



Création et utilisation de la base de données

Maure Kévin



Laplace Immo

Contexte du projet

Projet: DATAImmo

- Modification de la base de données
- Ajout de nouvelles données
- Analyse des nouvelles données
- Mieux prévoir le prix de vente des biens immobiliers



Les données initiales

DONNÉES INITIALES- Valeurs foncières

CODE	SIGNIFICATION
No_voie	Numéro des rues
Type_de_voie	Plusieurs valeurs (rue, avenue, chemin, etc.)
Voie	Nom de la rue
Reference document	
1 Articles CGI	
2 Articles CGI	
3 Articles CGI	
4 Articles CGI	
5 Articles CGI	
No disposition	Numéro de la mutation
Date de mutation	Date de signature de l'acte
Nature de mutation	Nature de la mutation (vente, expropriation ...)
Valeur fonciere	Montant de la mutation (NET mais sans frais notaires)
B/T/Q	Indice de répétition
Code type de voie	Code correspondant à un type de voie
Code voie	Code Rivoli
Code ID commune	ID de la commune
Code postal	Code postal
Commune	Libellé de la commune
Code departement	Code du département
Code commune	Code de la commune

Préfixe de section	Référence cadastrale de la parcelle
Section	Référence cadastrale de la parcelle
No plan	Référence cadastrale de la parcelle
1er lot	ID 1er lot de copropriété
Surface Carrez du 1er lot	Surface du 1er lot de copropriété
2eme lot	ID 2eme lot de copropriété
Surface carrez du 2eme lot	Surface du 2er lot de copropriété
3eme lot	ID 3eme lot de copropriété
Surface Carrez du 3eme lot	Surface du 3er lot de copropriété
4eme lot	ID 4eme lot de copropriété
Surface Carrez du 4eme lot	Surface du 4er lot de copropriété
5eme lot	ID 5eme lot de copropriété
Surface Carrez du 5eme lot	Surface du 5er lot de copropriété
Nombre de lots	Nombre de lots copropriétés
Code type local	Code du type de local
Type local	Type de local: maison, appartement, local commercial ...
Identifiant local	
Surface reelle bati	Surface mesurée au sol entre les murs de l'ensemble
Nombres pieces principale	Nombres de pieces principales
Nature culture	Code nature culture
Nature culture speciale	Code nature culture speciale
Surface terrain	Contenance du terrain
Nom de l'acquireur	Nom de l'acquireur

Les données initiales

DONNÉES INITIALES - Référentiel géographique

CODE	SIGNIFICATION
reggrp_nom	Regroupement régional
reg_nom	Libellé région
reg_nom_old	Ancien libellé région (avant 2016)
aca_nom	Libellé de l'académie
dep_nom	Libellé du département
com_code	Code commune
com_code1	Code commune, arrondissements marseillais et Lyonnais agrégés
com_code2	Code commune, arrondissements parisien, marseillais et Lyonnais agrégés
com_id	Id commune
com_nom_maj_cou	Libellé court de la commune en majuscule
com_nom_maj	Libellé de la commune en majuscule
com_nom	Libellé de la commune
uu_code	Code de l'unité urbaine(2020)
uu_id	Id de l'unité urbaine (2020)
uucr_id	Id de l'unité urbaine ou de la commune rurale (2020)
uucr_nom	Libellé de l'unité urbaine ou de la commune rurale (2020)
ze_id	Id de la zone d'emploi
dep_code	Code département

dep_id	Id département
dep_nom_num	Libellé du département et code
dep_num_nom	Code du département et libellé
aca_code	Code académie
aca_id	Id académie
reg_code	Code région
reg_id	Id région
reg_code_old	Ancien code région (Avant 2016)
reg_id_old	Ancien Id région (Avant 2016)
fd_id	France détaillée
fr_id	France regroupée
fe_id	France entière
uu_id_99	Ancien Id unité urbaine (1999)
au_code	Code de l'aire urbaine
au_id	Id de l'aire urbaine
auc_id	Id de l'aire urbaine ou de la commune
auc_nom	Libellé de l'aire urbaine ou de la commune
uu_id_10	Ancien Id de l'unité urbaine (2010)
geolocalisation	latitude et longitude

Les données initiales

DONNÉES INITIALES - Données communes

CODE	SIGNIFICATION
CODREG	Code de la région
CODDEP	Code du département
CODARR	Code arrondissement
CODCAN	Code canton
CODCOM	Code de la commune
COM	Nom commune
PMUN	Population municipale
PCAP	Population comptée à part
PTOT	Population total

La stratégie de sauvegarde et la conformité RGPD

❑ Fichier: referentiel_geographique: 38916 lignes

- Suppression des colonnes inutiles:
reg_nom_old; aca_nom; com_code1; com_code2; com_id;
com_nom_maj_court; com_nom; uu_code; uu_id; uucr_id;
uucr_nom; ze_id; dep_id; dep_nom_num; dep_num_nom; aca_code;
aca_id; reg_id; reg_code_old; reg_id_old; fd_id; fr_id; fe_id;
uu_id_99; au_code; au_id; auc_id; auc_nom; uu_id_10;
geolocalisation
- Modification nom de colonne:
com_code -> dep_code_com_code
- Création nouvelle colonne: com_code
=DROITE(colonne dep_code_com_code,3)
- Vérification aucune valeur null
- Filtre pour régions unique: reg_code unique -> 19 lignes
- Création fichier csv: referentiel_geographique_reg
- Filtre pour départements unique: dep_code unique -> 109 lignes
- Création fichier csv: referentiel_geographique_dep
- Filtre pour communes unique: dep_code_com_code unique -> 38916 lignes
- Création fichier csv: referentiel_geographique_commune

La stratégie de sauvegarde et la conformité RGPD

❑ Fichier: valeurs_foncières: 34169 lignes

- Suppression des colonnes inutiles:
code service ch; reference document; 1 articles cgi; 2 articles cgi; 3 articles cgi; 4 articles cgi; 5 articles cgi; no disposition; nature mutation; prefixe de section; section; no plan; no volume; 1^{er} lot; 2eme lot; surface carrez du 2eme lot; code type local; identifiant local
- Création de la colonne: dep_code_com_code
=CONCAT(dep_code; com_code)
- Création de la colonne: id_acquereur
création colonne nom_acquereur_unique (11112 lignes)
et colonne id_acquereur_unique("nom" & ligne())
et colonne id_acquereur(=RECHERCHEV(V3;\$AA\$2:\$AB\$11113;2;0))
- Création de la colonne: id_bien

La stratégie de sauvegarde et la conformité RGPD

❑ Fichier: valeurs_foncieres: 34169 lignes

- Vérification aucune valeur null:
no_voie; btq; type_de_voie; nature_culture; nature_culture_speciale -> remplacer vide par « null »
surface_terrain -> remplacer vide par 0
- Suppression de lignes:
8<= surface_carrez < 500 (83 lignes en moins) (0,24 % des données)
0< valeur_fonciere et valeur_fonciere non null (18 lignes en moins) (0,05% des données)
- Filtre pour bien unique:
(no_voie; btq; type_de_voie; voie; code_dep; code_com) unique (30703 lignes)
- Création fichier csv: valeurs_foncieres_bien
- Filtre pour vente unique:
(no_voie; btq; type_de_voie; voie; code_dep; code_com; date_mutation; valeur_fonciere) unique -> 33921 lignes
- Création fichier csv: valeurs_foncieres_vente

La stratégie de sauvegarde et la conformité RGPD

❑ Fichier: donnees_population: 34991 lignes

- Suppression des colonnes inutiles:
codreg; codarr; codcan; com; ptot
- Création de la colonne: dep_code_com_code
=CONCAT(CODREG; CODCOM)
- Vérification aucune valeur null:
Pas de valeur null
- Création fichier csv: donnees_population 34991 lignes

L'extrait du dictionnaire des données

DICTIONNAIRE DES DONNÉES - Valeurs foncières

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	JUSTIFICATION UTILISATION
id_vente	ID dans la base de données de la table vente	Integer	NC	Elémentaire	Ne doit pas être nul, Primary Key	Auto Increment	Descrit de manière unique une vente
id_bien	ID dans la base de données de la table bien	Integer	NC	Elémentaire	Ne doit pas être nul, Primary Key	Auto Increment	Descrit de manière unique un bien
No_voie	Numéro des rues	Integer	NC	Elémentaire			Descrit de manière unique un bien
Type_de_voie	Plusieurs valeurs (rue, avenue, chemin, etc.)	Varchar	4	Elémentaire			Descrit de manière unique un bien
Voie	Nom de la rue	Varchar	100	Elémentaire			Descrit de manière unique un bien
Date de mutation	Date de signature de l'acte	Date	NC	Elémentaire	Format de la date (AAAA/MM/JJ)		Descrit de manière unique une vente
Valeur fonciere	Montant de la mutation (NET mais sans frais notaires)	DECIMAL	10,2	Elémentaire	Ne doit pas être nul; 0<=valeur_fonciere		Information utile pour l'étude
B/T/Q	Indice de répétition	Varchar	1	Elémentaire			Descrit de manière unique un bien
Code departement	Code du département	Integer	NC	Elémentaire	Ne doit pas être nul		Utiliser pour calculer la variable dep_code_com_code
Code commune	Code de la commune	Integer	NC	Elémentaire	Ne doit pas être nul		Utiliser pour calculer la variable dep_code_com_code
Surface Carrez du 1er lot	Surface du 1er lot de copropriété	DECIMAL	5,2	Elémentaire	Ne doit pas être nul; 0<=surface		Information utile pour l'étude
Nombre de lots	Nombre de lots copropriétés	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=nb_lots		Information utile pour l'étude
Type local	Type de local: maison, appartement, local commercial ...	Varchar	11	Elémentaire	Ne doit pas être nul		Information utile pour l'étude
Surface reelle bati	Surface mesurée au sol entre les murs de l'ensemble	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=surface		Information utile pour l'étude
Nombres pieces principal	Nombres de pieces principales	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=nb_pieces		Information utile pour l'étude
Nature culture	Code nature culture	Varchar	2	Elémentaire			Information utile pour l'étude
Nature culture speciale	Code nature culture speciale	Varchar	5	Elémentaire			Information utile pour l'étude
Surface terrain	Contenance du terrain	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=surface		Information utile pour l'étude
ID de l'acqureur	ID de l'acqureur	Varchar	45	Elémentaire	Ne doit pas être nul	"NOM" & numero unique	Information utile pour l'étude

L'extrait du dictionnaire des données

DICTIONNAIRE DES DONNÉES - Référentiel géographique

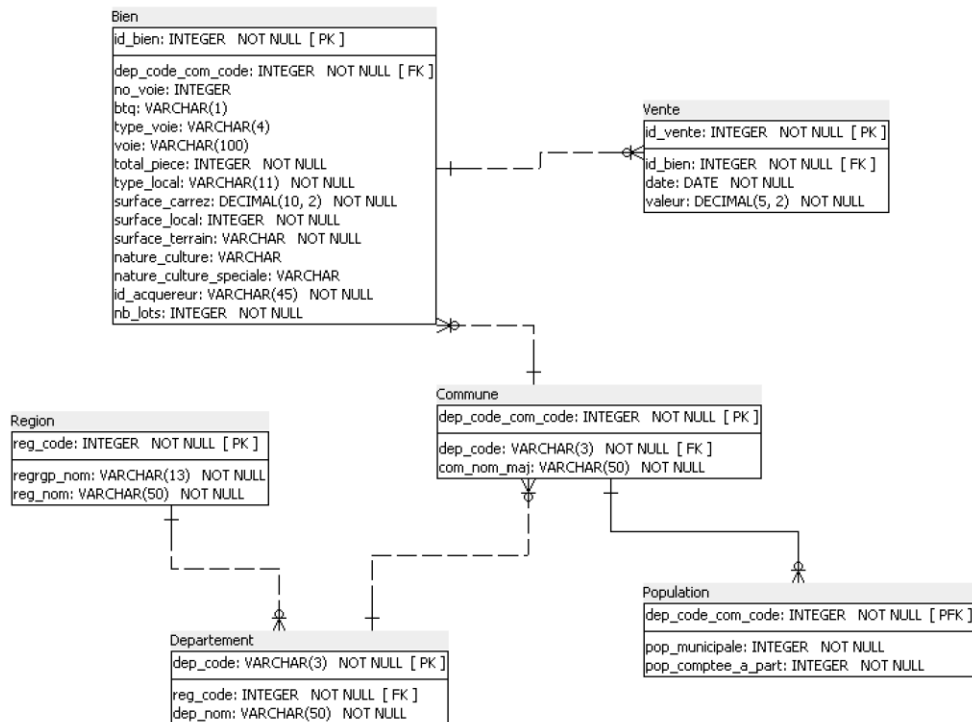
CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	JUSTIFICATION UTILISATION
reggrp_nom	Regroupement régional	Varchar	13	Elémentaire	Ne doit pas être nul		Information utile pour l'étude (table region)
reg_nom	Libellé région	Vachar	50	Elémentaire	Ne doit pas être nul		Information utile pour l'étude (table region)
dep_nom	Libellé du département	Vachar	50	Elémentaire	Ne doit pas être nul		Information utile pour l'étude (table departement)
com_code	Code commune	Vachar	6	Elémentaire	Ne doit pas être nul		Utiliser pour calculer la variable dep_code_com_code de la table commune
com_nom_maj	Libellé de la commune en majuscule	Varchar	50	Elémentaire	Ne doit pas être nul		Information utile pour l'étude
dep_code	Code département	Varchar	3	Elémentaire	Ne doit pas être nul; Primary Key		Décrit de manière unique un département
reg_code	Code région	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=reg_code<=94;Primary Key		Décrit de manière unique une région
dep_code_com_cod	Code département & code commune	Vachar	6	Concaténer	Ne doit pas être nul; Primary Key	dep_code & com_code	Décrit de manière unique une commune

L'extrait du dictionnaire des données

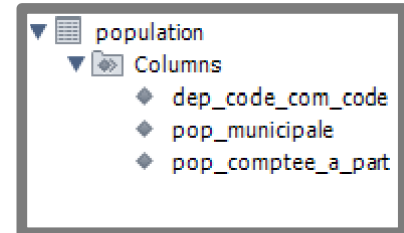
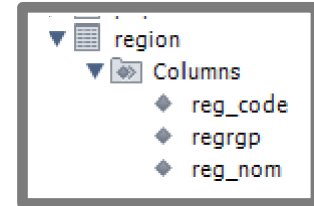
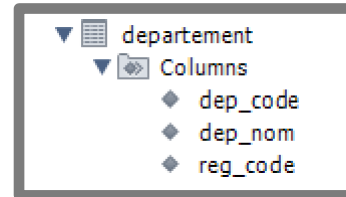
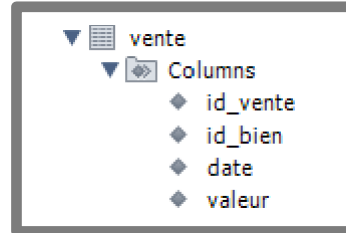
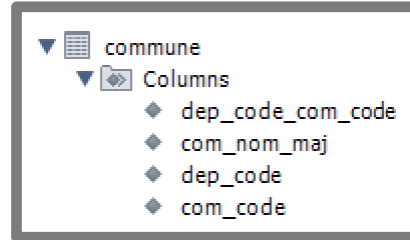
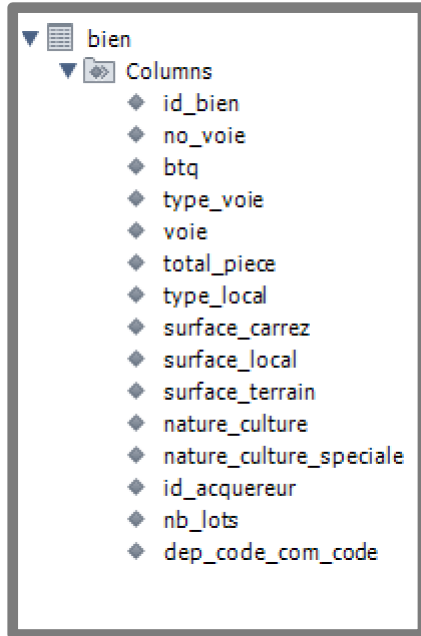
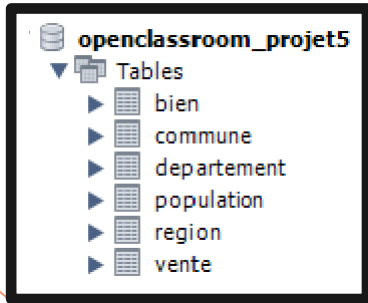
DICTIONNAIRE DES DONNÉES - Données communes

CODE	SIGNIFICATION	TYPE	LONGUEUR	NATURE	REGLE DE GESTION	REGLE DE CALCUL	JUSTIFICATION UTILISATION
CODDEP	Code du département	Vachar	3	Elémentaire	Ne doit pas être nul		Utiliser pour calculer la variable dep_code_com_code de la table population
CODCOM	Code de la commune	Integer	NC	Elémentaire	Ne doit pas être nul; non signé		Utiliser pour calculer la variable dep_code_com_code de la table population
PMUN	Population municipale	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=PMUN		Information utile
PCAP	Population comptée à part	Integer	NC	Elémentaire	Ne doit pas être nul; 0<=PCAP		Information utile
dep_code_com_code	Code département & code commune	Vachar	6	Concaténer	Ne doit pas être nul; Primary Key	CODDEP & CODCOM	Décrit de manière unique une population

Le schéma relationnel normalisé



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



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

```
136 • SELECT * FROM region;
```

```
137
```

Result Grid | Filter Rows: Edit:

	reg_code	regrgp	reg_nom
▶	0	DROM-COM	Collectivités d'outre-mer
	1	DROM-COM	Guadeloupe
	2	DROM-COM	Martinique
	3	DROM-COM	Guyane
	4	DROM-COM	La Réunion
	6	DROM-COM	Mayotte
	11	Ile-de-France	Ile-de-France
	24	Province	Centre-Val de Loire
	27	Province	Bourgogne-Franche-Comté
	28	Province	Normandie
	32	Province	Hauts-de-France
	44	Province	Grand Est
	52	Province	Pays de la Loire
	53	Province	Bretagne

Result 67 region 68 ×

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

```
136 • SELECT * FROM departement;  
137
```

Result Grid   Filter Rows:






	dep_code	dep_nom	reg_code
▶	1	Ain	84
	10	Aube	44
	11	Aude	76
	12	Aveyron	76
	13	Bouches-du-Rhône	93
	14	Calvados	28
	15	Cantal	84
	16	Charente	75
	17	Charente-Maritime	75
	18	Cher	24
	19	Corrèze	75
	2	Aisne	32
	21	Côte-d'Or	27
	22	Côtes-d'Armor	53

Result 71 [departement 72](#) ×

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

```
136 • SELECT * FROM commune;
```

```
137
```




Result Grid				Filter Rows: <input type="text"/>	Edit: 			Export
	dep_code_com_code	com_nom_maj		dep_code	com_code			
▶	01001	L'ABERGEMENT-CLEMENCIAT		1	1			
	01002	L'ABERGEMENT-DE-VAREY		1	2			
	01003	AMAREINS		1	3			
	01004	AMBERIEU-EN-BUGEY		1	4			
	01005	AMBERIEUX-EN-DOBES		1	5			
	01006	AMBLEON		1	6			
	01007	AMBRONAY		1	7			
	01008	AMBUTRIX		1	8			
	01009	ANDERT-ET-CONDON		1	9			
	01010	ANGLEFORT		1	10			
	01011	APREMONT		1	11			
	01012	ARANC		1	12			
	01013	ARANDAS		1	13			
	01014	ARRENT		1	14			

Result 73 commune 74 x

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

```
136 • SELECT * FROM population;
```

```
137
```

Result Grid   Filter Rows: <input type="text"/> Edit: 			
	dep_code_com_code	pop_municipale	pop_comtee_a_part
▶	01001	779	19
	01002	256	1
	01004	14134	380
	01005	1751	25
	01006	112	6
	01007	2800	115
	01008	762	15
	01009	326	9
	01010	1105	17
	01011	368	11
	01012	329	4
	01013	143	4
	01014	3349	93
	01015	650	17

Result 75 population 76 ×

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

136 • SELECT * FROM bien;

137

Result Grid																
Filter Rows:																
Edit:																
Export/Import:																
Wrap Cell Content:																
Fetch rows:																
id_bien	no_voie	btq	type_voie	voie	total_piece	type_local	surface_carrez	surface_local	surface_terrain	nature_culture	nature_culture_speciale	id_acquereur	nb_lots	dep_code_com_code		
1	3	NULL	PL	DU CHATEAU	3	Appartement	8.00	72	0	NULL	NULL	NOM5533	2	08409		
2	20	NULL	RUE	DE LA COMMUNE DE PARIS	1	Appartement	8.00	27	0	NULL	NULL	NOM1955	1	93001		
3	168	NULL	RUE	DU TEMPLE	1	Appartement	8.07	9	0	NULL	NULL	NOM2103	1	75103		
4	2	NULL	AV	GALOIS	1	Appartement	8.09	10	0	NULL	NULL	NOM4800	1	92014		
5	17	NULL	RUE	LOUIS BONNET	2	Appartement	8.10	31	0	NULL	NULL	NOM8766	1	75111		
6	3	NULL	RUE	JACOB	2	Appartement	8.20	20	0	NULL	NULL	NOM1240	1	75106		
7	9	NULL	RUE	DE LA VILLE L EVEQUE	1	Appartement	8.20	9	0	NULL	NULL	NOM2992	2	75108		
8	NULL	NULL	NULL	LA VILLE	1	Appartement	8.20	64	0	NULL	NULL	NOM6536	2	39013		
9	40	NULL	AV	DE LA REPUBLIQUE	1	Appartement	8.21	8	0	NULL	NULL	NOM9187	1	75111		
10	38	NULL	RUE	DAUPHINE	1	Appartement	8.24	9	0	NULL	NULL	NOM10287	1	75106		
11	15	NULL	AV	NIEL	1	Appartement	8.25	9	0	NULL	NULL	NOM543	1	75117		
12	12	NULL	RUE	MONGE	1	Appartement	8.30	10	0	NULL	NULL	NOM1367	1	75105		
13	4	NULL	RUE	MARCE DESPREZ	1	Appartement	8.36	10	0	NULL	NULL	NOM9434	1	38185		

Result 69 bien 70 x

Apply

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

```
136 • SELECT * FROM vente;
```

```
137
```

Result Grid



Filter Rows:

	id_vente	id_bien	date	valeur
▶	1	1	2020-05-20	39000.00
	2	2	2020-05-29	505000.00
	3	28	2020-03-20	510500.00
	4	55	2020-06-11	130000.00
	5	145	2020-05-20	95000.00
	6	217	2020-05-14	142000.00
	7	218	2020-05-20	113660.00
	8	219	2020-05-25	385360.00
	9	220	2020-05-29	108000.00
	10	221	2020-06-18	148800.00
	11	320	2020-02-19	117000.00
	12	321	2020-03-04	35000.00
	13	322	2020-03-17	60000.00
	14	323	2020-05-27	32500.00

Result 77

vente 78 x

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

136 • `SELECT COUNT(*) FROM commune;`

Result Grid | Filter Rows: | Export:

	COUNT(*)
▶	38916

136 • `SELECT COUNT(*) FROM region;`

Result Grid | Filter Rows: | Export:

	COUNT(*)
▶	19

136 • `SELECT COUNT(*) FROM departement;`

Result Grid | Filter Rows: | Export:

	COUNT(*)
▶	109

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

136 • `SELECT COUNT(*) FROM vente;`

Result Grid | Filter Rows:

	COUNT(*)
▶	33921

136 • `SELECT COUNT(*) FROM bien;`

137

Result Grid | Filter Rows:

	COUNT(*)
▶	30703

136 • `SELECT COUNT(*) FROM population;`

Result Grid | Filter Rows:

	COUNT(*)
▶	34991



Analyse de données

Requêtes SQL et résultats

Requête 1

Nombre total d'appartements vendus au 1^{er} semestre 2020

```
8      #1. Nombre total d'appartements vendus au 1er semestre 2020.
9      • SELECT COUNT(*) AS nombre_appartements_vendus
10     FROM vente v
11     JOIN bien b ON b.id_bien = v.id_bien
12     WHERE b.type_local LIKE "Appartement";
13
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	nombre_appartements_vendus
▶	31174

Requête 2

Le nombre de ventes d'appartement par région pour le 1^{er} semestre 2020

```
15 #2. Le nombre de ventes d'appartement par région pour le 1er semestre 2020.
16 • SELECT r.reg_nom AS region, COUNT(*) AS nombre_ventes
17 FROM vente v
18 JOIN bien b ON v.id_bien = b.id_bien
19 JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
20 JOIN departement d ON d.dep_code = c.dep_code
21 JOIN region r ON d.reg_code = r.reg_code
22 WHERE b.type_local LIKE "Appartement"
23 GROUP BY r.reg_nom
24 ORDER BY nombre_ventes DESC;
```

region	nombre_ventes
Ile-de-France	13902
Provence-Alpes-Côte d'Azur	3623
Auvergne-Rhône-Alpes	3234
Nouvelle-Aquitaine	1927
Occitanie	1632
Pays de la Loire	1349
Hauts-de-France	1249
Grand Est	978
Bretagne	977
Normandie	848
Centre-Val de Loire	691
Bourgogne-Franche-Comté	372
Corse	218
Martinique	94
La Réunion	44
Guyane	34
Guadeloupe	2

Result 15 x

Requête 3

Proportion des ventes d'appartements par le nombre de pièces

```
26  #3. Proportion des ventes d'appartements par le nombre de pièces.  
27  • SELECT B.total_piece AS nombre_pieces, ROUND(COUNT(*)/(SELECT COUNT(*) FROM vente)*100,2) AS pourcentage_nombre_ventes  
28  FROM Vente V  
29  JOIN Bien B ON V.id_bien = B.id_bien  
30  GROUP BY B.total_piece;  
31
```

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

	nombre_pieces	pourcentage_nombre_ventes
▶	3	27.11
	1	21.41
	2	30.38
	0	0.09
	4	15.02
	6	1.04
	10	0.01
	5	4.56
	9	0.04
	7	0.26
	11	0.00
	8	0.07

Requête 4

Liste des 10 départements où le prix du mètre carré est le plus élevé

```
34  #4. Liste des 10 départements où le prix du mètre carré est le plus élevé.
35  • SELECT d.dep_nom AS departement, ROUND(AVG(v.valeur / b.surface_carrez),2) AS prix_moyen_metre_carre
36  FROM vente v
37  JOIN bien b ON v.id_bien = b.id_bien
38  JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
39  JOIN Departement d ON d.dep_code = c.dep_code
40  GROUP BY d.dep_nom
41  ORDER BY prix_moyen_metre_carre DESC
42  LIMIT 10;
```

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

	departement	prix_moyen_metre_carre
▶	Paris	12644.76
	Hauts-de-Seine	7585.08
	Val-de-Marne	5384.48
	Alpes-Maritimes	4870.93
	Seine-Saint-Denis	4461.41
	Yvelines	4376.47
	Haute-Savoie	4325.20
	Corse-du-Sud	4235.32
	Rhône	4155.03
	Haute-Corse	4016.63

Requête 5

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

```
45      #5. Prix moyen du mètre carré d'une maison en Île-de-France.  
46 •    SELECT AVG(v.valeur / b.surface_carrez) AS prix_moyen_metre_carre  
47      FROM vente v  
48      JOIN bien b ON v.id_bien = b.id_bien  
49      JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code  
50      JOIN departement d on d.dep_code = c.dep_code  
51      JOIN region r ON d.reg_code = r.reg_code  
52      WHERE r.reg_nom = 'Île-de-France' AND b.type_local = 'Maison';  
53  
54      /*
```

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

	prix_moyen_metre_carre
▶	3782.4808548552

Requête 6

Liste des 10 appartements les plus chers avec la région et le nombre de mètres carrés

#6. Liste des 10 appartements les plus chers avec la région et le nombre de mètres carrés.

```
SELECT b.surface_carrez AS surface, v.valeur AS prix, r.reg_nom AS region
FROM vente v
JOIN bien b ON v.id_bien = b.id_bien
JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
JOIN departement d on d.dep_code = c.dep_code
JOIN region r ON d.reg_code = r.reg_code
WHERE b.type_local = 'Appartement'
ORDER BY v.valeur DESC
```

	surface	prix	region
▶	9.10	9000000.00	Ile-de-France
	64.00	8600000.00	Ile-de-France
	20.55	8577713.00	Ile-de-France
	42.77	7620000.00	Ile-de-France
	253.30	7600000.00	Ile-de-France
	139.90	7535000.00	Ile-de-France
	360.95	7420000.00	Ile-de-France
	122.56	7050000.00	Ile-de-France
	25.25	6600000.00	Ile-de-France
	317.05	6500000.00	Ile-de-France

Requête 7

Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020

```
66
67     #7. Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020.
68 • ○ SELECT ROUND((SUM(CASE WHEN EXTRACT(MONTH FROM date) > 3 THEN 1 ELSE 0 END)
69     - SUM(CASE WHEN EXTRACT(MONTH FROM date) <= 3 THEN 1 ELSE 0 END)) * 100.0
70     / SUM(CASE WHEN EXTRACT(MONTH FROM date) <= 3 THEN 1 ELSE 0 END),2) AS taux_evolution
71     FROM Vente;
72
73
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:

	taux_evolution
▶	3.60

Requête 8

Le classement des régions par rapport au prix au mètre carré des appartement de plus de 4 pièces

#8. Le classement des régions par rapport au prix au mètre carré des appartement de plus de 4 pièces.

```
SELECT r.reg_nom AS region, ROUND(AVG(v.valeur / b.surface_carrez),2) AS prix_moyen_metre_carre
FROM vente v
JOIN bien b ON v.id_bien = b.id_bien
JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
JOIN departement d ON d.dep_code = c.dep_code
JOIN region r ON d.reg_code = r.reg_code
WHERE b.type_local = 'Appartement' AND b.total_piece > 4
GROUP BY r.reg_nom
ORDER BY prix_moyen_metre_carre DESC;
```

region	prix_moyen_metre_carre
Ile-de-France	9262.99
Provence-Alpes-Côte d'Azur	3874.11
La Réunion	3641.81
Corse	3104.88
Auvergne-Rhône-Alpes	2958.66
Bretagne	2548.08
Nouvelle-Aquitaine	2438.93
Hauts-de-France	2376.30
Pays de la Loire	2305.18
Occitanie	2141.71
Normandie	1965.75
Grand Est	1542.91
Centre-Val de Loire	1480.16
Bourgogne-Franche-Comté	1238.86
Martinique	573.48

Requête 9

Liste des communes ayant eu au moins 50 ventes au 1^{er} trimestre

#9. Liste des communes ayant eu au moins 50 ventes au 1er trimestre

```
SELECT c.com_nom_maj AS commune, COUNT(*) AS nombre_ventes
FROM vente v
JOIN bien b ON v.id_bien = b.id_bien
JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
WHERE EXTRACT(MONTH FROM date) <= 3
GROUP BY c.com_nom_maj
HAVING COUNT(*) >= 50;
```


Requête 9

Liste des communes ayant eu au moins 50 ventes au 1^{er} trimestre

commune	nombre_ventes
ANTIBES	77
NICE	171
LA CIOTAT	62
MARSEILLE 1ER ARRONDISSEMENT	71
MARSEILLE 4E ARRONDISSEMENT	72
MARSEILLE 8E ARRONDISSEMENT	81
MARSEILLE 9E ARRONDISSEMENT	63
AJACCIO	54
NIMES	63
TOULOUSE	78
BORDEAUX	156
SETE	62
RENNES	61
GRENOBLE	106
NANTES	118
ANGERS	64
LILLE	67
PARIS 2E ARRONDISSEMENT	61
PARIS 3E ARRONDISSEMENT	77
PARIS 4E ARRONDISSEMENT	59
PARIS 5E ARRONDISSEMENT	76
PARIS 6E ARRONDISSEMENT	85
PARIS 7E ARRONDISSEMENT	87
PARIS 8E ARRONDISSEMENT	62
PARIS 9E ARRONDISSEMENT	103

Result 34

commune	nombre_ventes
PARIS 9E ARRONDISSEMENT	103
PARIS 10E ARRONDISSEMENT	109
PARIS 11E ARRONDISSEMENT	166
PARIS 12E ARRONDISSEMENT	106
PARIS 13E ARRONDISSEMENT	94
PARIS 14E ARRONDISSEMENT	145
PARIS 15E ARRONDISSEMENT	214
PARIS 16E ARRONDISSEMENT	164
PARIS 17E ARRONDISSEMENT	225
PARIS 18E ARRONDISSEMENT	208
PARIS 19E ARRONDISSEMENT	115
PARIS 20E ARRONDISSEMENT	125
VERSAILLES	54
TOULON	58
ASNIERES-SUR-SEINE	80
BOULOGNE-BILLANCOURT	98
COURBEVOIE	80
ISSY-LES-MOULINEAUX	50
LEVALLOIS-PERRET	59
PUTEAUX	53
RUEIL-MALMAISON	68
MONTREUIL	65
SAINT-MAUR-DES-FOSSES	55
VINCENNES	68

Requête 10

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

```
21
98  #10. Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces.
99  • SELECT ROUND(((AVG(CASE WHEN b.total_piece = 2 THEN v.valeur / b.surface_carrez END)
100    / AVG(CASE WHEN b.total_piece = 3 THEN v.valeur / b.surface_carrez END)) -1)
101    * 100,2) AS difference_pourcentage
102  FROM vente v
103  JOIN bien b ON v.id_bien = b.id_bien
104  WHERE b.type_local = 'Appartement' AND b.total_piece IN (2, 3);
105
106
```

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

	difference_pourcentage
▶	20.51

Requête 11

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

```
105      #11. Les moyennes de valeurs foncières pour le top 3 des communes des départements 6, 13, 33, 59 et 69.
106 •    SELECT dep_code, commune, moyenne_valeur_fonciere
107 FROM (
108     SELECT c.dep_code, c.com_nom_maj AS commune, AVG(v.valeur) AS moyenne_valeur_fonciere,
109            ROW_NUMBER() OVER (PARTITION BY c.dep_code ORDER BY AVG(v.valeur) DESC) AS row_num
110     FROM vente v
111     JOIN bien b ON v.id_bien = b.id_bien
112     JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
113     WHERE c.dep_code IN (6, 13, 33, 59, 69)
114     GROUP BY c.dep_code, c.com_nom_maj
115 ) AS ranked_communes
116 WHERE row_num <= 3;
117
```

Requête 11

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

dep_code	commune	moyenne_valeur_fonciere
13	GIGNAC-LA-NERTHE	330000.000000
13	SAINT-SAVOURNIN	314425.000000
13	CASSIS	313416.875000
33	LEGE-CAP-FERRET	549500.636364
33	VAYRES	335000.000000
33	ARCACHON	307435.927273
59	BERSEE	433202.000000
59	CYSOING	408550.000000
59	HALLUIN	322250.000000
6	SAINT-JEAN-CAP-FERRAT	968750.000000
6	EZE	655000.000000
6	MOUANS-SARTOUX	476898.100000
69	VILLE-SUR-JARNIOUX	485300.000000
69	LYON 2E ARRONDISSEM...	459583.453731
69	LYON 6E ARRONDISSEM...	426968.250000

Requête 12

Les 20 communes avec le plus de transactions pour 1000 habitants pour les communes qui dépassant les 10000 habitants

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

```
SELECT c.com_nom_maj AS commune, ROUND(COUNT(*) * 1000.0 / p.pop_municipale,2) AS transactions_par_1000_habitants
FROM vente v
JOIN bien b ON v.id_bien = b.id_bien
JOIN commune c ON c.dep_code_com_code = b.dep_code_com_code
JOIN population p ON p.dep_code_com_code = c.dep_code_com_code
WHERE p.pop_municipale > 10000
GROUP BY c.com_nom_maj, p.pop_municipale
ORDER BY transactions_par_1000_habitants DESC
LIMIT 20;
```

Requête 12

Les 20 communes avec le plus de transactions pour 1000 habitants
pour les communes qui dépassant les 10000 habitants

commune	transactions_par_1000_habitants
PARIS 2E ARRONDISSEMENT	5.79
PARIS 1ER ARRONDISSEMENT	4.84
ARCACHON	4.73
LA BAULE-ESCOUBLAC	4.68
PARIS 3E ARRONDISSEMENT	4.64
ROQUEBRUNE-CAP-MARTIN	4.05
PARIS 4E ARRONDISSEMENT	4.05
PARIS 8E ARRONDISSEMENT	3.81
SANARY-SUR-MER	3.55
LA LONDE-LES-MAURES	3.48
PARIS 6E ARRONDISSEMENT	3.42
PARIS 9E ARRONDISSEMENT	3.40
SAINT-CYR-SUR-MER	3.22
CHANTILLY	3.20
PORNICHET	3.14
SAINT-MANDE	3.04
PARIS 10E ARRONDISSEMENT	3.02
MENTON	2.98
SAINT-HILAIRE-DE-RIEZ	2.92
VINCENNES	2.83



Merci !