





- Le marché immobilier est un marché complexe avec une quantité d'information très forte.
- Analyser et comprendre ces données sont essentielles pour les différents acteurs de ce marché



Laplace Immo lance le projet DATAImmo, l'utilisation d'une base de données du marché immobilier Français afin de prévoir l'évolution du marché immobilier et mieux accompagné ses clients



Données communes

CODREG	CODDEP	CODARR	CODCAN	CODCOM	COM	PMUN	PCAP	PTOT
84	01	02	08	001	L'Abergement-Clémenciat	779	19	798
84	01	01	01	002	L'Abergement-de-Varey	256	1	257
84	01	01	01	004	Ambérieu-en-Bugey	14134	380	14514
84	01	02	22	005	Ambérieux-en-Dombes	1751	25	1776
84	01	01	04	006	Ambléon	112	6	118
84	01	01	01	007	Ambronay	2800	115	2915
84	01	01	01	008	Ambutrix	762	15	777



Référenciel géographique

reggrp_nc	reg_nom	reg_nom	aca_nom	dep_nom	com_code	com_code	com_code	com_id	com_nom	com_nom	com_nom	uu_code	uu_id	uucr_id	uucr_nom	ze_id	dep_code	dep_id	dep_nom	dep_num_nom
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01001		1001	1001	C01001	L'ABERGEME L'ABERGEME L'Abergement-Clémenciat	SO	CR01001	L'Abergement-Clémenciat	ZE8213	1 D001	Ain (01)	01 - Ain					
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01002		1002	1002	C01002	L'ABERGEME L'ABERGEME L'Abergement-de-Varey	SO	CR01002	L'Abergement-de-Varey	ZE8201	1 D001	Ain (01)	01 - Ain					
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01003		1003	1003	C01003	AMAREINS AMAREINS Amareins	SO	SO	Amareins	SO	1 D001	Ain (01)	01 - Ain					
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01004		1004	1004	C01004	AMBERIEU ET AMBERIEU-E Ambérieu-en-Dombes	1303	UU01303	AMBERIEU ET AMBERIEU-E Ambérieu-en-Dombes	UU01303	UU01303	Amberieu-en-Dombes	ZE8201	1 D001	Ain (01)	01 - Ain		
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01005		1005	1005	C01005	AMBERIEUX AMBERIEUX Ambérieux-en-Dombes	SO	CR01005	AMBERIEUX AMBERIEUX Ambérieux-en-Dombes	ZE8213	1 D001	Ain (01)	01 - Ain					
Province	Auvergne-Rhône-Alpes:Lyon	Ain	01006		1006	1006	C01006	AMBLEON AMBLEON Ambléon	SO	CR01006	AMBLEON AMBLEON Ambléon	ZE8216	1 D001	Ain (01)	01 - Ain					



Valeurs Foncières

Date mutation	Nature mutation	Valeur foncière	No vc	B/T	Code type de v	Type de v	Code vc	Voie	Code ID commu	Code pos	Commune	Code département	Code commune	Préfixe de sect	Secti
2020/01/02	Vente	165000	347		0 RUE	20 DU CHATEAU	1	1170 CHEVRY		01	103				A
2020/01/02	Vente	355680	4		15 BD	1000 EDOUARD BAUDOUIN	205	6160 ANTIBES		06	004				CP
2020/01/02	Vente	229500	20 B		0 RUE	3975 MARCEAU	142	6000 NICE		06	088				LS
2020/01/02	Vente	125000	550		3 RTE	1011 DES VESPINS RN7	228	6700 SAINT LAURENT DU VAR		06	123				AO
2020/01/02	Vente	90000	9300		18 RES	A084 LES ARPEGES BD DES AI	326	13400 AUBAGNE		13	005				AW
2020/01/02	Vente	93000	27		0 RUE	791 DU GRAND MADIER	331	13600 LA CIOTAT		13	028				AB
2020/01/02	Vente	298100	360		1 AV	7515 DU PRADO	327	13008 MARSEILLE 8EME		13	208				843 C
2020/01/02	Vente	163500	5076 F		25 PARC	A312 DESSUARD	343	13012 MARSEILLE 12EME		13	212				873 A
2020/01/02	Vente	53000	1194		0 RUE	1090 DE NORMANDIE	372	14510 HOULGATE		14	338				AC
2020/01/02	Vente	136000	30		2 ALL	1037 DES NOISETIERS	366	14100 LISIEUX		14	366				BH



fx = Table.TransformColumnTypes(#"En-têtes promus",{{"CODREG", type text}, {"CODDEP", type text}, {"CODARR", Int64.Type}, {"CODCAN", Int64.Type}, {"CODCOM", type text}, {"COM", type text}, {"PMUN", Int64.Type}, {"PCAP", Int64.Type}, {"PTOT", Int64.Type}})

A ^B _C	CODREG	A ^B _C	CODDEP	1 ² ₃	CODARR	1 ² ₃	CODCAN	A ^B _C	CODCOM	A ^B _C	COM	1 ² ₃	PMUN	1 ² ₃	PCAP	1 ² ₃	PTOT
84	01			2		8	001				L'Abergement-Clémenciat		779	19		798	
84	01			1		1	002				L'Abergement-de-Varey		256	1		257	
84	01			1		1	004				Ambérieu-en-Bugey		14134	380		14514	
84	01			2		22	005				Ambérieux-en-Dombes		1751	25		1776	
84	01			1		4	006				Ambléon		112	6		118	
84	01			1		1	007				Ambronay		2800	115		2915	
84	01			1		1	008				Ambutrix		762	15		777	
84	01			1		4	009				Anderet-et-Condon		326	9		335	
84	01			1		10	010				Anglefort		1105	17		1122	
84	01			4		14	011				Aprie						
84	01			1		10	012				Ara						
84	01			1		1	013				Ara						
84	01			4		15	014				Arb						
84	01			1		4	015				Arb						
84	01			2		17	016				Arb						
84	01			1		1	017				Arg						
84	01			1		10	019				Arm						
84	01			2		22	021				Arst						
84	01			1		10	022				Art						
84	01			2		17	023				Asn						
84	01			2		2	024				Att						
84	01			2		17	025										

Fusionner les colonnes

Choisissez comment fusionner les colonnes sélectionnées.

Séparateur: --Aucun--

Nouveau nom de colonne (facultatif): code_insee

OK Annuler

CODREG	CODDEP	CODARR	CODCAN	CODCOM	COM	PMUN	PCAP	PTOT	code_INSEE
84 01		2	8 001		L'Abergement-Clémenciat	779	19	798	01001
84 01		1	1 002		L'Abergement-de-Varey	256	1	257	01002
84 01		1	1 004		Ambérieu-en-Bugey	14134	380	14514	01004
84 01		2	22 005		Ambérieux-en-Dombes	1751	25	1776	01005
84 01		1	4 006		Ambléon	112	6	118	01006
84 01		1	1 007		Ambronay	2800	115	2915	01007
84 01		1	1 008		Ambutrix	762	15	777	01008

- Rajout du champ code_insee via l'outil Power Query sur Excel.
- Transformation de type des colonnes Code département et Code commune pour les concaténer.



= Table.AddColumn(#"Type modifié", "Code postale", each Text.PadStart(Text.From([Code postal]), 5, "0"))

	ABC 123	Identifiant local	ABC 123	Surface reelle bati	ABC 123	Nombre pieces principales	A ^B C	Nature culture	ABC 123	Nature culture speciale	ABC 123	Surface terrain	A ^B C	Nom de l'acquéreur	ABC 123	Code postal
1		null		48		3		null				null		GUIRAO		01170
2		null		40		1		null				null		HARNOIS		06160
3		null		82		3		null				null		ROGIER		06000
4		null		27		1		null				null		ROCOLUIFR		06700

Requêtes [2]

Le code postal est de type int ce qui supprime le 0 devant le code postal et donc donne un code postal incorrect

Même après l'avoir passé en text le format reste le même, on doit donc forcé a mettre un 0 si la chaîne de caractère <5

ABC 123	Code postal	ABC 123	Code postale
1170		01170	
6160		06160	
6000		06000	
6700		06700	
13400		13400	
13600		13600	
13008		13008	
13012		13012	
14510		14510	
14100		14100	

DICTIONNAIRE DES DONNÉES - Régions

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
reg_code	Code Région	Varchar		3 élémentaire	Code INSEE de la région		clé primaire
reg_nom	Nom région	Varchar		50 élémentaire			

DICTIONNAIRE DES DONNÉES - Départements

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
dep_code	Code Département	Varchar		3 élémentaire	code insee departement (2B ,974)		Clé primaire
dep_nom	Nom département	Varchar		50 élémentaire			
reg_code	code régions	Varchar		3 élémentaire	Lien vers régions		Clé étrangère

DICTIONNAIRE DES DONNÉES - communes

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
code_insee	Code insee unique	Varchar		6	Concaténatior Id unique de la commune	Concaténation code dep+code commune	Clé Primaire
com_nom	Nom de la commune	Varchar		100 élémentaire	Standardisé		
population	Population totale de la commune	Intgerer		calculé	somme de pmun et pcap	somme	
dep_code	code du département	Varchar		3 élémentaire	Lien vers départements		Clé étrangère

DICTIONNAIRE DES DONNÉES - Type_local

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
id_type_local	id unique du type	integer		élémentaire	Auto-incrémentation		
libelle_type_local	Description du type (Maison, apt,...)	Varchar	50	élémentaire			clé primaire

DICTIONNAIRE DES DONNÉES - Nature_Mutation

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
id_nature_mutation	identifiant unique	integer		élémentaire	auto-incrémentation		Clé Primaire
libelle_nature_mutation	Description de la nature mutation	Varchar	50	élémentaire			

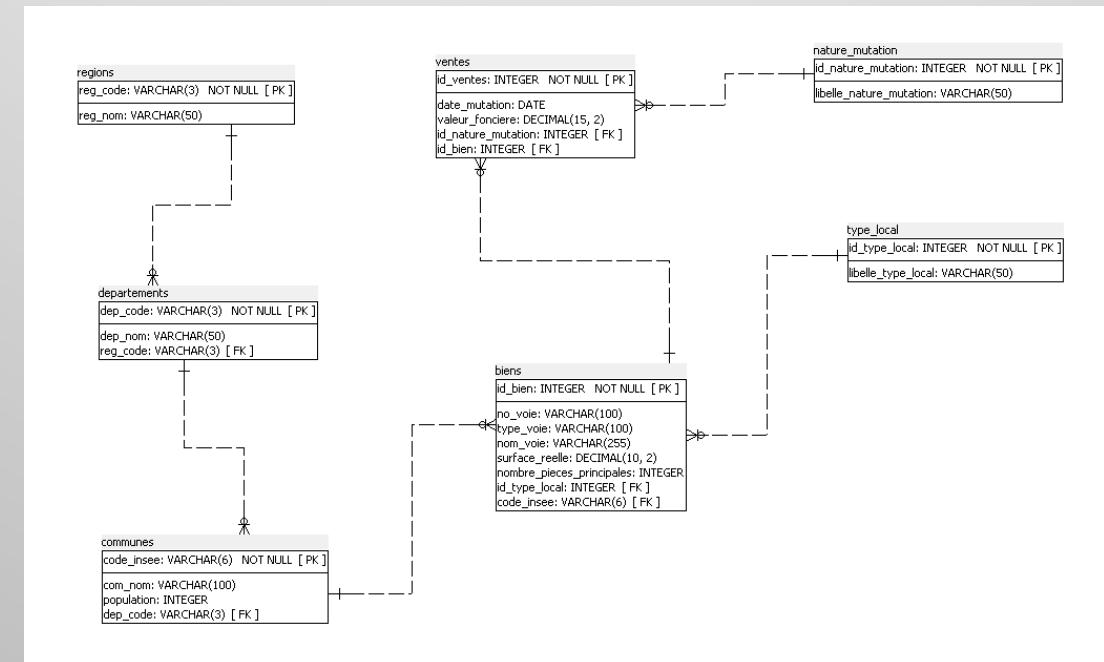
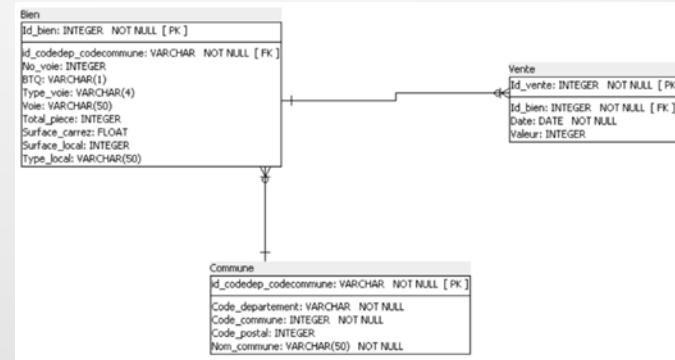
DICTIONNAIRE DES DONNÉES - biens

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
id_bien	Id unique du bien	integer		élémentaire	auto-incrémentation		
no_voie	Numéro de la rue du bien	varchar	20	élémentaire	peut être null		Clé Primaire
type_voie	Type de la voie du bien (rue, impasse, ...)	varchar	20	élémentaire			
nom_voie	Nom de la voie	varchar	50	élémentaire			
surface_reelle_bati	surface réelle bâti	decimal	10,2	élémentaire	en m ²		
nombre_pieces_principales	Nombre de pièces principales	integer		élémentaire			
id_type_local	type de bien	integer		élémentaire	lien vers type local		clé étrangère
code_insee	code unique de la commune du bien	varchar	6	élémentaire	lien vers communes		clé étrangère

DICTIONNAIRE DES DONNÉES - Ventes

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	Colonne1
id_ventes	id de la transaction	integer		élémentaire	auto-incrémentation		Clé primaire
date_mutation	Date de la signature	Date		élémentaire	Format yyyy-mm-dd		
Valeur_fonciere	Montant de la vente	décimal	15,2	élémentaire			
id_bien	Bien concerné	integer		élémentaire	lien vers biens		clé étrangère
id_nature_mutation	Nature de la vente	integer		élémentaire	lien vers nature_mutation		clé étrangère

Schéma relationnel de la Base de données



Création des tables selon notre dictionnaire de données.

```
create database LaplaceImmo;
Use LaplaceImmo;
Create table regions (
    reg_code VARCHAR(3) PRIMARY KEY,
    reg_nom VARCHAR(50));
Create table departements (
    dep_code Varchar(3) primary key,
    dep_nom varchar(50),
    reg_code varchar(3),
    foreign key (reg_code) references regions(reg_code));
create table communes (
    code_insee varchar(6) primary key ,
    com_nom varchar(100),
    population int,
    dep_code varchar(3),
    foreign key (dep_code) references departements(dep_code));
```

```
create table type_local (
    id_type_local int auto_increment primary key,
    libelle_type_local varchar (50));
create table nature_mutation (
    id_nature_mutation int auto_increment primary key,
    libelle_nature_mutation varchar(50));

create table biens (
    id_bien int auto_increment primary key,
    no_voie varchar(3),
    type_voie varchar(20),
    nom_voie varchar(50),
    surface_reelle decimal (10,2),
    nombre_pieces_principales int,
    id_type_local int,
    code_insee varchar(6),
    foreign key (id_type_local) references type_local(id_type_local),
    foreign key (code_insee) references communes(code_insee));
create table ventes (
    id_ventes int auto_increment primary key,
    date_mutation date,
    valeur_fonciere decimal(15,2),
    id_nature_mutation int,
    id_bien int,
    foreign key (id_nature_mutation) references nature_mutation(id_nature_mutation),
    foreign key (id_bien) references biens(id_bien));
```

On crée les tables temporaires avec tous les champs en varchar pour récupérer toutes les données sans casse.

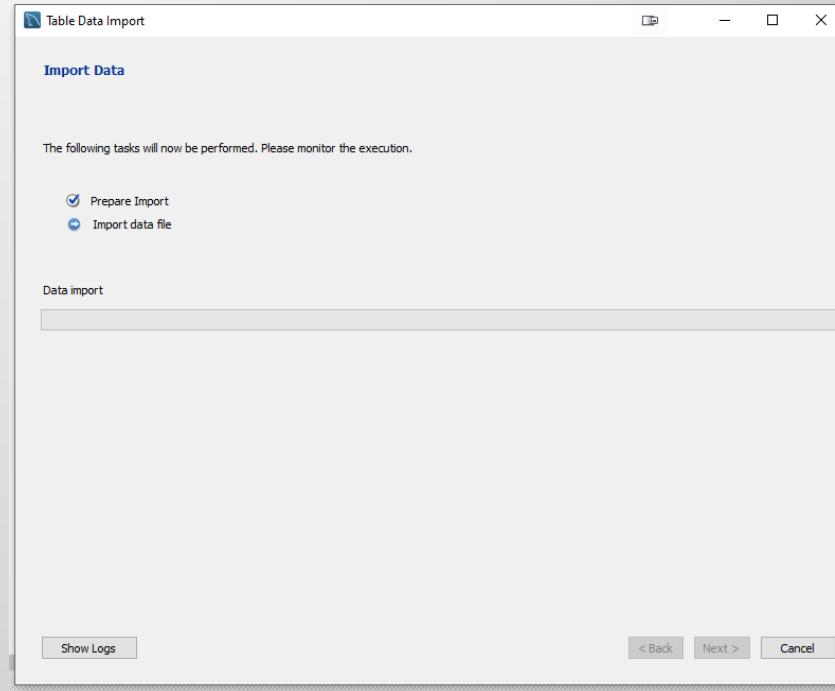
```

CREATE TABLE temp_comunes (
    CODREG VARCHAR(10),
    CODDEP VARCHAR(10),
    CODARR VARCHAR(10),
    CODCAN VARCHAR(10),
    CODCOM VARCHAR(10),
    COM VARCHAR(255),
    PMUN VARCHAR(20),
    PCAP VARCHAR(20),
    PTOT VARCHAR(20),
    CODE_INSEE VARCHAR(10)
);

CREATE table temp_geo (

    reggrp_nom Varchar(100), reg_nom Varchar(100), reg_nom_old Varchar(100), aca_nom Varchar(100), dep_nom Varchar(100),
    com_code Varchar(100), com_code1 Varchar(100), com_code2 Varchar(100), com_id Varchar(100),
    com_nom_maj_court Varchar(100), com_nom_maj Varchar(100), com_nom Varchar(100),
    uu_code TEXT, uu_id Varchar(100), uucr_nom Varchar(100), ze_id Varchar(100),
    dep_code Varchar(100), dep_id Varchar(100), dep_nom_num Varchar(100), dep_num_nom Varchar(100),
    aca_code Varchar(100), aca_id Varchar(100), reg_code Varchar(100), reg_id Varchar(100),
    reg_code_old Varchar(100), reg_id_old Varchar(100), fd_id Varchar(100), fr_id Varchar(100), fe_id Varchar(100),
    uu_id_99 Varchar(100), au_code Varchar(100), au_id Varchar(100), auc_id Varchar(100), auc_nom Varchar(100),
    uu_id_10 Varchar(100), geolocalisation Varchar(100)
);
CREATE TABLE temp_ventes (
    date_mutation VARCHAR(20), nature_mutation VARCHAR(100), valeur_fonciere VARCHAR(50),
    no_voie VARCHAR(20), btq VARCHAR(20), code_type_voie VARCHAR(20), type_voie VARCHAR(50),
    code_voie VARCHAR(20), voie VARCHAR(255), code_postal VARCHAR(10),
    commune VARCHAR(255), code_departement VARCHAR(10), code_commune VARCHAR(10),
    code_type_local VARCHAR(10), type_local VARCHAR(100),
    surface_reelle_bati VARCHAR(20), nombre_pieces_principales VARCHAR(10),
    surface_terrain VARCHAR(20), code_insee VARCHAR(20)
);

```



```

use Laplaceimmo;

select 'temps_ventes' as table_name, count(*) as nb_lignes from temp_ventes
Union all
select 'temps_geo' as table_name, count(*) as nb_lignes from temp_geo
Union all
select 'temps_comunes' as table_name, count(*) as nb_lignes from temp_comunes
;

```

	table_name	nb_lignes
▶	temps_ventes	34169
▶	temps_geo	38916
▶	temps_comunes	34991

On insert les données depuis les tables temporaires vers nos tables de productions en convertissant correctement les champs qu'on veut récupérer.

D'abord les données géographiques puis les données type_local et nature_mutation...

```
INSERT INTO regions (reg_code, reg_nom)
SELECT DISTINCT reg_code, reg_nom
FROM temp_geo
WHERE reg_code IS NOT NULL;

INSERT INTO departements (dep_code, dep_nom, reg_code)
SELECT dep_code, MIN(dep_nom), reg_code
FROM temp_geo
WHERE dep_code IS NOT NULL
GROUP BY dep_code, reg_code;

INSERT INTO type_local (libelle)
SELECT DISTINCT type_local
FROM temp_ventes
WHERE type_local IS NOT NULL AND type_local <> '';

INSERT INTO nature_mutation (libelle)
SELECT DISTINCT nature_mutation
FROM temp_ventes
WHERE nature_mutation IS NOT NULL;
```

```
SET FOREIGN_KEY_CHECKS = 0;

INSERT INTO communes (code_insee, nom_com, population_tot, dep_code)
SELECT
    LPAD(CODE_INSEE, 5, '0') as insee,
    MIN(COM),
    MAX(CAST(REPLACE(PMUN, ' ', '') AS UNSIGNED) + CAST(REPLACE(PCAP, ' ', '') AS UNSIGNED)),
    LPAD(CODE_INSEE, 2, '0')
FROM temp_communes
WHERE CODE_INSEE IS NOT NULL
GROUP BY LPAD(CODE_INSEE, 5, '0'), CODDEP;

SET FOREIGN_KEY_CHECKS = 1;
```

```
INSERT INTO type_local (libelle_type_local)
SELECT DISTINCT type_local
FROM temp_ventes
WHERE type_local IS NOT NULL
    AND type_local != ''
ORDER BY type_local;

INSERT INTO nature_mutation (libelle_nature_mutation)
SELECT DISTINCT nature_mutation
FROM temp_ventes
WHERE nature_mutation IS NOT NULL
    AND nature_mutation != '';
```

... et enfin ventes et biens

```

set foreign_key_checks = 0;
INSERT INTO biens (
    no_voie,
    type_voie,
    nom_voie,
    code_insee,
    surface_reelle,
    id_type_local,
    nombre_pieces_principales
)
SELECT DISTINCT
    tv.no_voie,
    tv.type_voie,
    tv.voie,
    tv.code_insee,
    tv.surface_reelle_bati,
    tl.id_type_local,
    tv.nombre_pieces_principales
FROM temp_ventes tv
JOIN type_local tl ON tv.type_local = tl.libelle_type_local
WHERE tv.surface_reelle_bati > 0;
set foreign_key_checks = 1;

```

```

set foreign_key_checks = 0;
INSERT INTO ventes (date_mutation, valeur_fonciere, id_nature_mutation, id_bien)
SELECT
    STR_TO_DATE(tv.date_mutation, '%d/%m/%Y'),
    CAST(
        NULLIF(
            REPLACE(
                REGEXP_REPLACE(TRIM(tv.valeur_fonciere), '[^0-9,.]', ''),
                ',', '.'
            ),
            ''
        ) AS DECIMAL(15,2)
    ),
    n.id_nature_mutation,
    b.id_bien
FROM temp_ventes tv
JOIN nature_mutation n ON tv.nature_mutation = n.libelle_nature_mutation
JOIN biens b ON b.code_insee = LPAD(tv.code_insee, 5, '0')
    AND b.nom_voie = tv.voie
    AND (b.no_voie = tv.no_voie OR (b.no_voie IS NULL AND tv.no_voie IS NULL))
    AND (b.nombre_pieces_principales = CAST(tv.nombre_pieces_principales AS UNSIGNED)
        OR (b.nombre_pieces_principales IS NULL AND (tv.nombre_pieces_principales IS NULL OR tv.nombre_pieces_principales = '')))
WHERE tv.valeur_fonciere IS NOT NULL
    AND tv.valeur_fonciere != '';

```

On vérifie maintenant que les inserts ont bien été exécutés

```
Use LaplaceImmo;
select 'regions' as table_name,count('*') as nb_lignes from regions
union all
select 'departements' as table_name,count('*') as nb_lignes from departements
union all
select 'communes' as table_name,count('*') as nb_lignes from communes
union all
select 'type_local' as table_name,count('*') as nb_lignes from type_local
union all
select 'nature_mutation' as table_name,count('*') as nb_lignes from nature_mutation
union all
select 'biens' as table_name,count('*') as nb_lignes from biens
union all
select 'ventes' as table_name,count('*') as nb_lignes from ventes
;
```

	table_name	nb_lignes
▶	regions	19
	departements	109
	communes	34893
	type_local	2
	nature_mutation	1
	biens	33450
	ventes	36838

On a maintenant une base de données propre et peuplé de données.

Pour l'étude de la base de données nous avons décidé de se concentré sur le premier semestre de 2020.

Posons les bases de notre analyse en répondant aux questions « quoi ? » « où ? » et « quand ? »



Voyons d'abord l'état du marché dans sa globalité pour les appartements.

Requête n°1

```
Use LaplaceImmo;
SELECT COUNT(DISTINCT v.valeur_fonciere, v.date_mutation, b.code_insee) AS nb_appartements
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
Where b.id_type_local In (Select id_type_local from type_local where libelle_type_local Like 'Appartement%')
AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30';
```

nb_appartements
31382



On force les code INSEE à être du même format et on force l'inclusion des DOM /TOM

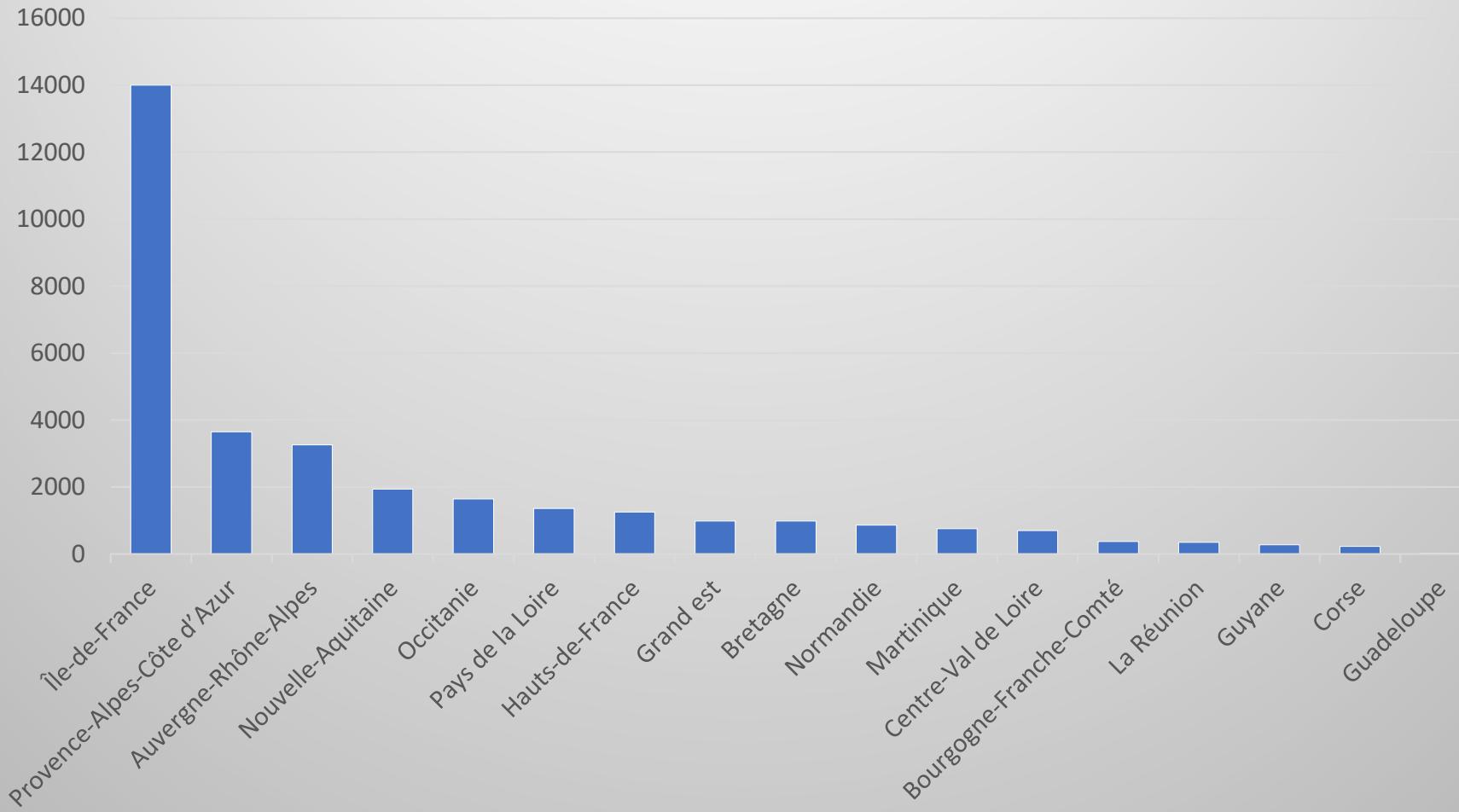
Requête n°2

```
Use LaplaceImmo;

SELECT
    r.reg_nom AS nom_region,
    COUNT(v.id_ventes) AS nombre_de_ventes
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN type_local tl ON b.id_type_local = tl.id_type_local
JOIN departements d ON d.dep_code = (
    CASE
        WHEN b.code_insee LIKE '97%' THEN LEFT(b.code_insee, 3)
        ELSE LEFT(LPAD(b.code_insee, 5, '0'), 2)
    END
)
JOIN regions r ON d.reg_code = r.reg_code
WHERE tl.libelle_type_local = 'Appartement'
    AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
GROUP BY r.reg_nom
ORDER BY nombre_de_ventes DESC;
```

nom_region	nombre_de_ventes
Ile-de-France	15198
Auvergne-Rhône-Alpes	3160
Provence-Alpes-Côte d'Azur	2369
Nouvelle-Aquitaine	2152
Occitanie	1726
Pays de la Loire	1469
Hauts-de-France	1257
Bretagne	1028
Normandie	1015
Grand Est	992
Centre-Val de Loire	723
Bourgogne-Franche-Comté	386
Martinique	98
La Réunion	44
Guyane	36
Guadeloupe	2

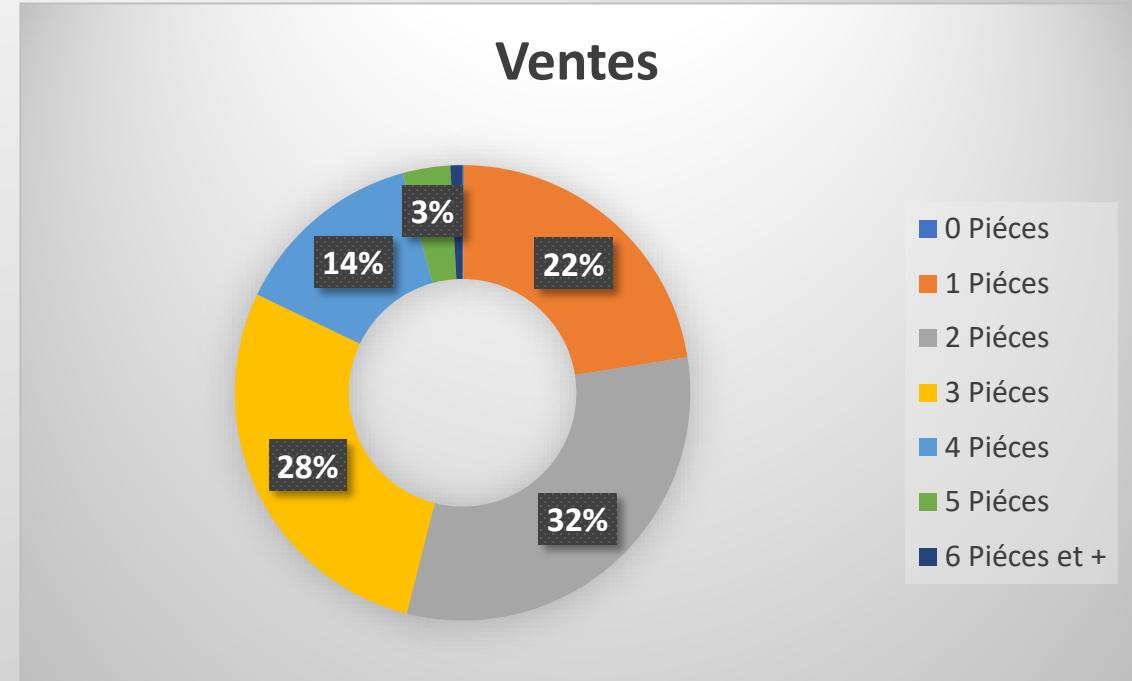
Nombre de ventes par Régions



Requête n°3

```
SELECT
    b.nombre_pieces_principales,
    COUNT(*) AS nb_ventes,
    ROUND(COUNT(*) * 100.0 / SUM(COUNT(*)) OVER(), 2) AS pourcentage
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
Where b.id_type_local In (Select id_type_local from type_local where libelle_type_local Like 'Appartement%')
AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
AND b.nombre_pieces_principales IS NOT NULL
GROUP BY b.nombre_pieces_principales
ORDER BY b.nombre_pieces_principales;
```

	nombre_pieces_principales	nb_ventes	pourcentage
0	30	0.09	
1	7605	22.41	
2	10672	31.44	
3	9557	28.16	
4	4639	13.67	
5	1145	3.37	
6	207	0.61	
7	56	0.16	
8	19	0.06	
9	8	0.02	
10	2	0.01	
11	1	0.00	



Requête n°4

```

SELECT
    d.dep_nom,
    CASE
        WHEN TRIM(b.code_insee) LIKE '97%' THEN LEFT(TRIM(b.code_insee), 3)
        ELSE LEFT(TRIM(b.code_insee), 2)
    END AS code_dep,
    ROUND(AVG(v.valeur_fonciere / b.surface_reelle), 2) AS prix_m2_moyen
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN departements d ON d.dep_code = (
    CASE
        WHEN TRIM(b.code_insee) LIKE '97%' THEN LEFT(TRIM(b.code_insee), 3)
        ELSE LEFT(TRIM(b.code_insee), 2)
    END
)
WHERE v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
    AND b.surface_reelle > 0
    AND v.valeur_fonciere > 0
GROUP BY d.dep_nom, code_dep
ORDER BY prix_m2_moyen DESC
LIMIT 10;

```

	dep_nom	code_dep	prix_m2_moyen
▶	Paris	75	12017.27
	Hauts-de-Seine	92	7383.15
	Val-de-Marne	94	5408.55
	Seine-Saint-Denis	93	4391.79
	Haute-Savoie	74	4172.86
	Yvelines	78	4114.50
	Rhône	69	4065.49
	Gironde	33	3872.39
	Var	83	3839.99
	Charente-Maritime	17	3620.14

Requête n°5

```

Use LaplaceImmo;
SELECT
    ROUND(AVG(v.valeur_fonciere / b.surface_reelle), 2) AS prix_m2_moyen_maison_IDF
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN departements d ON d.dep_code = (
    CASE
        WHEN TRIM(b.code_insee) LIKE '97%' THEN LEFT(TRIM(b.code_insee), 3)
        ELSE LEFT(TRIM(b.code_insee), 2)
    END
)
Where b.id_type_local In (Select id_type_local from type_local where libelle_type_local Like 'Maison%')
    AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
    AND b.surface_reelle > 0
    AND v.valeur_fonciere > 0
    AND d.reg_code = '11';

```

	prix_m2_moyen_maison_IDF
▶	4029.97

Requête n°8

```

SELECT
    r.reg_nom AS nom_region,
    ROUND(AVG(v.valeur_fonciere / b.surface_reelle), 2) AS prix_m2_moyen
FROM regions r
JOIN departements d ON r.reg_code = d.reg_code
JOIN biens b ON d.dep_code = (
CASE
    WHEN b.code_insee LIKE '97%' THEN LEFT(b.code_insee, 3)
    WHEN b.code_insee LIKE '201%' THEN '2A'
    WHEN b.code_insee LIKE '202%' THEN '2B'
    ELSE TRIM(LEADING '0' FROM LEFT(LPAD(b.code_insee, 5, '0'), 2))
END
)
JOIN ventes v ON b.id_bien = v.id_bien
WHERE b.id_type_local = 2 -- Appartement
    AND b.nombre_pieces_principales > 4
    AND b.surface_reelle > 0
    AND v.valeur_fonciere > 0
    AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
GROUP BY r.reg_nom
ORDER BY prix_m2_moyen DESC;

```

	nom_region	prix_m2_moyen
▶	Corse	6622.37
	Martinique	3796.95
	Provence-Alpes-Côte d'Azur	3691.85
	Nouvelle-Aquitaine	3638.14
	Île-de-France	3579.16
	Auvergne-Rhône-Alpes	3213.23
	Pays de la Loire	2615.73
	Normandie	2262.89
	Occitanie	2213.46
	Hauts-de-France	2058.63
	Bretagne	1893.62
	Centre-Val de Loire	1862.53
	Bourgogne-Franche-Comté	1787.66
	Grand Est	1786.26

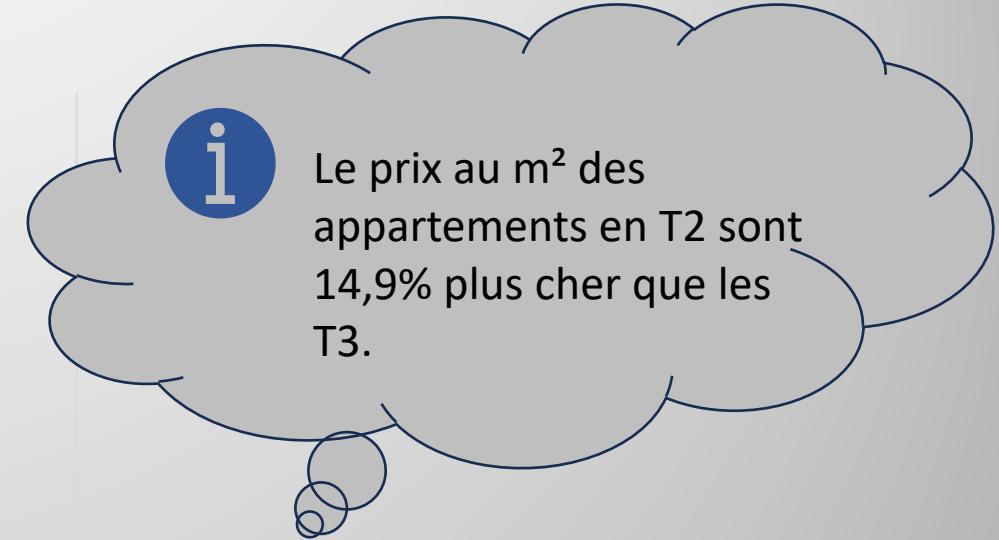
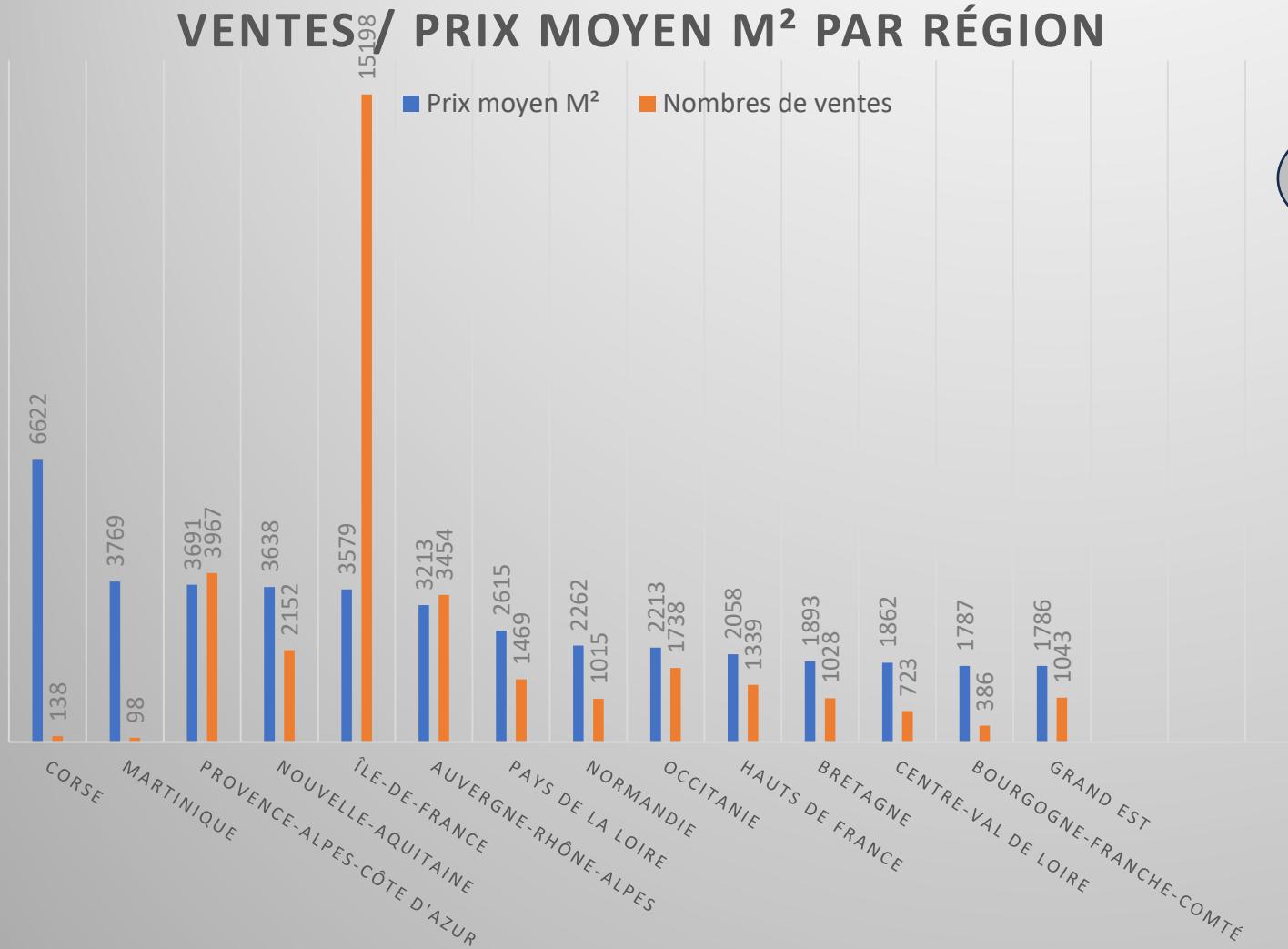
Requête n°10

```

Use LaplaceImmo;
WITH prix_T2 AS (
    SELECT AVG(v.valeur_fonciere / b.surface_reelle) AS avg_t2
    FROM ventes v
    JOIN biens b ON v.id_bien = b.id_bien
    WHERE b.id_type_local IN (Select id_type_local from type_local where libelle_type_local Like 'Appartement%')
        AND b.nombre_pieces_principales = 2
        AND b.surface_reelle > 10
        AND v.valeur_fonciere > 0
        AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
),
prix_T3 AS (
    SELECT AVG(v.valeur_fonciere / b.surface_reelle) AS avg_t3
    FROM ventes v
    JOIN biens b ON v.id_bien = b.id_bien
    WHERE b.id_type_local IN (Select id_type_local from type_local where libelle_type_local Like 'Appartement%')
        AND b.nombre_pieces_principales = 3
        AND b.surface_reelle > 10
        AND v.valeur_fonciere > 0
        AND v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
)
SELECT
    ROUND(avg_t2, 2) AS prix_m2_T2,
    ROUND(avg_t3, 2) AS prix_m2_T3,
    ROUND(((avg_t2 - avg_t3) / avg_t3) * 100, 2) AS difference_pourcentage
FROM prix_T2, prix_T3;

```

	prix_m2_T2	prix_m2_T3	difference_pourcentage
▶	4927.71	4288.73	14.90

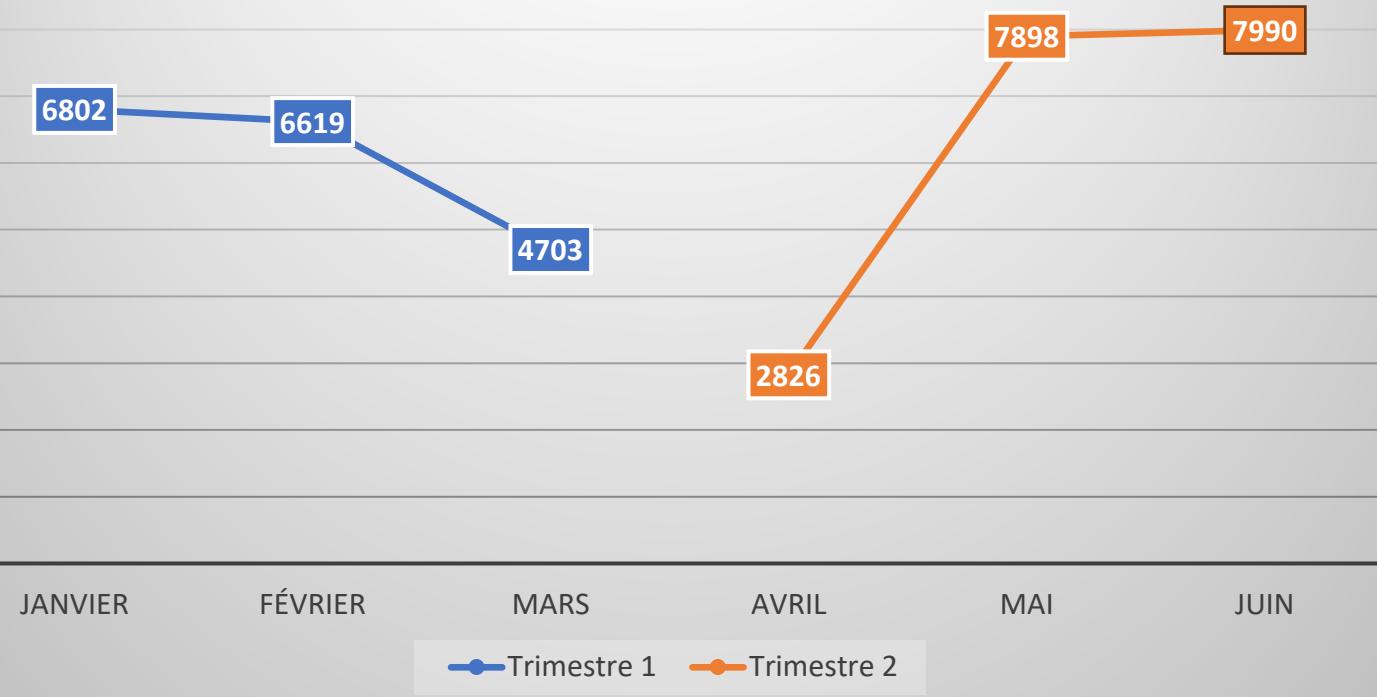


Requête n°7

```
Use LaplaceImmo;
WITH stats_trimestres AS (
    SELECT
        SUM(CASE WHEN MONTH(date_mutation) BETWEEN 1 AND 3 THEN 1 ELSE 0 END) AS ventes_Q1,
        SUM(CASE WHEN MONTH(date_mutation) BETWEEN 4 AND 6 THEN 1 ELSE 0 END) AS ventes_Q2
    FROM ventes
    WHERE YEAR(date_mutation) = 2020
)
SELECT
    ventes_Q1,
    ventes_Q2,
    ROUND((ventes_Q2 - ventes_Q1) / ventes_Q1 * 100, 2) AS taux_evolution_pourcentage
FROM stats_trimestres;
    
```

	ventes_Q1	ventes_Q2	taux_evolution_pourcentage
▶	18124	18714	3.26

Évolution des ventes premier semestre 2020



Requête n°9

```
Use LaplaceImmo;
SELECT
    c.com_nom,
    b.code_insee,
    COUNT(v.id_ventes) AS nombre_de_ventes
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN communes c ON b.code_insee = c.code_insee
WHERE v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
GROUP BY c.com_nom, b.code_insee
HAVING nombre_de_ventes >= 50
ORDER BY nombre_de_ventes DESC;
```

	com_nom	code_insee	nombre_de_ventes
▶	Saint-Pair	14640	158
	La Caure	51100	145
	Saint-Césaire-de-Gauzignan	30240	124
	Saint-Ferme	33400	115
	Anse	69009	107
	Saint-Léonard	76600	106
	Pézènes-les-Mines	34200	106
	Irigny	69100	104
	Affoux	69001	98
	Le Châtelet-en-Brie	77100	91
	Albigny-sur-Saône	69003	89
	Aurons	13008	88
	Villeneuve-la-Comtal	11430	88
	Corbières	11100	87

Requête n°6

```

SELECT
    v.valeur_fonciere,
    b.surface_reelle AS surface_m2,
    CASE
        WHEN d.reg_code = '11' THEN 'Île-de-France'
        WHEN d.reg_code = '93' THEN 'Provence-Alpes-Côte d'Azur'
        WHEN d.reg_code = '84' THEN 'Auvergne-Rhône-Alpes'
        WHEN d.reg_code = '76' THEN 'Occitanie'
        WHEN d.reg_code = '75' THEN 'Nouvelle-Aquitaine'
        ELSE CONCAT('Région ', d.reg_code)
    END AS region_nom,
    b.code_insee
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN departements d ON d.dep_code =
CASE
    WHEN TRIM(b.code_insee) LIKE '97%' THEN LEFT(TRIM(b.code_insee), 3)
    ELSE LEFT(TRIM(b.code_insee), 2)
END
)
Where b.id_type_local In (Select id_type_local from type_local where libelle_type_local Like 'Appartement%')
and b.surface_reelle > 10
and v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
ORDER BY v.valeur_fonciere DESC
LIMIT 10;
    
```

	valeur_fonciere	surface_m2	region_nom	code_insee
▶	8600000.00	62.00	Île-de-France	91100
	8577713.00	289.00	Île-de-France	75007
	7620000.00	42.00	Île-de-France	75017
	7600000.00	200.00	Île-de-France	75006
	7535000.00	143.00	Île-de-France	75001
	7420000.00	357.00	Île-de-France	75016
	7200000.00	241.00	Île-de-France	75016
	7050000.00	310.00	Île-de-France	75001
	6600000.00	76.00	Île-de-France	75001
	6500000.00	287.00	Île-de-France	75016

Requête n°11

```

Use LaplaceImmo;
WITH ClassementCommunes AS (
    SELECT
        d.dep_nom,
        c.com_nom,
        AVG(v.valeur_fonciere) AS moyenne_valeur_fonciere,
        DENSE_RANK() OVER (PARTITION BY d.dep_code ORDER BY AVG(v.valeur_fonciere) DESC) as rang
    FROM ventes v
    JOIN biens b ON v.id_bien = b.id_bien
    JOIN departements d ON d.dep_code = CAST(LEFT(LPAD(TRIM(b.code_insee), 5, '0'), 2) AS UNSIGNED)
    JOIN communes c ON b.code_insee = c.code_insee
    WHERE d.dep_code IN [6, 13, 33, 59, 69]
        and v.date_mutation BETWEEN '2020-01-01' AND '2020-06-30'
    GROUP BY d.dep_nom, d.dep_code, c.com_nom
)
SELECT
    dep_nom,
    com_nom,
    ROUND(moyenne_valeur_fonciere, 2) AS valeur_moyenne
FROM ClassementCommunes
WHERE rang <= 3
ORDER BY dep_nom, valeur_moyenne DESC;
    
```

	dep_nom	com_nom	valeur_moyenne
▶	Alpes-Maritimes	La Turbie	349918.33
	Alpes-Maritimes	Villeneuve-d'Entraunes	287205.08
	Alpes-Maritimes	Revest-les-Roches	211024.06
	Bouches-du-Rhône	Carnoux-en-Provence	314425.00
	Bouches-du-Rhône	Auriol	218161.08
	Bouches-du-Rhône	Marseille 10e Arrondissement	205000.00
	Gironde	Cazalis	705000.00
	Gironde	Cérons	307435.93
	Gironde	Saint-Pey-de-Castets	285308.00
	Nord	Fournes-en-Woëvre	322250.00
	Nord	Cartignies	306500.00
	Nord	Mousquier-en-Fagne	299232.22
	Rhône	Amblepuis	449348.03
	Rhône	Aigueperse	445746.36
	Rhône	Alix	369642.84

Requête n°12

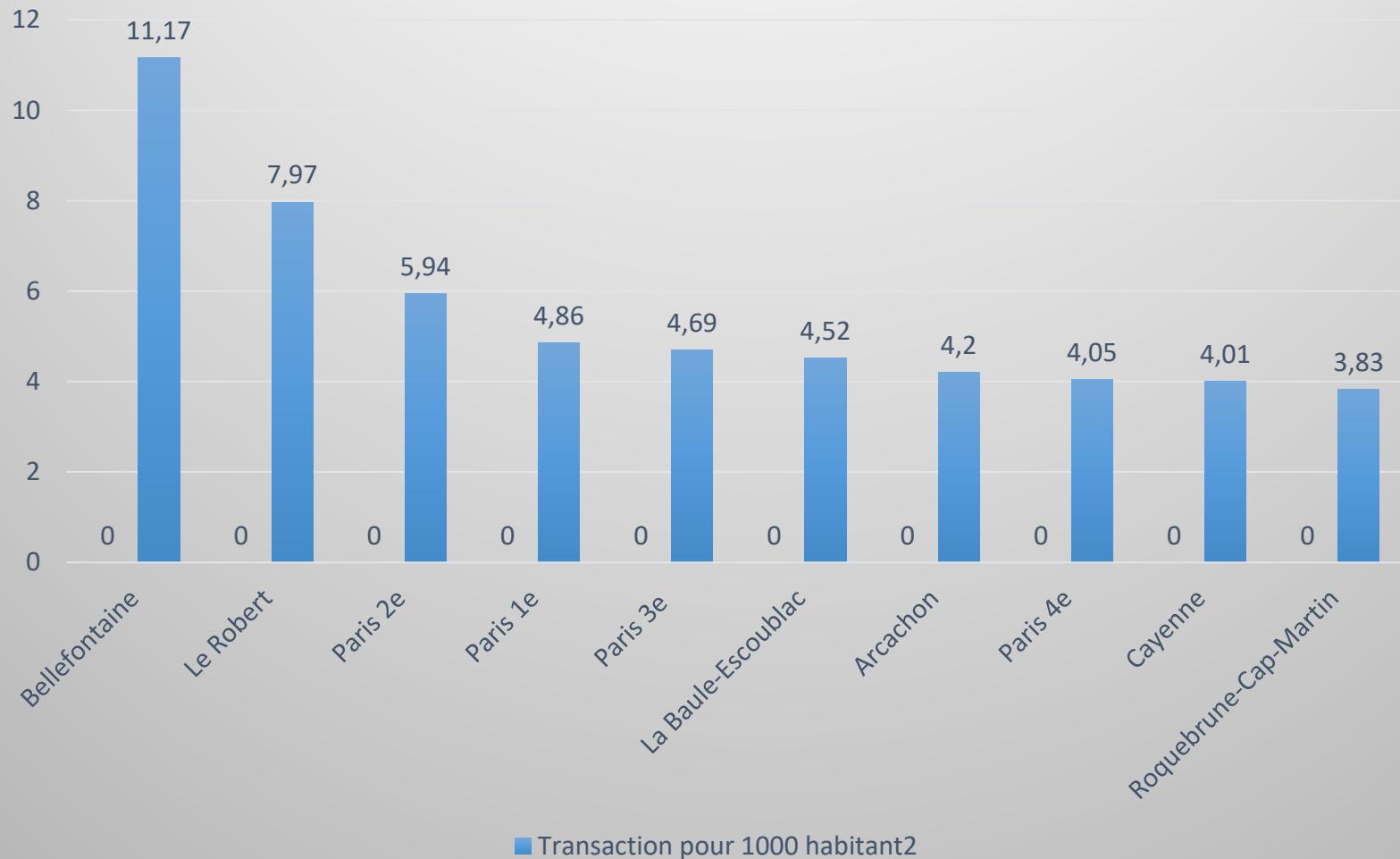
```

SELECT
    c.nom_com,
    c.dep_code,
    c.population_tot AS Population,
    COUNT(v.id_vente) AS Nb_Ventes,
    -- Calcul du ratio pour 1000 habitants
    ROUND((COUNT(v.id_vente) / c.population_tot) * 1000, 2) AS Transactions_pour_1000_hab
FROM ventes v
JOIN biens b ON v.id_bien = b.id_bien
JOIN communes c ON LPAD(b.code_insee, 5, '0') = LPAD(c.code_insee, 5, '0')
WHERE c.population_tot >= 10000
GROUP BY c.nom_com, c.dep_code, c.population_tot
ORDER BY Transactions_pour_1000_hab DESC
LIMIT 20;

```

	nom_com	dep_code	Population	Nb_Ventes	Transactions_pour_1000_hab
▶	Paris 2e Arrondissement	75	21735	127	5.84
	Paris 1er Arrondissement	75	16055	78	4.86
	Paris 3e Arrondissement	75	34306	161	4.69
	Arcachon	33	11898	55	4.62
	La Baule-Escoublac	44	16797	77	4.58
	Paris 4e Arrondissement	75	29390	119	4.05
	Roquebrune-Cap-Martin	06	13041	52	3.99
	Paris 8e Arrondissement	75	36250	139	3.83
	Sanary-sur-Mer	83	17160	60	3.50
	Paris 9e Arrondissement	75	60563	211	3.48
	La Londe-les-Maures	83	10776	37	3.43
	Paris 6e Arrondissement	75	41171	139	3.38
	Saint-Cyr-sur-Mer	83	11725	37	3.16
	Chantilly	60	11178	35	3.13
	Pornichet	44	11440	35	3.06
	Saint-Mandé	94	22576	69	3.06
	Paris 10e Arrondissement	75	86863	264	3.04
	Menton	06	30981	91	2.94
	Saint-Hilaire-de-Riez	85	11501	33	2.87
	Vincennes	94	50230	141	2.81

Enfin on va utiliser un indicateur de tension du marché en convertissant le nombre de vente pour 1000 Habitants.





Cadre européen qui protège la vie privée et les données personnelles.



Confidentialité des données



Crédibilité de l'agence



Qualité de la base de données



Anonymisation des noms des propriétaires.



Séparation des tables temporaires des tables de productions.



Utilisation de types de données stricts



Transformation de données brutes en base fonctionnel et sécurisé.



Visualisation du marché immobilier 2020 permettant d'orienté la stratégie de l'agence.



Affinage des prédictions de prix en intégrant des données socio-économiques.