

# REQUETES SQL

## SGBDR : SQLite Studio

### REQUETE 1

```
SELECT COUNT(DISTINCT b.id_bien) AS "nb_appartement_vendus"  
FROM vente v  
JOIN bien b  
ON v.id_bien = b.id_bien  
WHERE b.type_local = "Appartement" AND v.date_vente BETWEEN "2020-01-01" AND "2020-06-30";
```

Table    Formulaire

 Nombre de lignes chargées : 1

	nb_appartement_vendus
1	31378

## REQUETE 2

```
SELECT r.nomRegion, COUNT(DISTINCT b.id_bien) AS "nb_appartement_vendus_par_region"
FROM vente v
INNER JOIN bien b
    ON v.id_bien = b.id_bien
JOIN commune c
    ON c.id_codedep_codecommune = b.id_codedep_codecommune
JOIN region r
    ON r.id_region = c.id_region
WHERE b.type_local = "Appartement" AND
v.date_vente BETWEEN "2020-01-01" AND "2020-06-30"
GROUP BY c.id_region
ORDER BY nb_appartement_vendus_par_region DESC;
```

able Formulaire

Nombre de lignes chargées : 17

nomRegion	nb_appartement_vendus_par_region
Ile-de-France	13995
Provence-Alpes-Côte d'Azur	3649
Auvergne-Rhône-Alpes	3253
Nouvelle-Aquitaine	1932
Occitanie	1640
Pays de la Loire	1357
Hauts-de-France	1254
Grand Est	984
Bretagne	983
Normandie	862
Centre-Val de Loire	696
Bourgogne-Franche-Comté	376
Corse	223
Martinique	94
La Réunion	44
Guyane	34
Guadeloupe	2

### REQUETE 3

```
Requête Historique
1 SELECT
2   b.total_piece,
3   COUNT(b.total_piece) AS nombre_appartements_vendus,
4   ROUND(COUNT(b.total_piece) * 100.0 / (SELECT COUNT(*) FROM vente),2) AS proportion_ventes_par_pieces
5 FROM
6   vente v
7 JOIN
8   bien b
9     ON v.id_bien = b.id_bien
10 WHERE
11   lower(b.type_local) = "appartement"
12 GROUP BY
13   b.total_piece
14 ORDER BY
15   b.total_piece;
16
```

	total_piece	nombre_appartements_vendus	proportion_ventes_par_pieces
1	0	30	0.09
2	1	6739	19.72
3	2	9783	28.63
4	3	8966	26.24
5	4	4460	13.05
6	5	1114	3.26
7	6	204	0.6
8	7	54	0.16
9	8	17	0.05
10	9	8	0.02
11	10	2	0.01
12	11	1	0

## REQUETE 4

```
Requête Historique
1 SELECT
2     c.code_dep,
3     v.prix / b.surface_carrez AS prix_m2
4 FROM
5     vente v
6 INNER JOIN
7     bien b
8 ON
9     v.id_bien = b.id_bien
10 JOIN commune c
11     ON b.id_codedep_codecommune = c.id_codedep_codecommune
12 GROUP BY
13     c.code_dep
14 ORDER BY prix_m2 DESC
15 LIMIT 10;
16
```

Table Formulaire

Nombre de lignes chargées : 10

	code_dep	prix_m2
1	75	11900
2	06	9120
3	34	8743
4	73	8000
5	35	6428
6	05	5535
7	92	5464
8	78	5314
9	2A	5000
10	974	4801

## REQUETE 5

```
Requête Historique
1 SELECT
2     r.nomRegion,
3     b.type_local,
4     ROUND(AVG(v.prix / b.surface_carrez)) AS prix_moyen_m2,
5     ROUND(SUM(v.prix) / SUM(b.surface_carrez)) AS prix_moyen_m2_autre_calcul
6 FROM
7     bien b
8 JOIN
9     vente v
10 ON
11     v.id_bien = b.id_bien
12 JOIN commune c
13     ON b.id_codedep_codecommune = c.id_codedep_codecommune
14 JOIN region r
15     ON r.id_region = c.id_region
16 WHERE lower(b.type_local) LIKE "maison" AND lower(r.nomRegion) LIKE "_le_de_france";
17
```

Table Formulaire

Nombre de lignes chargées : 1

	nomRegion	type_local	prix_moyen_m2	prix_moyen_m2_autre_calcul
1	Ile-de-France	Maison	3764	3693

## REQUETE 6

```

1 SELECT b.id_bien,
2        v.prix,
3        r.nomRegion,
4        b.type_local,
5        ROUND(b.surface_carrez) as surface
6 FROM vente v
7 LEFT JOIN
8 bien b ON v.id_bien = b.id_bien
9 JOIN
10 commune c ON b.id_codedep_codecommune = c.id_codedep_codecommune
11 JOIN
12 region r ON c.id_region = r.id_region
13 WHERE lower(b.type_local) = "appartement" AND
14        b.id_bien NOT IN (1677,3621,5519,26725,28963,29553,31142,31700,32504,34011,3919,8281,11346,11517,18483,19395)
15 --GROUP BY v.prix
16 ORDER BY v.prix DESC
17 LIMIT 10;

```

Table

Formulaire

1

Nombre de lignes chargées : 10

	id_bien	prix	nomRegion	type_local	surface
1	30603	9000000	Ile-de-France	Appartement	9
2	5261	8600000	Ile-de-France	Appartement	64
3	3625	8577713	Ile-de-France	Appartement	20
4	7602	7620000	Ile-de-France	Appartement	42
5	9988	7600000	Ile-de-France	Appartement	253
6	17823	7535000	Ile-de-France	Appartement	139
7	410	7420000	Ile-de-France	Appartement	360
8	16357	7200000	Ile-de-France	Appartement	595
9	1924	7050000	Ile-de-France	Appartement	122
10	19161	6600000	Ile-de-France	Appartement	79

## REQUETE 7

```

1 WITH
2 second AS
3 (
4     SELECT COUNT(v.id_bien) AS secondTrimestre
5     FROM vente v
6     JOIN bien b ON v.id_bien = b.id_bien
7     WHERE v.date_vente BETWEEN "2020-04-01" AND "2020-06-30"
8     ORDER BY v.date_vente
9 ),
10
11 first AS (
12     SELECT COUNT(v.id_bien) AS premierTrimestre
13     FROM vente v
14     JOIN
15     bien b ON v.id_bien = b.id_bien
16     WHERE v.date_vente BETWEEN "2020-01-01" AND "2020-03-31"
17     ORDER BY v.date_vente
18 )
19 SELECT (ROUND(secondTrimestre - premierTrimestre)/premierTrimestre) * 100 AS tauxVariation
20 FROM
21     second,
22     first;

```

Table	Formulaire
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div>1</div> <div></div> <div></div> </div> <div>Nombre de lignes chargées : 1</div>	
	tauxVariation
1	3.67787315212208

## REQUETE 8

```
Requête Historique
1 SELECT
2     r.nomRegion,
3     ROUND(AVG(v.prix / b.surface_carrez),2) AS prix_moyen_m2,
4     b.total_piece AS appartement_plus_4,
5     ROW_NUMBER() OVER (ORDER BY AVG(v.prix / b.surface_carrez) DESC) AS row_number_
6 FROM vente v
7 JOIN bien b
8     ON v.id_bien = b.id_bien
9 JOIN commune c
10    ON c.id_codedep_codecommune = b.id_codedep_codecommune
11 JOIN region r
12    ON r.id_region = c.id_region
13 WHERE lower(b.type_local) LIKE 'ap_ar%' AND b.total_piece > 4
14 GROUP BY r.nomRegion;
```

Table		Formulaire			
		Nombre de lignes chargées : 15			
		Page suivante			
	nomRegion	prix_moyen_m2	appartement_plus_4	row_number_	
1	Ile-de-France	8806.2	5	1	
2	La Réunion	3659	5	2	
3	Provence-Alpes-Côte d'Azur	3616.21	5	3	
4	Corse	3117.2	5	4	
5	Auvergne-Rhône-Alpes	2903.36	5	5	
6	Nouvelle-Aquitaine	2476.01	5	6	
7	Bretagne	2426.7	5	7	
8	Pays de la Loire	2328.7	5	8	
9	Hauts-de-France	2199.43	5	9	
10	Occitanie	2106.77	5	10	
11	Normandie	2025.88	5	11	
12	Grand Est	1560.43	5	12	
13	Centre-Val de Loire	1459.52	5	13	
14	Bourgogne-Franche-Comté	1260.42	5	14	
15	Martinique	574	5	15	

## REQUETE 9

```
Requête Historique
1 WITH vente_premier_trimestre AS(
2     SELECT c.nom_commune, COUNT(v.id_vente) AS nb_vente
3     FROM vente v
4     JOIN bien b
5         ON b.id_bien = v.id_bien
6     JOIN commune c
7         ON c.id_codedep_codecommune = b.id_codedep_codecommune
8     WHERE v.date_vente BETWEEN "2020-01-01" AND "2020-03-31"
9     GROUP BY nom_commune
10 )
11 SELECT
12     nom_commune AS commune,
13     nb_vente
14 FROM vente_premier_trimestre
15 WHERE nb_vente >= 50;
16
```

Table Formulaire

Nombre de lignes chargées : 48

	commune	nb_vente
1	Ajaccio	54
2	Angers	64
3	Antibes	77
4	Asnières-sur-Seine	81
5	Bordeaux	157
6	Boulogne-Billancourt	99
7	Courbevoie	80
8	Grenoble	106
9	Issy-les-Moulineaux	50
10	La Ciotat	62
11	Levallois-Perret	59
12	Lille	67
13	Marseille 1er Arrondissement	71
14	Marseille 4e Arrondissement	72
15	Marseille 8e Arrondissement	81
16	Marseille 9e Arrondissement	66
17	Montreuil	65
18	Nantes	119

## REQUETE 10

```

Requête Historique
1 WITH prix_m2_3_pieces AS
2 (
3   SELECT ROUND(AVG(v.prix / b.surface_local), 2) AS prix_m2
4   FROM vente v
5   JOIN bien b ON v.id_bien = b.id_bien
6   WHERE b.total_piece = 3
7 ),
8 prix_m2_2_pieces AS
9 (
10  SELECT ROUND(AVG(v.prix / b.surface_local), 2) AS prix_m2
11  FROM vente v
12  JOIN bien b ON v.id_bien = b.id_bien
13  WHERE b.total_piece = 2
14 )
15 |
16 SELECT
17   prix_m2_2_pieces.prix_m2 AS prix_m2_2_pieces,
18   prix_m2_3_pieces.prix_m2 AS prix_m2_3_pieces,
19   ((prix_m2_3_pieces.prix_m2 - prix_m2_2_pieces.prix_m2) / prix_m2_2_pieces.prix_m2) * 100 AS difference_pourcentage
20 FROM prix_m2_2_pieces, prix_m2_3_pieces
21

```

Table			
Formulaire			
<div> <div> <div>+</div> <div>✓</div> <div>✕</div> <div>↺</div> <div>↻</div> <div>1</div> <div>↷</div> <div>↶</div> <div>🖨</div> </div> <div>Nombre de lignes chargées : 1</div> </div>			
	prix_m2_2_pieces	prix_m2_3_pieces	difference_pourcentage
1	4900.96	4241.39	-13.45797558029447

## REQUETE 11

```

Requête Historique
1 SELECT nom_commune, code_dep, moyenne_valeur_fonciere, rank_
2 FROM (
3     SELECT c.nom_commune, c.code_dep,
4           ROUND(AVG(v.prix)) AS moyenne_valeur_fonciere,
5           RANK() OVER(PARTITION BY c.code_dep ORDER BY AVG(v.prix) DESC) AS rank_
6     FROM vente v
7     JOIN bien b
8         ON v.id_bien = b.id_bien
9     JOIN commune c
10        ON b.id_codedep_codecommune = c.id_codedep_codecommune
11     WHERE c.code_dep IN ("06", "13", "33", "59", "69")
12     GROUP BY c.code_dep, c.nom_commune
13 ) ranked_data
14 WHERE rank_ <= 3;

```

	nom_commune	code_dep	moyenne_valeur_fonciere	rank_
1	Saint-Jean-Cap-Ferrat	06	968750	1
2	Eze	06	655000	2
3	Mouans-Sartoux	06	476898	3
4	Gignac-la-Nerthe	13	330000	1
5	Saint-Savournin	13	314425	2
6	Cassis	13	313417	3
7	Lège-Cap-Ferret	33	549501	1
8	Vayres	33	335000	2
9	Arcachon	33	307436	3
10	Bersée	59	433202	1
11	Cysoing	59	408550	2
12	Halluin	59	322250	3
13	Ville-sur-Jarnioux	69	485300	1
14	Lyon 2e Arrondissement	69	455217	2
15	Lyon 6e Arrondissement	69	426968	3



## REQUETE 12

Requête Historique

```

1 SELECT
2   c.nom_commune AS commune,
3   COUNT(v.id_vente) AS nombre_transactions,
4   c.population_totale,
5   ROUND(COUNT(v.id_vente) * 1000.0 / c.population_totale, 2) AS vente_pour_1000_habitants
6 FROM vente v
7 JOIN bien b
8   ON v.id_bien = b.id_bien
9 JOIN commune c
10  ON c.id_codedep_codecommune = b.id_codedep_codecommune
11 WHERE c.population_totale > 10000
12 GROUP BY c.population_totale , c.nom_commune
13 ORDER BY vente_pour_1000_habitants DESC
14 LIMIT 20;

```

Table Formulaire

Nombre de lignes chargées : 20

	commune	nombre_transactions	population_totale	vente_pour_1000_habitants
1	Paris 2e Arrondissement	127	21735	5.84
2	Paris 1er Arrondissement	79	16055	4.92
3	Paris 3e Arrondissement	161	34306	4.69
4	Arcachon	55	11898	4.62
5	La Baule-Escoublac	77	16797	4.58
5	Paris 4e Arrondissement	120	29390	4.08
7	Roquebrune-Cap-Martin	52	13041	3.99
3	Paris 8e Arrondissement	139	36250	3.83
3	Sanary-sur-Mer	60	17160	3.5
10	La Londe-les-Maures	37	10776	3.43
11	Paris 9e Arrondissement	208	60563	3.43
12	Paris 6e Arrondissement	139	41171	3.38
13	Saint-Cyr-sur-Mer	38	11725	3.24
14	Chantilly	35	11178	3.13
15	Pornichet	35	11440	3.06
16	Saint-Mandé	69	22576	3.06
17	Paris 10e Arrondissement	264	86863	3.04
18	Menton	91	30981	2.94
19	Saint-Hilaire-de-Riez	33	11501	2.87
20	Vincennes	141	50230	2.81