente
- ⚠ COURS OpenClassrooms très longs
- ⚠ Requête 8 (création de tables éphémères)
- ⚠ Requête 9 (Rank Over partition)

## COMPÉTENCES DÉVELOPÉES

- ✓ Modélisation UML
- ✓ Dépendances fonctionnelles
- ✓ Normalisation
- ✓ Requêtes avancées MySQL
- ✓ Draw.io
- ✓ SQL PowerArchitect

# Merci !