



Création et utilisation de la base de données

DJIMA Kémi



Laplace Immo

Contexte du projet

LOREM IPSUM DOLOR SIT AMET





Objectif stratégique : Créer un **modèle prédictif des prix de l'immobilier** pour optimiser l'accompagnement des clients et affiner la stratégie commerciale.

Contexte métier : Le marché immobilier est très concurrentiel, et la valorisation des données devient un levier clé de différenciation.

Rôle du projet DATAImmo : Collecter, structurer et analyser les données foncières et démographiques afin d'alimenter ce futur modèle prédictif.



La stratégie de sauvegarde et la conformité RGPD

-  **Aucune donnée personnelle sensible n'a été utilisée** (pas de noms, prénoms, coordonnées, etc.)
-  **Les données traitées sont purement statistiques et anonymisées**
-  Respect du **principe de minimisation** des données : uniquement les informations nécessaires à l'analyse
-  **Pas d'identification possible des individus** dans les ventes immobilières

Les données initiales

Les données initiales utilisées


1. Valeurs foncières

Données issues des ventes immobilières

 Date de vente, montant, surface, type de bien


2. Données communales (INSEE)

Données démographiques et géographiques

 Code commune, nom, population, unité urbaine

3. Référentiel géographique (data.gouv.fr)

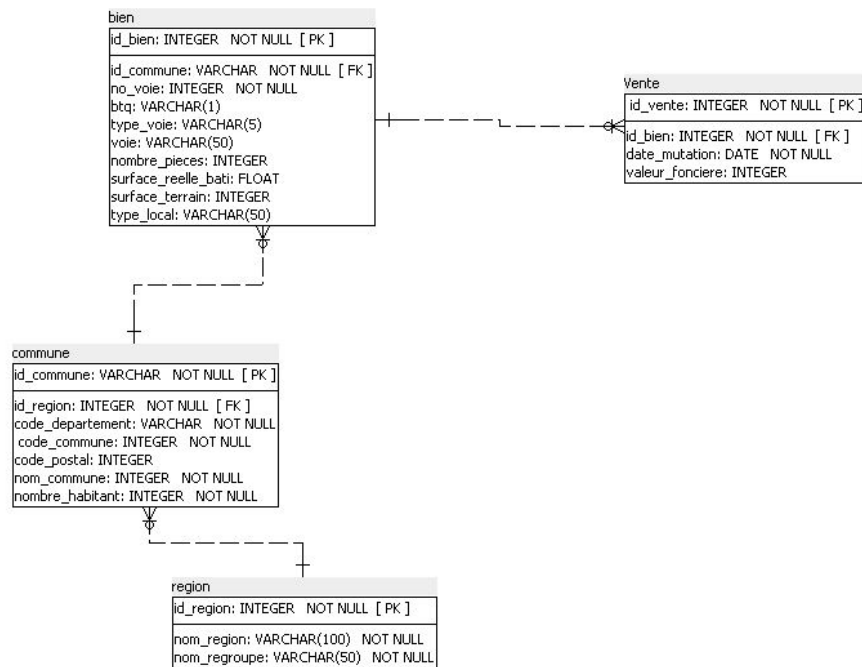
Structure territoriale française

 Liens entre communes, départements, régions, académies

L'extrait du dictionnaire des données

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION
No disposition	Numéro d'enregistrement de la disposition	Entier	10	Élémentaire	Doit être renseignée
Date mutation	Date de la transaction	Date	10	Élémentaire	
Nature mutation	Type de transaction (vente, échange...)	/VARCHAR(50	50	Élémentaire	>= 0
Valeur fonciere	Montant de la transaction	Float	10	Élémentaire	
No voie	Numéro de voie	Entier	5	Élémentaire	Facultatif
B/T/Q	Indication de bâtiment, tour ou quartier	/VARCHAR(10	10	Facultatif	
Code type de voie	Code du type de voie	VARCHAR(2)	2	Élémentaire	Élémentaire
Type de voie	Libellé du type de voie (rue, avenue...)	VARCHAR(50	50	Élémentaire	
Code voie	Code de la voie	VARCHAR(5)	5	Élémentaire	Élémentaire
Voie	Nom de la voie	VARCHAR(100	100	Élémentaire	
Code ID commune	Identifiant unique de la commune	VARCHAR(10	10	Élémentaire	Élémentaire
Code postal	Code postal du bien	VARCHAR(5)	5	Élémentaire	
Commune	Nom de la commune	VARCHAR(100	100	Élémentaire	Élémentaire
Code departement	Code du département	VARCHAR(3)	3	Élémentaire	
Code commune	Code de la commune (INSEE)	VARCHAR(3)	3	Élémentaire	Facultatif
Préfixe de section	Préfixe cadastral de la section	VARCHAR(5)	5	Facultatif	
Section	Code de la section cadastrale	VARCHAR(5)	5	Élémentaire	Élémentaire
No plan	Numéro du plan cadastral	VARCHAR(10	10	Élémentaire	
1er lot	Identifiant du premier lot	VARCHAR(10	10	Facultatif	Facultatif
Surface Carrez du 1	Surface Carrez en m² du 1er lot	Float	10	Facultatif	
Nombre de lots	Nombre total de lots	Entier	2	Élémentaire	Élémentaire
Code type local	Code du type de local	VARCHAR(2)	2	Élémentaire	
Type local	Type du bien (Maison, Appartement...)	VARCHAR(50	50	Élémentaire	Élémentaire
Surface reelle bati	Surface réelle bâtie (m²)	Float	10	Élémentaire	

Le schéma relationnel normalisé



La base de données avec les tables créées et les données chargées

```
mysql> SHOW TABLES;
```

```
+-----+
| Tables_in_dataimmo |
+-----+
| bien                |
| commune             |
| region              |
| vente               |
+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM region;
```

id_region	nom_region	nom_regroupe
0	Collectivités d'outre-mer	DROM-COM
1	Guadeloupe	DROM-COM
2	Martinique	DROM-COM
3	Guyane	DROM-COM
4	La Réunion	DROM-COM
6	Mayotte	DROM-COM
11	Ile-de-France	Ile-de-France
24	Centre-Val de Loire	Province
27	Bourgogne-Franche-Comté	Province
28	Normandie	Province
32	Hauts-de-France	Province
44	Grand Est	Province
52	Pays de la Loire	Province
53	Bretagne	Province
75	Nouvelle-Aquitaine	Province
76	Occitanie	Province
84	Auvergne-Rhône-Alpes	Province
93	Provence-Alpes-Côte d'Azur	Province
94	Corse	Province

```
19 rows in set (0.00 sec)
```


La base de données avec les tables créées et les données chargées

```
mysql> SELECT * FROM commune LIMIT 5;
```

id_commune	id_region	code_departement	code_commune	code_postal	nom_commune	nombre_habitant
01001	84	01	1	84001	L'Abergement-Clémenciat	798
01002	84	01	2	84002	L'Abergement-de-Varey	257
01004	84	01	4	84004	Ambérieu-en-Bugey	14514
01005	84	01	5	84005	Ambérieux-en-Dombes	1776
01006	84	01	6	84006	Ambléon	118

```
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM bien LIMIT 5;
```

id_bien	id_commune	no_voie	btq	type_voie	voie	nombre_pieces	surface_reelle_bati	sur
1	01103		347	RUE	DU CHATEAU	3	48	
2	06004		4	BD	EDOUARD BAUDOIN	1	40	
3	06088		20	B	RUE MARCEAU	3	82	
4	06123		550	RTE	DES VESPINS RN7	1	27	
5	13005		9300	RES	LES ARPEGES BD DES ABA	2	47	

```
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM vente LIMIT 5;
```

id_vente	id_bien	date_mutation	valeur_foncieres
1	1	2020-01-02	165000.00
2	2	2020-01-02	355680.00
3	3	2020-01-02	229500.00
4	4	2020-01-02	125000.00
5	5	2020-01-02	90000.00

```
5 rows in set (0.00 sec)
```


Les requêtes ou screenshot qui permettent de démontrer le bon chargement des données

```
mysql> LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/vente.csv'  
-> INTO TABLE vente  
-> FIELDS TERMINATED BY ','  
-> ENCLOSED BY '"'  
-> IGNORE 1 LINES  
-> (id_vente, id_bien, date_mutation, valeur_fonciere);  
Query OK, 34169 rows affected, 34169 warnings (0.31 sec)  
Records: 34169  Deleted: 0  Skipped: 0  Warnings: 34169
```



Requêtes SQL et résultats

Sous-titre

Requête 1

```
mysql> SELECT COUNT(*) AS total_appartements_vendus  
-> FROM vente  
-> JOIN bien ON vente.id_bien = bien.id_bien  
-> WHERE bien.type_local = 'Appartement'  
-> AND vente.date_mutation BETWEEN '2020-01-01' AND '2020-06-30';
```

total_appartements_vendus
31378

1 row in set (0.07 sec)

Requête 2

```
mysql> SELECT r.nom_region, COUNT(*) AS nb_ventes
-> FROM vente v
-> JOIN bien b ON v.id_bien = b.id_bien
-> JOIN commune c ON b.id_commune = c.id_commune
-> JOIN region r ON c.id_region = r.id_region
-> WHERE b.type_local = 'Appartement'
-> AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
-> GROUP BY r.nom_region
-> ORDER BY nb_ventes DESC;
```

nom_region	nb_ventes
Ile-de-France	13996
Provence-Alpes-Côte d'Azur	3649
Auvergne-Rhône-Alpes	3252
Nouvelle-Aquitaine	1932
Occitanie	1640
Pays de la Loire	1357
Hauts-de-France	1254
Grand Est	984

Requête 3

```
mysql> SELECT b.nombre_pieces, COUNT(*) AS nb_ventes,  
->      ROUND(100 * COUNT(*) / (SELECT COUNT(*)  
->      FROM vente v  
->      JOIN bien b ON v.id_bien = b.id_bien  
->      WHERE b.type_local = 'Appartement'), 2) AS pourcentage  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> WHERE b.type_local = 'Appartement'  
-> GROUP BY b.nombre_pieces  
-> ORDER BY nb_ventes DESC;
```

nombre_pieces	nb_ventes	pourcentage
2	9783	31.18
3	8966	28.57
1	6739	21.48
4	4460	14.21
5	1114	3.55
6	204	0.65
7	54	0.17
0	30	0.10
8	17	0.05
9	8	0.03
10	2	0.01
11	1	0.00

12 rows in set (0.09 sec)

Requête 4

```
mysql> SELECT c.code_departement,  
->      ROUND(AVG(v.valeur_fonciere / b.surface_reelle_bati), 2) AS prix_m2  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> WHERE b.surface_reelle_bati > 0  
-> GROUP BY c.code_departement  
-> ORDER BY prix_m2 DESC  
-> LIMIT 10;
```

code_departement	prix_m2
75	12122.16
92	7413.65
94	5395.98
06	4681.76
93	4363.01
74	4149.56
78	4126.9
69	4063.35
2A	3905.71
33	3838.74

10 rows in set (0.10 sec)

Requête 5

```
mysql> SELECT ROUND(AVG(v.valeur_fonciere / b.surface_reelle_bati), 2) AS prix_m2_maison_idf  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> JOIN region r ON c.id_region = r.id_region  
-> WHERE b.surface_reelle_bati > 0  
-> AND b.type_local = 'Maison'  
-> AND r.nom_region = 'Île-de-France';
```

```
+-----+  
| prix_m2_maison_idf |  
+-----+  
|           3997.71 |  
+-----+  
1 row in set (0.03 sec)
```


Requête 6

```
mysql> SELECT v.valeur_fonciere, r.nom_region, b.surface_reelle_bati  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> JOIN region r ON c.id_region = r.id_region  
-> WHERE b.type_local = 'Appartement'  
-> ORDER BY v.valeur_fonciere DESC  
-> LIMIT 10;
```

valeur_fonciere	nom_region	surface_reelle_bati
9000000.00	Ile-de-France	10
8600000.00	Ile-de-France	62
8577713.00	Ile-de-France	289
7620000.00	Ile-de-France	42
7600000.00	Ile-de-France	200
7535000.00	Ile-de-France	143
7420000.00	Ile-de-France	357
7200000.00	Ile-de-France	241
7050000.00	Ile-de-France	310
6600000.00	Ile-de-France	76

10 rows in set (0.17 sec)

Requête 7

```
mysql> SELECT
->     ROUND(
->         100 * (t2.nb - t1.nb) / t1.nb, 2
->     ) AS taux_evolution_pourcent
-> FROM (
->     SELECT COUNT(*) AS nb FROM vente
->     WHERE date_mutation BETWEEN '2020-01-01' AND '2020-03-31'
-> ) t1,
-> (
->     SELECT COUNT(*) AS nb FROM vente
->     WHERE date_mutation BETWEEN '2020-04-01' AND '2020-06-30'
-> ) t2;
```

taux_evolution_pourcent
3.68

1 row in set (0.03 sec)

Requête 8

```
mysql> SELECT r.nom_region,  
-> ROUND(AVG(v.valeur_fonciere / b.surface_reelle_bati), 2) AS prix_m2  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> JOIN region r ON c.id_region = r.id_region  
-> WHERE b.surface_reelle_bati > 0  
-> AND b.type_local = 'Appartement'  
-> AND b.nombre_pieces > 4  
-> GROUP BY r.nom_region  
-> ORDER BY prix_m2 DESC;
```

nom_region	prix_m2
Ile-de-France	8003.39
La Réunion	3659.83
Corse	3046.47
Provence-Alpes-Côte d'Azur	3005.24
Auvergne-Rhône-Alpes	2768.87
Nouvelle-Aquitaine	2510.18
Bretagne	2271.86
Hauts-de-France	2203.61
Pays de la Loire	2186.72
Occitanie	2096.42
Normandie	1994.25
Centre-Val de Loire	1428.51
Grand Est	1313.26
Bourgogne-Franche-Comté	1068.93

Requête 9

```
mysql> SELECT c.nom_commune, COUNT(*) AS nb_ventes
-> FROM vente v
-> JOIN bien b ON v.id_bien = b.id_bien
-> JOIN commune c ON b.id_commune = c.id_commune
-> WHERE v.date_mutation BETWEEN '2020-01-01' AND '2020-03-31'
-> GROUP BY c.nom_commune
-> HAVING nb_ventes >= 50
-> ORDER BY nb_ventes DESC;
```

nom_commune	nb_ventes
Paris 17e Arrondissement	228
Paris 15e Arrondissement	215
Paris 18e Arrondissement	209
Nice	173
Paris 11e Arrondissement	169
Paris 16e Arrondissement	165
Bordeaux	157
Paris 14e Arrondissement	146
Paris 20e Arrondissement	127
Nantes	119
Paris 19e Arrondissement	116
Paris 12e Arrondissement	110
Paris 10e Arrondissement	109
Grenoble	106
Paris 9e Arrondissement	106
Boulogne-Billancourt	99
Paris 13e Arrondissement	94

Requête 10

```
mysql> SELECT
->   ROUND(
->     100 * (m3 - m2) / m2, 2
->   ) AS difference_pourcent
-> FROM (
->   SELECT AVG(valeur_fonciere / surface_reelle_bati) AS m2
->   FROM vente v JOIN bien b ON v.id_bien = b.id_bien
->   WHERE b.surface_reelle_bati > 0 AND b.nombre_pieces = 2
-> ) a,
-> (
->   SELECT AVG(valeur_fonciere / surface_reelle_bati) AS m3
->   FROM vente v JOIN bien b ON v.id_bien = b.id_bien
->   WHERE b.surface_reelle_bati > 0 AND b.nombre_pieces = 3
-> ) b;
```

```
+-----+
| difference_pourcent |
+-----+
|          -13.46 |
+-----+
1 row in set (0.05 sec)
```

Requête 11

```
mysql> WITH moyenne_par_commune AS (  
-> SELECT  
-> LEFT(c.id_commune, 2) AS departement,  
-> c.nom_commune,  
-> ROUND(AVG(v.valeur_fonciere), 2) AS moyenne_valeur,  
-> ROW_NUMBER() OVER (  
-> PARTITION BY LEFT(c.id_commune, 2)  
-> ORDER BY AVG(v.valeur_fonciere) DESC  
-> ) AS rang  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> WHERE LEFT(c.id_commune, 2) IN ('06', '13', '33', '59', '69')  
-> GROUP BY c.nom_commune, c.id_commune  
-> )  
-> SELECT departement, nom_commune, moyenne_valeur  
-> FROM moyenne_par_commune  
-> WHERE rang <= 3  
-> ORDER BY departement, rang;
```

departement	nom_commune	moyenne_valeur
06	Saint-Jean-Cap-Ferrat	968750.00
06	Eze	655000.00
06	Mouans-Sartoux	476898.10
13	Gignac-la-Nerthe	330000.00
13	Saint-Savournin	314425.00
13	Cassis	313416.88

Requête 12

```
mysql> SELECT c.nom_commune,  
->      ROUND(COUNT(*) / c.nombre_habitant * 1000, 2) AS ventes_pour_1000_hab  
-> FROM vente v  
-> JOIN bien b ON v.id_bien = b.id_bien  
-> JOIN commune c ON b.id_commune = c.id_commune  
-> WHERE c.nombre_habitant >= 10000  
-> GROUP BY c.nom_commune, c.nombre_habitant  
-> ORDER BY ventes_pour_1000_hab DESC  
-> LIMIT 20;
```

nom_commune	ventes_pour_1000_hab
Paris 2e Arrondissement	5.84
Paris 1er Arrondissement	4.92
Paris 3e Arrondissement	4.69
Arcachon	4.62
La Baule-Escoublac	4.58
Paris 4e Arrondissement	4.08
Roquebrune-Cap-Martin	3.99
Paris 8e Arrondissement	3.83
Sanary-sur-Mer	3.50
Paris 9e Arrondissement	3.43
La Londe-les-Maures	3.43
Paris 6e Arrondissement	3.38
Saint-Cyr-sur-Mer	3.24
Chantilly	3.13



Merci !