

PRESENTATION DES REQUETES DE LA BASE DE DONNEES DATAIMMIO

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

```
1 • select
2   count(id_bien) as "nbre appartement"
3   from
4   bien
5   where
6   type_local='Appartement';
```

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

nbre appartement
40802

Result 7 x

Requêtes 2 : Le nombre de ventes d'appartement par région pour le 1er semestre 2020.

```
1 • SELECT
2   Count(b.id_bien),
3   b.type_local,
4   r.nom_region
5   FROM
6   bien b
7   JOIN vente as v ON b.id_bien= v.id_bien
8   JOIN commune as c ON c.id_codedep_codecommune= b.id_codedep_codecommune
9   JOIN region as r ON r.id_region= c.id_region
10  WHERE
11  b.Type_local='appartement'
12  group by
13  r.nom_region
14  order by
15  count(*) desc;
```

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

Count(b.id_bien)	type_local	nom_region
28263	Appartement	Île-de-France
5253	Appartement	Nouvelle-Aquitaine
4513	Appartement	Auvergne-Rhône-Alpes
4204	Appartement	Occitanie
2975	Appartement	Hauts-de-France
2854	Appartement	Provence-Alpes-Côte d'Azur

Result 4 x

Requêtes 3 : Proportion des ventes d'appartements par le nombre de pièces.

```
1 • select
2   round(100*count(id_vente)/(select
3     count(id_vente) as "nbre appartement"
4   from
5     vente
6   join bien using (id_bien)
7   where
8     type_local='Appartement'),2) as "proportion vente",
9   b.total_piece
10  from
11    vente v
12  join bien as b on b.id_bien=v.id_bien
13  group by
14    b.total_piece
15  order by
16    total_piece;
```

Result 7

proportion vente	total_piece
17.71	1
31.75	2
33.18	3
19.74	4
5.81	5
0.98	6
0.23	7

Requêtes 4 : Liste des 10 départements où le prix du mètre carré est le plus élevé.

```
1 • select
2   c.code_departement,
3   round(avg(v.valeur/b.surface_carrez),2) as prix_m2
4  from
5    commune c
6  join bien as b on c.id_codedep_codecommune=b.id_codedep_codecommune
7  join vente v on b.id_bien=v.id_bien
8  group by
9    code_departement
10 order by
11   prix_m2 desc
12 limit
13   10
14
```

Result 2

code_departement	prix_m2
92	7137.11
75	6135.14
85	5608.26
94	5275.82
62	5224.94
28	5149.72

Requêtes 5 : Prix moyen du mètre carré d'une maison en Île-de-France.

```
1 • select
2   round(avg(v.valeur/b.surface_carrez),2) as prix_m2,
3   b.type_local,
4   r.id_region,
5   r.nom_region
6 from
7   bien b
8   join commune as c on b.id_codedep_codecommune= c.id_codedep_codecommune
9   join region as r on c.id_region= r.id_region
10  join vente as v on v.id_bien=b.id_bien
11  where
12    type_local='maison' and r.id_region=11
13  group by
14    r.nom_region;
```

Result Grid

	prix_m2	type_local	id_region	nom_region
▶	3859.25	Maison	11	Île-de-France

Result 2 x

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

```
1 • select
2   count(b.id_bien),
3   b.type_local,
4   b.surface_local,
5   round(avg(v.valeur),0) as valeur_fonciere,
6   r.nom_region
7 from
8   vente v
9   join bien as b on v.id_bien=b.id_bien
10  join commune as c on b.id_codedep_codecommune=c.id_codedep_codecommune
11  join region as r on c.id_region=r.id_region
12  where
13    type_local='appartement'
14  group by
15    b.id_bien
16  order by
17    valeur_fonciere desc
18  limit
19    10
20
```

Result Grid

	count(b.id_bien)	type_local	surface_local	valeur_fonciere	nom_region
▶	1	Appartement	46	3600000	Île-de-France
4		Appartement	230	3370000	Île-de-France
4		Appartement	357	2950000	Île-de-France
4		Appartement	270	2839300	Île-de-France
4		Appartement	281	2655000	Île-de-France
4		Appartement	204	2589600	Île-de-France
4		Appartement	166	2524710	Île-de-France

Result 2 x

Requêtes 7 : Taux d'évolution du nombre de ventes entre le premier et le second trimestre de 2020.

```

1 • with
2   vente_1er_trimestre as (select
3     count(id_vente) as vente_1
4   from
5     vente
6   where
7     date between '2020/01/01' and '2020/03/31'),
8   vente_2eme_trimestre as (select
9     count(id_vente) as vente_2
10  from
11    vente
12  where
13    date between '2020/04/01' and '2020/06/30')
14  select
15    round((vente_2-vente_1)/vente_1*100,2) as taux_evolution
16  from
17    vente_1er_trimestre, vente_2eme_trimestre;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

taux_evolution
2.59

Result 6 x

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

```

1 • select
2   count(r.id_region),
3   b.total_piece,
4   b.type_local,
5   round(avg(v.valeur/b.surface_carrez),2) as prix_m2,
6   r.nom_region
7 from
8   vente v
9 join bien as b on v.id_bien=b.id_bien
10 join commune as c on b.id_codedep_codecommune=c.id_codedep_codecommune
11 join region as r on c.id_region=r.id_region
12 where
13   b.type_local='Appartement'
14   and total_piece>=4
15 group by
16   nom_region
17 order by
18   prix_m2 desc

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

count(r.id_region)	total_piece	type_local	prix_m2	nom_region
5706	4	Appartement	4579.14	Île-de-France
527	4	Appartement	3888.74	Centre-Val de Loire
392	4	Appartement	3555.85	Pays de la Loire
67	4	Appartement	3545	La Réunion
516	4	Appartement	3406.61	Provence-Alpes-Côte d'Azur
726	5	Appartement	3087.76	Hauts-de-France

Result 2 x

Requêtes 9 : Liste des communes ayant eu au moins 50 ventes au 1er trimestre

```
1 • select
2   count(v.id_vente),
3   c.code_commune,
4   c.nom_commune
5 from
6   vente v
7 join bien as b on v.id_bien=b.id_bien
8 join commune as c on c.id_codedep_codecommune=b.id_codedep_codecommune
9 join region as r on c.id_region=r.id_region
10 where
11   v.date between '2020/01/01' and '2020/03/31'
12 group by
13   nom_commune
14 having
15   count(v.id_vente)>=50;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

count(v.id_vente)	code_commune	nom_commune
550	C44184	SAINT-NAZAIRE
62	C13028	LA CIOTAT
1377	C02043	BAGNEUX
54	C78646	VERSAILLES
68	C94071	SUCY-EN-BRIE
56	C75042	JOINVILLE-LE-PONT
52	C78199	DOMONT

Result 1 x

Requêtes 10 : Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces.

```
1 • with
2   piece_2 as (select
3     count(b.id_bien),
4     avg(valeur/surface_carrez) as prix_m2_2p
5   from
6     vente v
7   join bien as b on v.id_bien=b.id_bien
8   where
9     total_piece=2 and type_local='appartement'),
10  piece_3 as (select
11    count(b.id_bien),
12    avg(valeur/surface_carrez) as prix_m2_3p
13  from
14    vente v
15  join bien as b on v.id_bien=b.id_bien
16  where
17    total_piece=3 and type_local='appartement')
18 select
19   round((prix_m2_3p-prix_m2_2p)/prix_m2_2p*100,2) as "difference_prix"
20 from
21   piece_2, piece_3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

difference_prix
-10.5

Result 3 x

Requêtes 11 : Les moyennes de valeurs foncières pour le top 3 des communes des départements 6, 13, 33, 59 et 69.

```

1  with
2  valeur_moyenne_par_ville as (select
3    round(avg(v.valeur),0) as valeur_moyenne,
4    nom_commune,
5    code_departement
6  from
7    vente v
8  join bien as b on v.id_bien=b.id_bien
9  join commune as c on c.id_codedep_codecommune=b.id_codedep_codecommune
10 where
11   code_departement in (6 , 13, 33, 59, 69)
12 group by
13   nom_commune)
14 select valeur_moyenne, nom_commune,code_departement, classement
15 from
16 (select valeur_moyenne, nom_commune,code_departement,
17   rank() over (partition by code_departement order by valeur_moyenne) as classement
18 from
19   valeur_moyenne_par_ville) as v
20 where
21   classement <=3

```

	valeur_moyenne	nom_commune	code_departement	classement
▶	8000	BREIL-SUR-ROYA	6	1
	22000	UTELLE	6	2
	32000	MOULINET	6	3
	95000	EYGUIERES	13	1
	95000	MARIGNANE	13	1
	100000	GEMENOS	13	3

Result 2 x

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

```

1  • select
2    count(c.id_codedep_codecommune),
3    round(count(v.id_vente)/(p.nbre_habitant/1000),0) as vente_habitant,
4    nom_commune,
5    nbre_habitant,
6    valeur
7  from
8    vente v
9  join bien as b on v.id_bien=b.id_bien
10 join commune as c on b.id_codedep_codecommune=c.id_codedep_codecommune
11 join population as p on p.id_codedep_codecommune=c.id_codedep_codecommune
12 where
13   nbre_habitant>=1000 and Nbre_habitant>10000
14 group by
15   c.nom_commune
16 order by
17   vente_habitant desc
18 limit
19   20

```

	count(c.id_codedep_codecommune)	vente_habitant	nom_commune	nbre_habitant	valeur
▶	1860	65	FRESNES	28822	279000
	700	57	LA TRINITE	12177	277000
	2106	51	BAGNEUX	41070	279000
	686	45	FRANCHEVILLE	15164	161200
	1404	38	CHATILLON	36934	473000
	4860	31	SAINT-DENIS	155302	129000
	3450	31	MONTREUIL	111810	445000
	1375	19	SAINT-NAZAIRE	73139	57500

Result 21 x

