

MySQL Workbench interface showing a query execution result.

Query:

```
-- 1. nombre total d'appartements vendus au 1er semestre 2020
1
2
3 • select type_bien, count(*) as "appartements vendus" from bien, vente
4 where date_vente between "2020-01-01" and "2020-06-30" and bien_id_bien = id_bien and type_bien="appartement";
```

Result Grid:

#	type_bien	appartements vendu
1	Appartement	31203

Action Output:

#	Time	Action	Message	Duration / Fetch
1	22:29:11	select type_bien, count(*) as "appartements vendus" fro...	1 row(s) returned	0,166 sec / 0,00001...

Query Completed

MySQL Workbench interface showing a query execution result.

Query:

```
-- 2. Proportion des ventes d'appartements par nombre de pièces
5
6
7
8 • with
9 table1 as (
10 select nombre_pieces, count(*) as compte from bien, vente
11 where bien_id_bien=id_bien and type_bien="appartement"
12 group by nombre_pieces order by nombre_pieces ),
13 table2 as (
14 select type_bien, count(*) as tot from bien, vente
15 where date_vente between "2020-01-01" and "2020-06-30" and bien_id_bien=id_bien and type_bien="appartement")
16 select nombre_pieces, (compte/tot)*100 as proportion_appartement
17 from table1, table2;
```

Result Grid:

#	nombre_pieces	proportion_apparteme
1	0	0.0961
2	1	21.4851
3	2	31.1220
4	3	28.5806
5	4	14.2390
6	5	3.5606
7	6	0.6538
8	7	0.1731
9	8	0.0545
10	9	0.0256
11	10	0.0064
12	11	0.0032

Action Output:

#	Time	Action	Message	Duration / Fetch
1	22:29:38	with table1 as (select nombre_pieces, count(*) as compt...	12 row(s) returned	0,287 sec / 0,00002...

Query Completed

MySQL Workbench - 27 juil. 22:30

db_immo_mod.mwb - MySQL Workbench

Administration Schemas

SCHEMAS

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

- id_bien varchar(50) PK
- type_bien varchar(50)
- adresse varchar(50)
- nombre_pieces int UN
- surface_batie int UN
- surface_carrez float UN
- surface_terrain int
- commune_id_commune varchar(5)

commune vente - Table bien - Table questionnaire bien - Table Administration - Data Export

Don't Limit

18

19

20 -- 3. Lste des 10 départements où le prix du mètre carré est le plus élevé.

21

22 • select C.id departemen,avg(V.prix_vente/B.surface_carrez)from commune C, vente V, bien B

23 where B.commune_id_commune=C.id_commune and B.id_bien=V.bien_id_bien

24 group by C.id_departement order by max(V.prix_vente/B.surface_carrez) desc limit 10;

25

Result Grid

#	id_departemen	avg(V.prix_vente/B.surface_carrez)
1	75	12045.41972618264
2	94	5340.505849151457
3	45	2973.5890044806953
4	78	4225.253666877494
5	91	3208.4953421578657
6	5	3713.6606870629116
7	74	4667.126654960422
8	6	4696.829125718664
9	59	2857.9204245495903
10	92	7219.388967558968

Result 12

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:29:59	select C.id_departement,avg(V.prix_vente/B.surface_car...	10 row(s) returned	0,217 sec / 0,00002...

Query Completed

MySQL Workbench - 27 juil. 22:30

db_immo_mod.mwb - MySQL Workbench

Administration Schemas

SCHEMAS

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

- id_bien varchar(50) PK
- type_bien varchar(50)
- adresse varchar(50)
- nombre_pieces int UN
- surface_batie int UN
- surface_carrez float UN
- surface_terrain int
- commune_id_commune varchar(5)

commune vente - Table bien - Table questionnaire bien - Table Administration - Data Export

Don't Limit

27

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

29

30 • select avg(V.prix_vente/B.surface_carrez)as px_moyen_maison_ile_de_france from commune C,vente V,bien B

31 where B.commune_id_commune=C.id_commune and B.id_bien=V.bien_id_bien and B.type_bien="maison" and

32 id_departement in(75,77,78,91,92,93,94,95); -- liste des départements de l'Île-de-France

32

Result Grid

#	px_moyen_maison_ile_de_fran
1	3745.0136684942727

Result 13

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:30:35	select avg(V.prix_vente/B.surface_carrez)as px_moyen_...	1 row(s) returned	0,020 sec / 0,00000...

Query Completed

MySQL Workbench interface showing a query execution for the 'questionnaire' table. The query selects the top 10 most expensive apartments based on price and surface area.

```
-- 5. Liste des 10 appartements les plus chers avec le département et le nombre de mètres carrés
select V.prix_vente, C.id_departement, B.surface_carrez as "nb m2" from vente V, bien B, commune C
where B.commune_id_commune = C.id_commune and B.id_bien = V.bien_id_bien and B.type_bien = "appartement"
order by V.prix_vente desc limit 10;
```

Result Grid:

#	prix_vente	id_departement	nb m2
1	9000000	75	9.1
2	8600000	91	64
3	8577713	75	20.55
4	7620000	75	42.77
5	7600000	75	253.3
6	7535000	75	139.9
7	7420000	75	360.95
8	7200000	75	595
9	7050000	75	122.56
10	6600000	75	79.38

Object Info for Table: bien

Columns:

- id_bien varchar(50) PK
- type_bien varchar(50)
- adresse varchar(50)
- nombre_pieces int UN
- surface_batie int UN
- surface_carrez float UN
- surface_terrain int
- commune_id_commune varchar(5)

Action Output:

#	Time	Action	Message	Duration / Fetch
1	22:31:21	select V.prix_vente, C.id_departement, B.surface_carrez...	10 row(s) returned	0,176 sec / 0,00002...

Query Completed

MySQL Workbench interface showing a query execution for the 'questionnaire' table. The query calculates the percentage change in the number of sales between the first and second trimesters of 2020.

```
-- 6. Taux d'évolution du nombre de ventes entre le premier et le second trimestre 2020.
with
table1 as (
select count(date_vente) as q2 from vente
where (date_vente) between "2020-04-01" and "2020-06-30" ),
table2 as (
select count(date_vente) as q1 from vente
where (date_vente) between "2020-01-01" and "2020-03-31" )
select ((q2-q1)/q1*100) as taux_evolution_vente
from table1, table2;
```

Result Grid:

#	taux_evolution_vente
1	3.6779

Object Info for Table: bien

Columns:

- id_bien varchar(50) PK
- type_bien varchar(50)
- adresse varchar(50)
- nombre_pieces int UN
- surface_batie int UN
- surface_carrez float UN
- surface_terrain int
- commune_id_commune varchar(5)

Action Output:

#	Time	Action	Message	Duration / Fetch
1	22:32:08	with table1 as (select count(date_vente) as q2 from vent...	1 row(s) returned	0,032 sec / 0,00001...

Query Completed

MySQL Workbench - 27 juil. 22:32

db_immo_mod.mwb - MySQL Workbench

test MySQL Model EER Diagram

File Edit View Query Database Server Tools Scripting Help

Administration Schemas

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

id_bien varchar(50) PK

type_bien varchar(50)

adresse varchar(50)

nombre_pieces int UN

surface_batie int UN

surface_carrez float UN

surface_terrain int

commune_id_commune varchar(5)

commune

vente - Table

bien - Table

questionnaire

bien - Table

Administration - Data Export

Don't Limit

```

74 -- 7. Liste des communes où le nombre de ventes a augmenté d'au moins 20% entre le premier et le second trimestre de 2020
75
76 • with
77   table1 as (
78     select C.nom_commune,C.id_commune,count(V.date_vente) as vente_trim1 from vente V, bien B,commune C
79     where B.id_bien = V.bien_id_bien and C.id_commune=B.commune_id_commune and V.date_vente between "2020-01-01" and "2020-03-31"
80     group by C.id_commune), -- nombre de ventes au 1er trimestre 2020
81   table2 as (
82     select C.nom_commune,C.id_commune,count(V.date_vente) as vente_trim2 from vente V, bien B,commune C
83     where B.id_bien=V.bien_id_bien and C.id_commune = B.commune_id_commune and V.date_vente between "2020-04-01" and "2020-06-30"
84     group by C.id_commune) -- nombre de ventes au 2eme trimestre 2020
85   select table1.nom_commune,((vente_trim2-vente_trim1)/vente_trim1)*100 as taux_evolution_vente from table1,table2
86   where table1.id_commune=table2.id_commune and (vente_trim2-vente_trim1)/vente_trim1>=0.2
87   group by table1.id_commune; -- calcul d taux d'évolution des ventes par commune en pourcent et affichage si
88                                -- évolution des ventes est d'au moins 20 pourcents.

```

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

#	nom_commune	taux_evolution_vent
1	DIVONNE-LES-BAINS	20.0000
2	LAON	27.2727
3	VILLERS-COTTERETS	66.6667
4	CHATEAU-ARNOUX-SAINT...	100.0000
5	BARCELONNETTE	150.0000
6	SAINT-MARTIN-DE-BROMES	100.0000
7	EMBRUN	100.0000
8	BRIANCON	20.0000
9	ORCIERES	400.0000
10	GAP	200.0000
11	LE DEVOLUY	1000.0000
12	LA SALLE	200.0000
13	RISOUL	100.0000

Result 19

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:32:46	with table1 as (select C.nom_commune,C.id_commune,c...	564 row(s) returned	0,180 sec / 0,00022...

Query Completed

MySQL Workbench - 27 juil. 22:33

db_immo_mod.mwb - MySQL Workbench

test MySQL Model EER Diagram

File Edit View Query Database Server Tools Scripting Help

Administration Schemas

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

id_bien varchar(50) PK

type_bien varchar(50)

adresse varchar(50)

nombre_pieces int UN

surface_batie int UN

surface_carrez float UN

surface_terrain int

commune_id_commune varchar(5)

commune

vente - Table

bien - Table

questionnaire

bien - Table

Administration - Data Export

Don't Limit

```

71 -- 8. Différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces et un appartement de 3 pièces.
72
73 • with
74   table1 as (
75     select avg(prix_vente/surface_carrez) as N2 from vente,bien -- moyenne de prix des appartements de 2 pièces
76     where type_bien="appartement" and nombre_pieces=2 and id_bien=bien_id_bien),
77   table2 as (
78     select avg(prix_vente/surface_carrez) as N3 from vente,bien -- moyenne de prix des appartements de 3 pièces
79     where type_bien="appartement" and nombre_pieces=3 and id_bien=bien_id_bien)
80   select N2,N3,((N2-N3)/N2)*100 as difference_pourcent_Prix_m2
81   from table1,table2;

```

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

#	N2	N3	difference_pourcent_Prix_m
1	4919.990659038932	4309.577471382491	12.406795661999864

Result 20

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:33:11	with table1 as (select avg(prix_vente/surface_carrez) as ...	1 row(s) returned	0,112 sec / 0,00002...

Query Completed

MySQL Workbench - 27 juil. 22:33

db_immo_mod.mwb - MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Administration Schemas

SCHEMAS

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

id_bien varchar(50) PK

type_bien varchar(50)

adresse varchar(50)

nombre_pieces int UN

surface_batie int UN

surface_carrez float UN

surface_terrain int

commune_id_commune varchar(5)

Query Completed

commune vente - Table bien - Table questionnaire bien - Table Administration - Data Export

Don't Limit

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

```

83
84
85 (select id_departement,nom_commune,avg(prix_vente) from vente
86 inner join bien on id_bien = bien_id_bien
87 inner join commune on commune_id_commune=id_commune
88 where id_departement =6 -- 1er département de la liste
89 group by id_departement,nom_commune
90 order by id_departement,avg(prix_vente) desc limit 3)
91 union
92 (select id_departement,nom_commune,avg(prix_vente) from vente
93 inner join bien on id_bien = bien_id_bien
94 inner join commune on commune_id_commune=id_commune
95 where id_departement =13 -- 2eme département de la liste
96 group by id_departement,nom_commune
97 order by id_departement,avg(prix_vente) desc limit 3)
98 union
99 (select id_departement,nom_commune,avg(prix_vente) from vente
100 inner join bien on id_bien = bien_id_bien
101 inner join commune on commune_id_commune=id_commune
102 where id_departement =33 -- 3eme département de la liste
103 group by id_departement,nom_commune
104 order by id_departement,avg(prix_vente) desc limit 3)
105 union
106 (select id_departement,nom_commune,avg(prix_vente) from vente
107 inner join bien on id_bien = bien_id_bien
108 inner join commune on commune_id_commune=id_commune
109 where id_departement =59 -- 4eme département de la liste
110 group by id_departement,nom_commune
111 order by id_departement,avg(prix_vente) desc limit 3)
112 union
113 (select id_departement,nom_commune,avg(prix_vente) from vente
114 inner join bien on id_bien = bien_id_bien
115 inner join commune on commune_id_commune=id_commune
116 where id_departement =69 -- 5eme département de la liste
117 group by id_departement,nom_commune
118 order by id_departement,avg(prix_vente) desc limit 3)

```

MySQL Workbench - 27 juil. 22:34

db_immo_mod.mwb - MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Administration Schemas

SCHEMAS

Filter objects

db_immo

Tables

bien

commune

vente

Views

Stored Procedures

Functions

mydb

sys

Object Info Session

Table: bien

Columns:

id_bien varchar(50) PK

type_bien varchar(50)

adresse varchar(50)

nombre_pieces int UN

surface_batie int UN

surface_carrez float UN

surface_terrain int

commune_id_commune varchar(5)

Query Completed

commune vente - Table bien - Table questionnaire bien - Table Administration - Data Export

Don't Limit

```

93 inner join bien on id_bien = bien_id_bien
94 inner join commune on commune_id_commune=id_commune
95 where id_departement =13 -- 2eme département de la liste
96 group by id_departement,nom_commune
97 order by id_departement,avg(prix_vente) desc limit 3)
98 union
99 (select id_departement,nom_commune,avg(prix_vente) from vente

```

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

#	id_departement	nom_commune	avg(prix_vente)
1	6	SAINT-JEAN-CAP-FERRAT	968750.0000
2	6	EZE	655000.0000
3	6	MOUANS-SARTOUX	476898.1250
4	13	GIGNAC-LA-NERTHE	330000.0000
5	13	SAINT SAVOURNIN	314425.0000
6	13	CASSIS	313416.8750
7	33	LEGE-CAP-FERRET	549500.6364
8	33	VAYRES	335000.0000
9	33	ARCACHON	307435.9273
10	59	BERSEE	433202.0000
11	59	CYSOING	408550.0000
12	59	HALLUIN	322250.0000
13	69	VILLE SUR JARNIOUX	485300.0000
14	69	LYON 2EME	455217.2609
15	69	LYON 6EME	426968.2500

Result 22

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:34:42	(select id_departement,nom_commune,avg(prix_vente) ...	15 row(s) returned	0,043 sec / 0,00002...