

Nombre total d'appartements vendus au 1er semestre 2020

```
select COUNT(Id_vente) as nb_ventes  
from vente v  
join bien b on b.Id_bien = v.Id_bien  
where Nat_mut = 'Vente'  
and b.type_local = 'Appartement'  
;
```

Nombre total d'appartements vendus au 1er semest

Grille		123 nb_ventes
	1	15 987

<localhost> query 2 ×

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

```
SELECT r.Nom_reg as region, count(v.Id_vente) as apparts_vendus
FROM vente v
INNER JOIN region r ON r.Id_region = v.Id_region
INNER JOIN bien b ON v.Id_bien = b.Id_bien
WHERE v.Nat_mut = 'Vente'
AND b.type_local = 'Appartement'
GROUP BY r.Nom_reg
order by apparts_vendus desc
;
```

region 1 ×

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

	Grille	Entrez une expression SQL pour filtrer les résultats (u	
		region	appart_s_vendus
1		Île-de-France	6 916
2		Auvergne-Rhône-Alpes	1 852
3		Provence-Alpes-Côte d'Azur	1 524
4		Nouvelle-Aquitaine	977
5		Occitanie	881
6		Pays de la Loire	753
7		Hauts-de-France	666
8		Grand Est	584
9		Bretagne	565
10		Normandie	495
11		Centre-Val de Loire	392
12		Bourgogne-Franche-Comté	187
13		Corse	100
14		Martinique	58
15		La Réunion	26
16		Guyane	11

<localhost> query 3 ×

Proportion des ventes d'appartements par le nombre de pièces

SELECT

b.nb_pieces,

ROUND(COUNT(DISTINCT v.id_vente) / (SELECT COUNT(DISTINCT v.id_vente) FROM vente v INNER JOIN bien b ON v.id_bien = b.id_bien WHERE b.Type_local = 'appartement'and v.Nat_mut = 'Vente') * 100, 2)

AS proportion

FROM vente v

INNER JOIN bien b ON v.id_bien = b.id_bien

WHERE b.Type_local = 'appartement'

and v.Nat_mut = 'Vente'

GROUP BY b.nb_pieces

;

bien 1 ×

Proportion des ventes d'appartements par le nombre de pièces

Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	nb_pieces	proportion
1	0	0,09
2	1	20,88
3	2	31,29
4	3	28,57
5	4	14,65
6	5	3,62
7	6	0,61
8	7	0,17
9	8	0,06
10	9	0,04
11	10	0,01
12	11	0,01

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

```
select c.Code_dep , ROUND(AVG(v.Valeur / b.Surface_carrez), 2) AS prix_moy_m2
FROM vente v
INNER JOIN bien b ON v.id_bien = b.id_bien
INNER JOIN commune c ON v.Id_codedep_codecomm = c.Id_codedep_codecomm
GROUP BY c.Code_dep
order by prix_moy_m2 desc
limit 10
;
```

Liste des 10 départements où le prix du mètre carré

Entrez une expression SQL pour filtrer les résultats (utilisez

	Code_dep	123 prix_moy_m2
1	75	12 052,89
2	92	7 219,39
3	94	5 343,28
4	06	4 700,33
5	74	4 667,13
6	93	4 344,78
7	78	4 225,25
8	69	4 059,31
9	2A	4 026,97
10	33	3 764,14

<localhost> Query 5 ×

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

```
SELECT r.Id_region, r.Nom_reg AS region, ROUND(AVG(v.Valeur / b.Surface_carrez), 0) AS prix_moy_m2
FROM vente v
INNER JOIN bien b ON v.Id_bien = b.Id_bien
INNER JOIN region r ON v.Id_region = r.Id_region
WHERE b.Type_local = 'Maison' AND r.id_region = 11
GROUP BY r.id_region, r.Nom_reg
;
```

region 1 ×

Prix moyen du mètre carré d'une maison en Île-de-Fr | Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

Grille		123 Id_region ↕	ABC region ↕	123 prix_moy_m2 ↕
	1	11	Île-de-France	3 745

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

```

SELECT v.Valeur, b.Surface_carrez, r.Nom_reg
FROM vente v
JOIN bien b ON v.Id_bien = b.Id_bien
JOIN region r ON v.Id_region = r.Id_region
WHERE b.Type_local = 'appartement'
ORDER BY v.valeur DESC
LIMIT 10
;

```

vente(+) 1 ×

Liste des 10 appartements les plus chers avec la régi *Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)*

	123 Valeur	123 Surface_carrez	ABC Nom_reg
1	9 000 000	9,1	Île-de-France
2	8 600 000	64	Île-de-France
3	8 577 710	20,55	Île-de-France
4	7 620 000	42,77	Île-de-France
5	7 600 000	253,3	Île-de-France
6	7 535 000	139,9	Île-de-France
7	7 420 000	360,95	Île-de-France
8	7 200 000	595	Île-de-France
9	7 050 000	122,56	Île-de-France
10	6 600 000	79,38	Île-de-France

<localhost> query 7 ×

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

```
WITH v1 AS  
(SELECT COUNT(Id_vente) AS 'ventetrimestre1'  
FROM vente  
WHERE Date_mut BETWEEN '2020-01-01' AND '2020-03-31'),  
  
v2 AS  
(SELECT COUNT(Id_vente) AS 'ventetrimestre2'  
FROM vente  
WHERE Date_mut BETWEEN '2020-04-01' AND '2020-06-30')  
  
SELECT ventetrimestre1 AS 'nombre de ventes au 1er trimestre 2020',  
ventetrimestre2 AS 'nombre de ventes au 2nd trimestre 2020',  
(ventetrimestre2-ventetrimestre1)/ventetrimestre1*100 as taux_evolution  
from v1, v2  
;
```

Résultats 1 ×

Taux d'évolution du nombre de ventes entre le prem | Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	123 nombre de ventes au 1er trimestre 2020	123 nombre de ventes au 2nd trimestre 2020	123 taux_evolution
1	16 776	17 393	3,6779

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

```
SELECT r.Nom_reg AS region, ROUND(AVG(v.Valeur / b.Surface_carrez), 2) AS prix_moy_m2
FROM bien b
INNER JOIN vente v ON b.Id_bien = v.Id_bien
INNER JOIN region r ON v.Id_region = r.Id_region
WHERE b.Type_local = 'Appartement' and b.Nb_pieces >4
group by r.Nom_reg
order by prix_moy_m2 desc
;
```

Le classement des régions par rapport au prix au mètre *Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)*

	ABC region	123 prix_moy_m2
1	Île-de-France	8 770,44
2	La Réunion	3 641,81
3	Provence-Alpes-Côte d'Azur	3 587,65
4	Corse	3 104,88
5	Auvergne-Rhône-Alpes	2 891,38
6	Nouvelle-Aquitaine	2 465,48
7	Bretagne	2 412,05
8	Pays de la Loire	2 315,76
9	Hauts-de-France	2 189,93
10	Occitanie	2 097,23
11	Normandie	2 015,77
12	Grand Est	1 540,89
13	Centre-Val de Loire	1 453,11
14	Bourgogne-Franche-Comté	1 251,19
15	Martinique	573,48

<localhost> query 9 ×

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

SELECT c.nom_comm as 'commune', COUNT(v.id_vente) as 'ventes'
FROM vente v
JOIN commune c ON v.Id_codedep_codecomm = c.Id_codedep_codecomm
WHERE v.Date_mut BETWEEN '2020-01-01' AND '2020-03-31'
GROUP BY c.nom_comm
HAVING COUNT(v.id_vente) >= 50
order by ventes DESC
;
```

commune 1 ×

Liste des communes ayant eu au moins 50 ventes au | Entrez une expression SQL pour filtrer les résu

Grille	commune		123ventes
	1	2	3
Texte	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	Grenoble	106
	15	Paris 9e Arrondissement	106
	16	Boulogne-Billancourt	99
	17	Paris 13e Arrondissement	94
	18	Paris 7e Arrondissement	87
	19	Paris 6e Arrondissement	86
	20	Marseille 8e Arrondissement	81
	21	Asnières-sur-Seine	81

<localhost> query 10 ×

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

WITH prix_m2_2_pieces AS (
    SELECT ROUND(AVG(v.valeur / b.Surface_carrez), 2) as 'prix_m2_2_pieces'
    FROM vente v
    JOIN bien b ON v.Id_bien = b.Id_bien
    WHERE b.nb_pieces = 2
),
prix_m2_3_pieces AS (
    SELECT ROUND(AVG(v.valeur / b.Surface_carrez), 2) as 'prix_m2_3_pieces'
    FROM vente v
    JOIN bien b ON v.Id_bien = b.Id_bien
    WHERE b.nb_pieces = 3
)

SELECT (prix_m2_3_pieces - prix_m2_2_pieces) / prix_m2_2_pieces * 100 as 'difference_pourcentage'
FROM prix_m2_2_pieces, prix_m2_3_pieces
;
```

Résultats 1 ×

Différence en pourcentage du prix au mètre carré en *Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)*

Grille		123 difference_pourcentage
1		-12,8604693185

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

```
SELECT *
FROM (
  SELECT c.code_dep, c.nom_comm, ROUND(AVG(v.valeur), 0) as 'valeur_moyenne',
  RANK() OVER (PARTITION BY c.code_dep ORDER BY AVG(v.valeur) DESC) as 'rang'
  FROM vente v
  JOIN commune c ON v.Id_codedep_codecomm = c.Id_codedep_codecomm
  WHERE c.code_dep IN (6, 13, 33, 59, 69)
  GROUP BY c.code_dep, c.nom_comm
) t
WHERE t.rang <= 3
;
```

Les moyennes de valeurs foncières pour le top 3 des | Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	code_dep	nom_comm	valeur_moyenne	rang
1	06	Saint-Jean-Cap-Ferrat	968 750	1
2	06	Eze	655 000	2
3	06	Mouans-Sartoux	476 898	3
4	13	Gignac-la-Nerthe	330 000	1
5	13	Saint-Savournin	314 425	2
6	13	Cassis	313 417	3
7	33	Lège-Cap-Ferret	549 501	1
8	33	Vayres	335 000	2
9	33	Arcachon	307 436	3
10	59	Bersée	433 202	1
11	59	Cysoing	408 550	2
12	59	Halluin	322 250	3
13	69	Ville-sur-Jarnioux	485 300	1
14	69	Lyon 2e Arrondissement	455 217	2
15	69	Lyon 6e Arrondissement	426 968	3

```

-- ## Les 20 communes avec le plus de transactions pour 1000 habitants pour les communes qui dépassent les 10 000 habitants
WITH transactions_pour_1000_habitants AS (
    SELECT c.nom_comm, ROUND(COUNT(v.Id_vente) / (c.Population / 1000), 2) as 'transactions_pour_1000_habitants'
    FROM vente v
    JOIN commune c ON v.Id_codedep_codecomm = c.Id_codedep_codecomm
    WHERE c.Population > 10000
    GROUP BY c.nom_comm, c.Population
)
SELECT *
FROM transactions_pour_1000_habitants
ORDER BY transactions_pour_1000_habitants DESC
LIMIT 20
;

```

commune 1 ×

Les 20 communes avec le plus de transactions pour 1 | Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)

	nom_comm	transactions_pour_1000_habitants
2	Paris 1er Arrondissement	4,92
3	Paris 3e Arrondissement	4,69
4	Arcachon	4,62
5	La Baule-Escoublac	4,58
6	Paris 4e Arrondissement	4,08
7	Roquebrune-Cap-Martin	3,99
8	Paris 8e Arrondissement	3,83
9	Sanary-sur-Mer	3,5
10	La Londe-les-Maures	3,43
11	Paris 9e Arrondissement	3,43
12	Paris 6e Arrondissement	3,38
13	Saint-Cyr-sur-Mer	3,24
14	Chantilly	3,13
15	Pornichet	3,06
16	Saint-Mandé	3,06
17	Paris 10e Arrondissement	3,04
18	Menton	2,94
19	Saint-Hilaire-de-Riez	2,87
20	Vincennes	2,81