

## Nombre total d'appartements vendus au 1<sup>er</sup> semestre 2020

Laplace\_Immo/postgres@PostgreSQL 15

Query Query History

```

1 SELECT COUNT(*) AS Nb_appt_vendus_1erSem2020
2 FROM vente v
3 JOIN bien b ON b.id_bien = v.id_bien
4 WHERE b.type_local = 'Appartement' AND v.date_vente BETWEEN '2020-01-01' AND '2020-06-30';

```

Data Output Messages Notifications

	nb_appt_vendus_1ersem2020 bigint
1	31378

Total rows: 1 of 1    Query complete 00:00:00.117



# Le nombre de ventes d'appartement par région pour le 1<sup>er</sup> semestre 2020

Laplace\_Immo/postgres@PostgreSQL 15

No limit

Query

Query History

```

1 SELECT
2     r.nom_region,
3     COUNT(v.id_vente) AS Ventes_apt_S1_2020
4 FROM vente v
5 JOIN bien b ON b.id_bien = v.id_bien
6 JOIN departement d ON d.code_departement = b.code_departement
7 JOIN region r ON r.code_region = d.code_region
8 WHERE b.type_local = 'Appartement' AND v.date_vente BETWEEN '2020-01-01' AND '2020-06-30'
9 GROUP BY r.nom_region
10 ORDER BY ventes_apt_S1_2020 DESC;

```

Data Output

Messages

Notifications

	nom_region character varying (50)	ventes_apt_s1_2020 bigint
1	Ile-de-France	13995
2	Provence-Alpes-Côte d'Azur	3649
3	Auvergne-Rhône-Alpes	3253
4	Nouvelle-Aquitaine	1932
5	Occitanie	1640
6	Pays de la Loire	1357
7	Hauts-de-France	1254
8	Grand Est	984
9	Bretagne	983
10	Normandie	862
11	Centre-Val de Loire	696
12	Bourgogne-Franche-Comté	376
13	Corse	223
14	Martinique	94
15	La Réunion	44
16	Guyane	34
17	Guadeloupe	2

Total rows: 17 of 17

Query complete 00:00:00.237





# Proportion des ventes d'appartements par le nombre de pièces

Laplace\_Immo/postgres@PostgreSQL 15

No limit

Query Query History

1 SELECT

2     b.nombre\_pieces,

3     ROUND((COUNT(v.id\_vente)/

4         CAST ((SELECT COUNT (\*)

5             FROM vente v

6             JOIN bien b ON b.id\_bien = v.id\_bien

7             WHERE b.type\_local = 'Appartement') AS NUMERIC) \* 100),2) AS "Proportion ventes (en %)"

8 FROM vente v

9 JOIN bien b ON b.id\_bien = v.id\_bien

10 WHERE b.type\_local = 'Appartement'

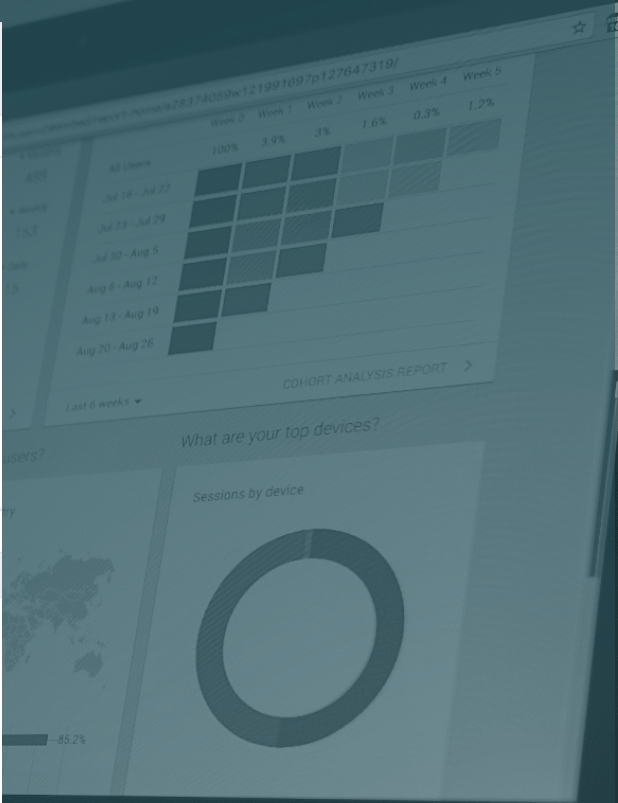
11 GROUP BY b.nombre\_pieces

12 ORDER BY b.nombre\_pieces;

Data Output Messages Notifications

	nombre_pieces integer	Proportion ventes (en %) numeric
1	0	0.10
2	1	21.48
3	2	31.18
4	3	28.57
5	4	14.21
6	5	3.55
7	6	0.65
8	7	0.17
9	8	0.05
10	9	0.03
11	10	0.01
12	11	0.00

Total rows: 12 of 12    Query complete 00:00:00.091



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

Laplace\_Immo/postgres@PostgreSQL 15

Query

Query History

```
1 SELECT
2   d.code_departement,
3   d.nom_departement,
4   ROUND(CAST(AVG(v.valeur/b.surface_carrez) AS NUMERIC),2) AS Prix_m2
5 FROM vente v
6 JOIN bien b ON b.id_bien = v.id_bien
7 JOIN departement d ON d.code_departement = b.code_departement
8 GROUP BY d.code_departement, d.nom_departement
9 ORDER BY Prix_m2 DESC
10 LIMIT 10;
```

Data Output

Messages

Notifications

	code_departement [PK] character varying (3)	nom_departement character varying (50)	prix_m2 numeric
1	75	Paris	12052.89
2	92	Hauts-de-Seine	7219.39
3	94	Val-de-Marne	5343.28
4	06	Alpes-Maritimes	4700.33
5	74	Haute-Savoie	4667.13
6	93	Seine-Saint-Denis	4344.78
7	78	Yvelines	4225.25
8	69	Rhône	4059.31
9	2A	Corse-du-Sud	4026.97
10	33	Gironde	3764.14

Total rows: 10 of 10

Query complete 00:00:00.182





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

Laplace\_Immo/postgres@PostgreSQL 15

No limit

Query

Query History

1

SELECT

2

r.nom\_region,

3

ROUND(CAST(AVG(v.valeur/b.surface\_carrez) AS NUMERIC),2) AS Prix\_m2

4

FROM vente v

5

JOIN bien b ON b.id\_bien = v.id\_bien

6

JOIN departement d ON d.code\_departement = b.code\_departement

7

JOIN region r ON r.code\_region = d.code\_region

8

WHERE b.type\_local = 'Maison' AND r.nom\_region = 'Île-de-France'

9

GROUP BY r.nom\_region;

Data Output

Messages

Notifications

	nom_region character varying (50)	prix_m2 numeric
1	Île-de-France	3745.01

Total rows: 1 of 1    Query complete 00:00:00.078



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

Laplace\_Immo/postgres@PostgreSQL 15

No limit

Query

Query History

```

1  SELECT
2      b.id_bien,
3      b.surface_carrez,
4      r.nom_region,
5      CAST(v.valeur AS NUMERIC) AS Prix_Apt
6  FROM bien b
7  JOIN vente v ON v.id_bien = b.id_bien
8  JOIN departement d ON d.code_departement = b.code_departement
9  JOIN region r ON r.code_region = d.code_region
10 WHERE b.type_local = 'Appartement' AND v.valeur IS NOT NULL
11 ORDER BY v.valeur DESC
12 LIMIT 10;

```

Data Output

Messages

Notifications

	id_bien integer	surface_carrez real	nom_region character varying (50)	prix_apt numeric
1	32250	9.1	Ile-de-France	9000000
2	21815	64	Ile-de-France	8600000
3	29776	20.55	Ile-de-France	8577710
4	32408	42.77	Ile-de-France	7620000
5	29827	253.3	Ile-de-France	7600000
6	29499	139.9	Ile-de-France	7535000
7	31948	360.95	Ile-de-France	7420000
8	32110	595	Ile-de-France	7200000
9	29330	122.56	Ile-de-France	7050000
10	29490	79.38	Ile-de-France	6600000

Total rows: 10 of 10

Query complete 00:00:00.157





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

Laplace\_Immo/postgres@PostgreSQL 15

Query Query History

```
1 WITH
2   ventes_t1 AS (SELECT COUNT(*) AS Nb_ventes_trimestre1 FROM vente WHERE date_vente BETWEEN '2020-01-01' AND '2020-03-31'),
3   ventes_t2 AS (SELECT COUNT(*) AS Nb_ventes_trimestre2 FROM vente WHERE date_vente BETWEEN '2020-04-01' AND '2020-06-30')
4 SELECT
5   Nb_ventes_trimestre1,
6   Nb_ventes_trimestre2,
7   ROUND(CAST((Nb_ventes_trimestre2 - Nb_ventes_trimestre1) / CAST (Nb_ventes_trimestre1 AS FLOAT)) AS NUMERIC)* 100,2) AS "Evolution du nombre de ventes (en %)"
8 FROM ventes_t1, ventes_t2;
```

Data Output Messages Notifications

	nb_ventes_trimestre1 bigint	nb_ventes_trimestre2 bigint	Evolution du nombre de ventes (en %) numeric
1	16776	17393	3.68

Total rows: 1 of 1 Query complete 00:00:00.083

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

Laplace\_Immo/postgres@PostgreSQL 15

Query

Query History

```
1 SELECT
2     r.nom_region,
3     ROUND(CAST(AVG(v.valeur/b.surface_carrez) AS NUMERIC),2) AS Prix_m2
4 FROM bien b
5 JOIN vente v ON v.id_bien = b.id_bien
6 JOIN departement d ON d.code_departement = b.code_departement
7 JOIN region r ON r.code_region = d.code_region
8 WHERE b.type_local = 'Appartement' AND b.nombre_pieces > 4
9 GROUP BY r.nom_region
10 ORDER BY Prix_m2 DESC;
```

Data Output

Messages

Notifications

	nom_region character varying (50)	prix_m2 numeric
1	Ile-de-France	8770.44
2	La Réunion	3641.81
3	Provence-Alpes-Côte d'Azur	3587.65
4	Corse	3104.88
5	Auvergne-Rhône-Alpes	2891.38
6	Nouvelle-Aquitaine	2465.48
7	Bretagne	2412.05
8	Pays de la Loire	2315.76
9	Hauts-de-France	2189.93
10	Occitanie	2097.23
11	Normandie	2015.77
12	Grand Est	1540.89
13	Centre-Val de Loire	1453.11
14	Bourgogne-Franche-Comté	1251.19
15	Martinique	573.48

Total rows: 15 of 15

Query complete 00:00:00.212





Laplace/Immo/postgres@PostgreSQL 15

No limit

Query

Query History

```

1 SELECT
2     c.nom_commune,
3     COUNT(v.id_vente) AS Nb_ventes
4 FROM vente v
5 JOIN bien b ON b.id_bien = v.id_bien
6 JOIN commune c ON (c.code_departement = b.code_departement AND c.code_commune = b.code_commune
7 WHERE v.date_vente BETWEEN '2020-01-01' AND '2020-03-31'
8 GROUP BY c.nom_commune
9 HAVING COUNT(v.id_vente) >=50
10 ORDER BY Nb_ventes DESC;

```

Data Output

Messages

Notifications

	nom_commune character varying (70)	nb_ventes bigint
1	Paris 17e Arrondissement	228
2	Paris 15e Arrondissement	215
3	Paris 18e Arrondissement	209
4	Nice	173
5	Paris 11e Arrondissement	169
6	Paris 16e Arrondissement	165
7	Bordeaux	157
8	Paris 14e Arrondissement	146
9	Paris 20e Arrondissement	127
10	Nantes	119
11	Paris 19e Arrondissement	116
12	Paris 12e Arrondissement	110
13	Paris 10e Arrondissement	109
14	Paris 9e Arrondissement	106
15	Grenoble	106
16	Boulogne-Billancourt	99
17	Paris 13e Arrondissement	94
18	Paris 7e Arrondissement	87

Total rows: 48 of 48

Query complete 00:00:00.165



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

Laplace\_Immo/postgres@PostgreSQL 15

Query

Query History

```
1 WITH
2   apt_2p AS (SELECT ROUND(CAST(AVG(v.valeur/b.surface_carrez) AS NUMERIC),2) AS Prix_m2_t2
3               FROM vente v
4               JOIN bien b ON b.id_bien = v.id_bien
5               WHERE b.type_local = 'Appartement' AND b.nombre_pieces = 2),
6   apt_3p AS (SELECT ROUND(CAST(AVG(v.valeur/b.surface_carrez) AS NUMERIC),2) AS Prix_m2_t3
7               FROM vente v
8               JOIN bien b ON b.id_bien = v.id_bien
9               WHERE b.type_local = 'Appartement' AND b.nombre_pieces = 3)
10 SELECT
11   Prix_m2_t2,
12   Prix_m2_t3,
13   ROUND(((Prix_m2_t3 - Prix_m2_t2)/Prix_m2_t2) * 100, 2) AS "Différence du prix au m2 (en %)"
14 FROM apt_2p, apt_3p;
```

Data Output

Messages

Notifications

	prix_m2_t2 numeric	prix_m2_t3 numeric	Différence du prix au m2 (en %) numeric
1	4908.86	4299.46	-12.41

Total rows: 1 of 1    Query complete 00:00:00.259





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

Laplace\_Immo/postgres@PostgreSQL 15

No limit

Query Query History

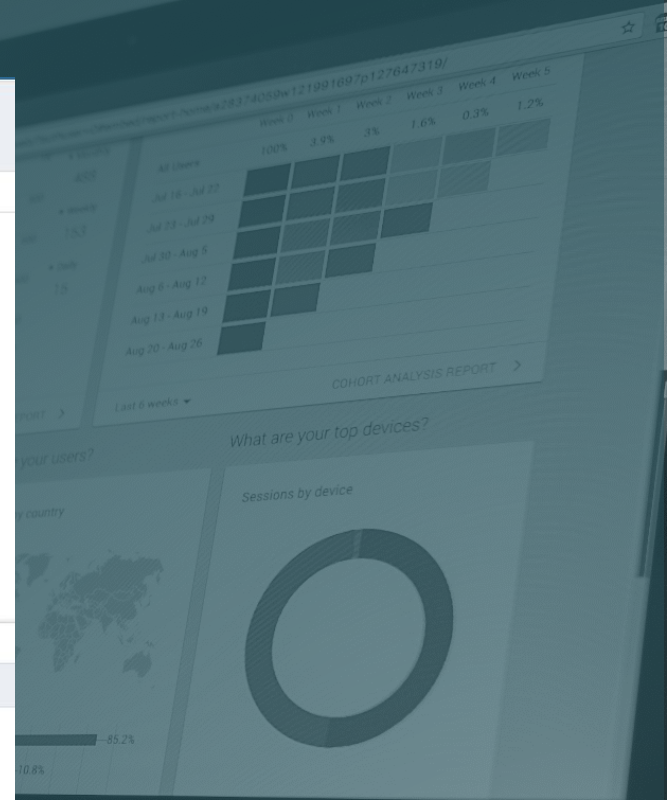
```

1 SELECT *
2 FROM (
3     SELECT
4         c.nom_commune,
5         c.code_departement,
6         ROUND(CAST(AVG(v.valeur) AS NUMERIC),2) AS "Moyenne des valeurs foncières",
7         RANK() OVER (PARTITION BY c.code_departement ORDER BY AVG(v.valeur) DESC) AS "Rang"
8     FROM vente v
9     JOIN bien b ON b.id_bien = v.id_bien
10    JOIN commune c ON (c.code_departement = b.code_departement AND c.code_commune = b.code_commune)
11   WHERE c.code_departement IN ('06','13','33','59','69')
12  GROUP BY c.nom_commune, c.code_departement
13 ) r
14 WHERE r."Rang" <= 3
15 ORDER BY r.code_departement, r."Moyenne des valeurs foncières" DESC;
```

Data Output Messages Notifications

	nom_commune character varying (70)	code_departement character varying (3)	Moyenne des valeurs foncières numeric	Rang bigint
1	Saint-Jean-Cap-Ferrat	06	968750.00	1
2	Eze	06	655000.00	2
3	Mouans-Sartoux	06	476898.09	3
4	Gignac-la-Nerthe	13	330000.00	1
5	Saint-Savournin	13	314425.00	2
6	Cassis	13	313416.88	3
7	Lège-Cap-Ferret	33	549500.64	1
8	Vayres	33	335000.00	2
9	Arcachon	33	307435.93	3
10	Bersée	59	433202.00	1
11	Cysoing	59	408550.00	2
12	Halluin	59	322250.00	3
13	Ville-sur-Jarniou	69	485300.00	1
14	Lyon 2e Arrondissement	69	455217.27	2
15	Lyon 6e Arrondissement	69	426968.25	3

Total rows: 15 of 15    Query complete 00:00:00.150



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

Laplace\_Immo/postgres@PostgreSQL 15

Query

Query History

```
1 SELECT
2   c.nom_commune,
3   (COUNT(v.id_vente)*1000/CAST(c.population AS FLOAT)) AS "Ventes pour 1000 habitants"
4 FROM vente v
5 JOIN bien b ON b.id_bien = v.id_bien
6 JOIN commune c ON (c.code_departement = b.code_departement AND c.code_commune = b.code_commune)
7 WHERE c.population > 10000
8 GROUP BY c.nom_commune, c.population
9 ORDER BY "Ventes pour 1000 habitants" DESC
10 LIMIT 20;
```

Data Output

Messages

Notifications

	nom_commune character varying (70)	Ventes pour 1000 habitants double precision
1	Paris 2e Arrondissement	5.843110190936278
2	Paris 1er Arrondissement	4.920585487387107
3	Paris 3e Arrondissement	4.693056608173498
4	Arcachon	4.622625651369978
5	La Baule-Escoublac	4.5841519318926
6	Paris 4e Arrondissement	4.0830214358625385
7	Roquebrune-Cap-Martin	3.9874242772793496
8	Paris 8e Arrondissement	3.8344827586206898
9	Sanary-sur-Mer	3.4965034965034967
10	Paris 9e Arrondissement	3.4344401697406006
11	La Londe-les-Maures	3.433556050482554
12	Paris 6e Arrondissement	3.3761628330621067
13	Saint-Cyr-sur-Mer	3.2409381663113006
14	Chantilly	3.131150474145643
15	Pornichet	3.0594405594405596
16	Saint-Mandé	3.056343019135365
17	Paris 10e Arrondissement	3.039268733522904
18	Menton	2.9372841418934184
19	Saint-Hilaire-de-Riez	2.8693157116772454
20	Vincennes	2.807087397969341

Total rows: 20 of 20    Query complete 00:00:00.155

