

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

The screenshot shows a database query interface. On the left, a 'SCHEMAS' panel lists the database structure, including tables like 'bien', 'commune', 'region', and 'vente'. The main query editor contains the following SQL code:

```
-- Question 1 : Nombre total d'appartements vendus au 1er semestre 2020  
  
select count(distinct(bien.id_bien)) as "Nombre total d'appartement vendu au premier semestre 2020" from bien  
join vente on bien.id_bien = vente.id_bien  
where type_local = "Appartement" and date_mutation  
between '2020-01-01' and '2020-06-30';
```

The 'Result Grid' shows a single row with the value 31364.

id_bien	reg_nom	nombre de vente d'appartement par region premier semestre 2020
84	Auvergne-Rhône-Alpes	3253
27	Bourgogne-Franche-Comté	376
53	Bretagne	983
24	Centre-Val de Loire	696
94	Corse	223
44	Grand Est	983
1	Guadeloupe	2
3	Guyane	34
32	Hauts-de-France	1252

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

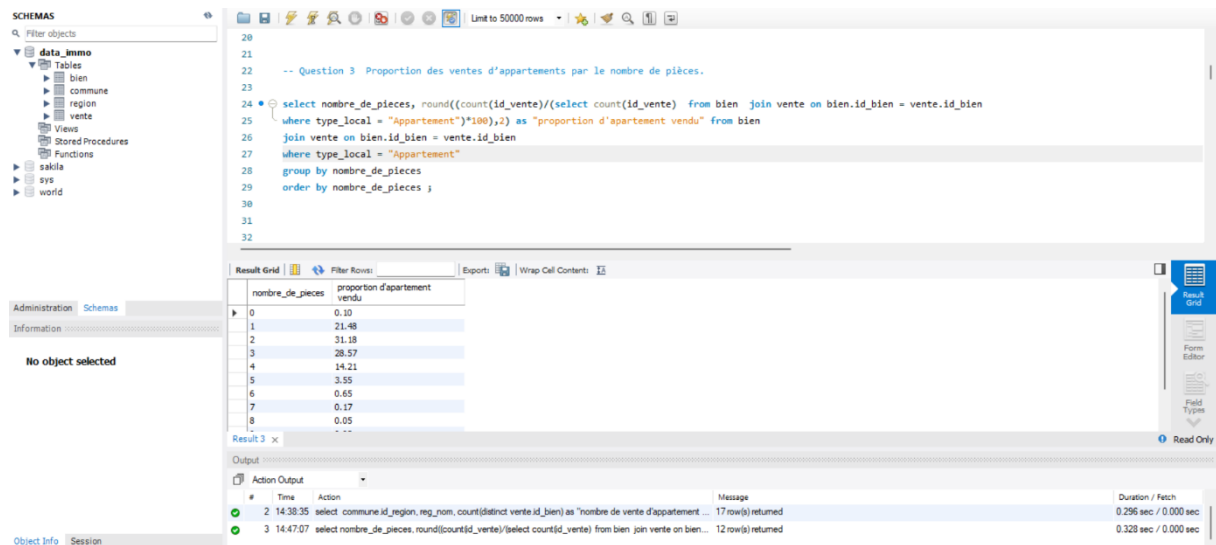
The screenshot shows a database query interface. On the left, a 'SCHEMAS' panel lists the database structure, including tables like 'bien', 'commune', 'region', and 'vente'. The main query editor contains the following SQL code:

```
-- Question 2 : Le nombre de ventes d'appartement par région pour le 1er semestre 2020  
  
select commune.id_region, reg_nom, count(distinct vente.id_bien) as "nombre de vente d'appartement par region premier semestre 2020" from vente  
join bien on vente.id_bien = bien.id_bien  
join commune on bien.id_codedep_codecommune = commune.id_codedep_codecommune  
join region on commune.id_region = region.id_region  
where type_local = "Appartement" and date_mutation  
between '2020-01-01' and '2020-06-30'  
group by reg_nom, commune.id_region;
```

The 'Result Grid' shows a table with 9 rows, representing the number of apartment sales by region for the first semester of 2020.

id_region	reg_nom	nombre de vente d'appartement par region premier semestre 2020
84	Auvergne-Rhône-Alpes	3253
27	Bourgogne-Franche-Comté	376
53	Bretagne	983
24	Centre-Val de Loire	696
94	Corse	223
44	Grand Est	983
1	Guadeloupe	2
3	Guyane	34
32	Hauts-de-France	1252

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



```
-- Question 3 Proportion des ventes d'appartements par le nombre de pièces.

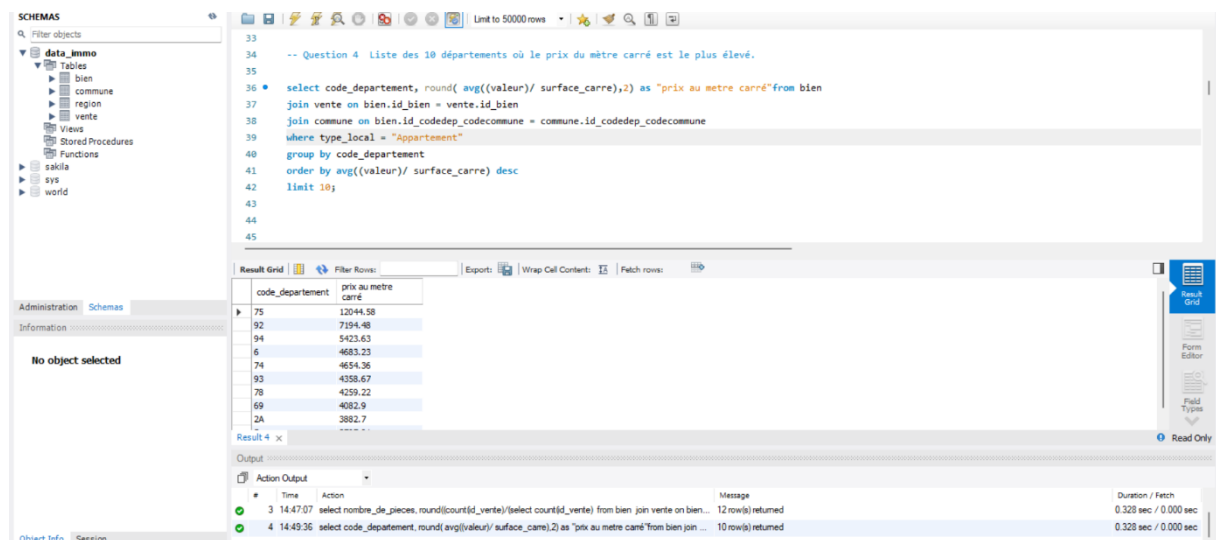
select nombre_de_pieces, round((count(id_vente)/(select count(id_vente) from bien join vente on bien.id_bien = vente.id_bien
where type_local = "Appartement")*100),2) as "proportion d'appartement vendu" from bien
join vente on bien.id_bien = vente.id_bien
where type_local = "Appartement"
group by nombre_de_pieces
order by nombre_de_pieces ;
```

nombre_de_pieces	proportion d'appartement vendu
0	0.10
1	21.48
2	31.18
3	28.57
4	14.21
5	3.55
6	0.65
7	0.17
8	0.05

Result 3 x

#	Time	Action	Message	Duration / Fetch
2	14:38:35	select commune_id_region, reg_nom, count(distinct vente.id_bien) as "nombre de vente d'appartement ...	17 row(s) returned	0.296 sec / 0.000 sec
3	14:47:07	select nombre_de_pieces, round((count(id_vente)/(select count(id_vente) from bien join vente on bien...	12 row(s) returned	0.328 sec / 0.000 sec

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



```
-- Question 4 Liste des 10 départements où le prix du mètre carré est le plus élevé.

select code_departement, round( avg((valeur)/ surface_carre),2) as "prix au metre carré" from bien
join vente on bien.id_bien = vente.id_bien
join commune on bien.id_codedep_codecommune = commune.id_codedep_codecommune
where type_local = "Appartement"
group by code_departement
order by avg((valeur)/ surface_carre) desc
limit 10;
```

code_departement	prix au metre carré
75	12044.58
92	7394.48
94	5423.63
6	4683.23
74	4654.36
93	4358.67
78	4259.22
69	4082.9
2A	3882.7
...	...

Result 4 x

#	Time	Action	Message	Duration / Fetch
3	14:47:07	select nombre_de_pieces, round((count(id_vente)/(select count(id_vente) from bien join vente on bien...	12 row(s) returned	0.328 sec / 0.000 sec
4	14:49:36	select code_departement, round(avg((valeur)/ surface_carre),2) as "prix au metre carré" from bien join ...	10 row(s) returned	0.328 sec / 0.000 sec

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

The screenshot shows a database management tool interface. On the left, the 'SCHEMAS' panel displays a tree view of the database structure, including tables like 'bien', 'commune', 'region', and 'vente'. The main query editor contains the following SQL code:

```
-- Question 5 Prix moyen du mètre carré d'une maison en Île-de-France

select round((avg(valeur)/ surface_carre),2) " prix moyen du metre carré d'une maison en ile de france" from bien
join vente on bien.id_bien = vente.id_bien
join commune on bien.id_codedep_codecommune = commune.id_codedep_codecommune
join region on commune.id_region = region.id_region
where type_local = "Maison" and reg_nom = "Île-de-France";
```

Below the query editor, the 'Result Grid' shows a single row with the value 1613.38. The 'Output' panel at the bottom displays the execution log, showing that the query was executed successfully and returned 1 row(s).

Question 6 : Liste des 10 appartements les plus chers avec la région et le nombre de mètre carré

The screenshot shows the same database management tool interface. The main query editor contains the following SQL code:

```
select bien.id_bien, reg_nom, surface_carre, round((valeur),2) as " prix " from bien
join vente on bien.id_bien = vente.id_bien
join commune on bien.id_codedep_codecommune = commune.id_codedep_codecommune
join region on commune.id_region = region.id_region
where type_local = "Appartement"
group by bien.id_bien, reg_nom, surface_carre
order by valeur desc
limit 10;
```

Below the query editor, the 'Result Grid' displays a table with 4 columns: id_bien, reg_nom, surface_carre, and prix. The table contains 10 rows of data, representing the top 10 most expensive apartments. The 'Output' panel at the bottom shows the execution log, indicating that the query was executed successfully and returned 10 row(s).

id_bien	reg_nom	surface_carre	prix
30591	Île-de-France	9.1	9000000
5261	Île-de-France	64	8600000
3625	Île-de-France	20.55	8577713
7602	Île-de-France	42.77	7620000
9988	Île-de-France	253.3	7600000
17821	Île-de-France	139.9	7535000
410	Île-de-France	360.95	7420000
18355	Île-de-France	595	7200000
1924	Île-de-France	122.56	7050000
19158	Île-de-France	79.38	6600000

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

```
-- Question 7 Taux d'évolution du nombre de ventes entre le premier et le second trimestre 2020
72
73
74
75
76 • select @nombreMutationPremierTrimestre := count(vente.id_vente)
77 from vente where date_mutation between '2020-01-01' and '2020-03-31';
78 • select @nombreMutationDeuxiemeTrimestre := count(vente.id_vente)
79 from vente where date_mutation between '2020-04-01' and '2020-06-30';
80 • select round(((@nombreMutationDeuxiemeTrimestre - @nombreMutationPremierTrimestre) / @nombreMutationPremierTrimestre * 100), 2)
81 AS "Taux d'évolution du nombre de ventes";
82
83
84
```

Result Grid

Taux d'évolution du nombre de ventes
3.68

Output

#	Time	Action	Message	Duration / Fetch
8	15:12:01	select @nombreMutationDeuxiemeTrimestre := count(vente.id_vente) from vente where date_mutation ...	1 row(s) returned	0.031 sec / 0.000 sec
9	15:12:01	select round(((@nombreMutationDeuxiemeTrimestre - @nombreMutationPremierTrimestre) / @nombreMut...	1 row(s) returned	0.000 sec / 0.000 sec

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

```
-- Question 8 . Le classement des régions par rapport au prix au mètre carré des appartement de plus de 4 pièces
84
85
86
87
88 • select reg_nom, round(avg(valeur/surface_carre), 2) as "prix au metre carré" from bien
89 join vente on bien.id_bien = vente.id_bien
90 join commune on bien.id_code_dep_codecommune = commune.id_code_dep_codecommune
91 join region on commune.id_region = region.id_region
92 where nombre_de_pieces > 4 and type_local = "Appartement" and surface_carre != 0
93 group by reg_nom
94 order by (avg(valeur) / surface_carre) desc;
95
96
```

Result Grid

reg_nom	prix au metre carré
Ile-de-France	8757.64
Provence-Alpes-Côte d'Azur	3587.65
Auvergne-Rhône-Alpes	2891.38
La Réunion	3641.81
Nouvelle-Aquitaine	2465.48
Corse	3104.88
Occitanie	2997.23
Bretagne	2412.05
Hauts-de-France	2189.93

Output

#	Time	Action	Message	Duration / Fetch
9	15:12:01	select round(((@nombreMutationDeuxiemeTrimestre - @nombreMutationPremierTrimestre) / @nombreMut...	1 row(s) returned	0.000 sec / 0.000 sec
10	15:21:43	select reg_nom, round(avg(valeur/surface_carre), 2) as "prix au metre carré" from bien join vente on bie...	15 row(s) returned	0.109 sec / 0.000 sec

Question 9 : Liste des communes ayant eu au moins 50 ventes au 1er trimestre 2020

```
-- Question 9 Liste des communes ayant eu au moins 50 ventes au 1er trimestre 2020
99 • drop table if exists vente_t1;
100 • create temporary table vente_t1 as
101 select commune, count(vente.id_vente) as "nombre_de_vente" from vente
102 join bien on vente.id_bien = bien.id_bien
103 join commune on bien.id_codecommune = commune.id_codecommune
104 where date_mutation between '2020-01-01' and '2020-03-31'
105 group by commune;
106 • select*from vente_t1
107 where nombre_de_vente > 50;
108
```

commune	nombre_de_vente
ANTIBES	77
NICE	173
LA CIOTAT	62
MARSEILLE 9EME	81
TOULOUSE	78
BORDEAUX	157
GRENOBLE	106
PARIS 05	79
PARIS 10	109
PARIS 11	169

vente_t1 11 x

Output

#	Time	Action	Message	Duration / Fetch
12	15:33:41	create temporary table vente_t1 as select commune, count(vente.id_vente) as 'nombre_de_vente' fr...	2353 row(s) affected Records: 2353 Duplicates: 0 Warnings: 0	0.141 sec
13	15:33:42	select*from vente_t1 where nombre_de_vente > 50 LIMIT 0, 50000	47 row(s) returned	0.016 sec / 0.000 sec

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

```
114 • select@PourcentagePrixaumetreCarreAppartement2pieces := round(avg(valeur/surface_carre),2)
115 from vente
116 join bien on vente.id_bien = bien.id_bien
117 where nombre_de_pieces = 2 and type_local = "Appartement";
118 • select@PourcentagePrixaumetreCarreAppartement3pieces := avg(valeur/surface_carre)
119 from vente
120 join bien on vente.id_bien = bien.id_bien
121 where nombre_de_pieces = 3 and type_local = "Appartement";
122 • select round(((@PourcentagePrixaumetreCarreAppartement3pieces - @PourcentagePrixaumetreCarreAppartement2pieces )/@PourcentagePrixaumetreCarreAppartement2pieces*100,2)
123 "différence en pourcentage du prix du metre carré entre appartement à 2 et 3 pieces";
124
125
126
```

différence en pourcentage du prix du metre carré entre appartement à 2 et 3 pieces
-12.31

Result 12 Result 13 Result 14 x

Output

#	Time	Action	Message	Duration / Fetch
15	15:39:21	select@PourcentagePrixaumetreCarreAppartement3pieces -> avg(valeur/surface_carre) from vente join...	1 row(s) returned	0.063 sec / 0.000 sec
16	15:39:21	select round(((@PourcentagePrixaumetreCarreAppartement3pieces - @PourcentagePrixaumetreCarreAp...	1 row(s) returned	0.000 sec / 0.000 sec

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

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

```

127 -- Question 11 . Les moyennes de valeurs foncières pour le top 3 des communes des départements 6, 13, 33, 59 et 69.
128
129 • drop table if exists valeur_par_commune;
130 • create temporary table valeur_par_commune as
131 select code_departement, commune, round((valeur)/2) as valeur_moyenne
132 from vente
133 join bien on bien.id_bien = vente.id_bien
134 join commune on bien.id_codedep_codecommune = commune.id_codedep_codecommune
135 where code_departement in ('6','13','33','59','69')
136 group by code_departement, commune;
137
138 • select*from(select code_departement, commune, valeur_moyenne, rank () over(partition by Code_departement order by valeur_moyenne desc) rang
139 from valeur_par_commune) as resultat
140 where rang <=3 order by code_departement, rang
141
142
143

```

code_departement	commune	valeur_moyenne	rang
13	MARSEILLE 14EME	700000	1
13	SAINT SAVOURIN	330850	2
13	GIGNAC-LA-NERTHE	330000	3
33	LEGE-CAP-FERRET	755000	1
33	CESTAS	349900	2

Result 15 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
18	15:44:29	create temporary table valeur_par_commune as select code_departement, commune, round((valeur)/2)	425 row(s) affected Records: 425 Duplicates: 0 Warnings: 0	0.063 sec
19	15:44:29	select*from(select code_departement, commune, valeur_moyenne, rank () over(partition by Code_departement order by valeur_moyenne desc) rang from valeur_par_commune) as resultat where rang <=3 order by code_departement, rang	15 row(s) returned	0.000 sec / 0.000 sec

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

ETYA_Jordan_requetes_022023

```

146
147
148 • select commune, avg(surface_carre) as surface_moyenne, avg(valeur) as valeur_moyenne, round((count(bien.id_bien)*1000/nombre_habitant),2) as
149 'nombre d achat_pour_mille_habitant'
150 from vente
151 join bien on bien.id_bien = vente.id_bien
152 join commune on commune.id_codedep_codecommune = bien.id_codedep_codecommune
153 where nombre_habitant > 10000 and code_departement is not null
154 group by commune
155 order by round((count(bien.id_bien)*1000/nombre_habitant),2) DESC
156 limit 20;
157

```

commune	surface_moyenne	valeur_moyenne	nombre d achat_pour_mille_habitant
PARIS 02	41.74598425196849	522065.05944881885	5.84
PARIS 01	54.393037974683544	1000674.7468354431	4.92
PARIS 03	46.51434782608693	659953.7816770186	4.69
ARCACHON	49.88336363636362	307435.92727272725	4.62
LA BAULE ESCOUBLAC	49.77623376623375	264745.25974025973	4.58
PARIS 04	47.721083333333354	645484.525	4.08
ROQUEBRUNE CAP MARTIN	47.104038461538465	274173.3653846154	3.99
PARIS 08	75.3061151079137	1010536.071942446	3.83
SANARY SUR MER	48.489666666666664	268489.25	3.50
LA LONDE LES MAURES	37.13434343434325	151213.1081081081	3.43
PARIS 09	49.88735576923079	583176.6425961538	3.43

Result 23 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
31	22:23:57	select commune, avg(surface_carre) as surface_moyenne, avg(valeur) as valeur_moyenne, round((count(bien.id_bien)*1000/nombre_habitant),2) as 'nombre d achat_pour_mille_habitant' from vente join bien on bien.id_bien = vente.id_bien join commune on commune.id_codedep_codecommune = bien.id_codedep_codecommune where nombre_habitant > 10000 and code_departement is not null group by commune order by round((count(bien.id_bien)*1000/nombre_habitant),2) DESC limit 20;	20 row(s) returned	0.218 sec / 0.000 sec
32	22:25:06	select commune, avg(surface_carre) as surface_moyenne, avg(valeur) as valeur_moyenne, round((count(bien.id_bien)*1000/nombre_habitant),2) as 'nombre d achat_pour_mille_habitant' from vente join bien on bien.id_bien = vente.id_bien join commune on commune.id_codedep_codecommune = bien.id_codedep_codecommune where nombre_habitant > 10000 and code_departement is not null group by commune order by round((count(bien.id_bien)*1000/nombre_habitant),2) DESC limit 20;	20 row(s) returned	0.218 sec / 0.000 sec