

Requête 1 :

```

1  -- requete 1: nombres total d'appartement vendu au 1er semestre 2020
2  SELECT count(DISTINCT vente.Id_bien) as nbr_appt_vendu
3  FROM Vente
4  JOIN Bien on Vente.Id_bien = Bien.Id_bien
5  WHERE Type_local = 'Appartement'
6  AND Date BETWEEN '2020-01-01' AND '2020-06-30'

```

	nbr_appt_vendu
1	31372

Requête 2 :

```

1  -- requete2: nombre de vente d'appartement par région pour le 1er semestre 2020
2  SELECT Nom_region, count(*) as nbs_appt_vendu
3  FROM Vente
4  INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  INNER join Region on Region.Id_region = Commune.Id_region
7  WHERE Type_local = 'Appartement'
8  AND Date BETWEEN '2020-01-01' AND '2020-06-30'
9  GROUP BY Nom_region

```

	Nom_region	nbs_appt_vendu
1	AUVERGNE-RHÔNE-ALPES	3253
2	BOURGOGNE-FRANCHE-COMTÉ	376
3	BRETAGNE	983
4	CENTRE-VAL DE LOIRE	696
5	CORSE	223
6	GRAND EST	984
7	GUADELOUPE	2
8	GUYANE	34
9	HAUTS-DE-FRANCE	1254
10	LA RÉUNION	44

L'exécution s'est terminée sans erreur.
 Résultat : 17 enregistrements ramenés en 132ms

Requête 3 :

```

1  -- requete 3: proportion du nombre de vente d'appartement par le nombre de piece
2  WITH vente_total as (
3      SELECT count(DISTINCT vente.Id_bien) as nbr_appt_vendu
4      FROM Vente
5      JOIN Bien on Vente.Id_bien = Bien.Id_bien
6      WHERE Type_local = 'Appartement')
7  SELECT Total_piece,
8      round(count(DISTINCT vente.Id_bien) * 100.0 / (SELECT nbr_appt_vendu FROM vente_total),2)
9      as prop_vente_appt
10 FROM Vente
11 JOIN Bien on Vente.Id_bien = Bien.Id_bien
12 WHERE Type_local = 'Appartement'
13 GROUP by Total_piece
14 ORDER by Total_piece

```

	Total_piece	prop_vente_appt
1	0	0.1
2	1	21.47
3	2	31.18
4	3	28.57
5	4	14.21
6	5	3.55
7	6	0.65
8	7	0.17
9	8	0.05
10	9	0.03

L'exécution s'est terminée sans erreur.
 Résultat : 12 enregistrements ramenés en 204ms

Requête 4 :

```

1  --requete4: liste des 10 dptment ou le prix du metre carré est le plus élevé
2  SELECT Code_departement, round(avg(Valeur/Surface_carrez), 2) as prix_m2
3  FROM Vente
4  INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  GROUP by Code_departement
7  ORDER by 2 DESC
8  LIMIT 10

```

	Code_departement	prix_m2
1	75	12045.42
2	92	7219.39
3	94	5340.51
4	6	4696.83
5	74	4667.13
6	93	4336.7
7	78	4225.25
8	69	4059.31
9	2A	4010.6
10	33	3764.14

L'exécution s'est terminée sans erreur.
 Résultat : 10 enregistrements ramenés en 120ms

Requête 5 :

```

1  -- requete 5: prix moyen du metre carré d'une maison en Ile de France
2  SELECT Nom_region, round(avg(valeur/Surface_carrez), 2) as prix_moyen
3  FROM Vente
4  INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  INNER JOIN Region on Commune.Id_region = Region.Id_region
7  WHERE Type_local = 'Maison' AND Region.Id_region = 11

```

	Nom_region	prix_moyen
1	ÎLE-DE-FRANCE	3745.01

Requête 6 :

```

1  -- requete 6: liste des 10 appartements les plus chers avec la région et le nombre de metre carré
2  SELECT bien.Id_bien, Nom_region, Surface_carrez, Valeur
3  FROM Bien
4  INNER JOIN Vente on Bien.Id_bien = Vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  INNER JOIN Region on Commune.Id_region = Region.Id_region
7  WHERE Type_local = 'Appartement'
8  ORDER by Valeur DESC
9  LIMIT 10

```

	Id_bien	Nom_region	Surface_carrez	Valeur
1	32267	ÎLE-DE-FRANCE	9.1	9000000
2	21831	ÎLE-DE-FRANCE	64.0	8600000
3	29793	ÎLE-DE-FRANCE	20.55	8577713
4	32425	ÎLE-DE-FRANCE	42.77	7620000
5	29844	ÎLE-DE-FRANCE	253.3	7600000
6	29516	ÎLE-DE-FRANCE	139.9	7535000
7	31965	ÎLE-DE-FRANCE	360.95	7420000
8	32127	ÎLE-DE-FRANCE	595.0	7200000
9	29347	ÎLE-DE-FRANCE	122.56	7050000
10	29507	ÎLE-DE-FRANCE	79.38	6600000

L'exécution s'est terminée sans erreur.
 Résultat : 10 enregistrements ramenés en 87ms

Requête 7 :

```

1  -- requete 7: taux d'évolution du nombre de vente entre le premier et le second trimestre 2020
2  WITH table1 as (
3  |   SELECT count(Vente.Id_vente) as premier_trim FROM Vente
4  |   WHERE Date BETWEEN '2020-01-01' AND '2020-03-31'),
5  table2 as (
6  |   SELECT count(Vente.Id_vente) as deuxieme_trim FROM Vente
7  |   WHERE Date BETWEEN '2020-04-01' AND '2020-06-30')
8  SELECT round(((deuxieme_trim - premier_trim) * 100.0 / premier_trim), 2) as taux_evolution
9  FROM table1, table2

```

	taux_evolution
1	3.68

Requête 8 :

```
1  -- requete 8: classement des régions par rapport au prix au mètre carré des appartements de plus de 4 piece
2  SELECT Nom_region, round(avg(Valeur/Surface_carrez), 2) as prx_m2_appt_plus_4_pieces
3  FROM Vente
4  INNER JOIN Bien on Bien.Id_bien = vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  INNER JOIN Region on Commune.Id_region = Region.Id_region
7  WHERE Type_local = 'Appartement' AND Total_piece > 4
8  GROUP by Nom_region
9  ORDER by 2 DESC
```

	Nom_region	prx_m2_appt_plus_4_pieces
1	ÎLE-DE-FRANCE	8757.64
2	LA RÉUNION	3641.81
3	PROVENCE-ALPES-CÔTE D'AZUR	3587.65
4	CORSE	3104.88
5	AUVERGNE-RHÔNE-ALPES	2891.38
6	NOUVELLE-AQUITAINE	2465.48
7	BRETAGNE	2412.05
8	PAYS DE LA LOIRE	2315.76
9	HAUTS-DE-FRANCE	2189.93
10	OCCITANIE	2097.23

L'exécution s'est terminée sans erreur.
Résultat : 15 enregistrements ramenés en 52ms

Requête 9 :

```

1  -- requete 9: liste des communes ayant eu au moins 50 ventes au 1er trimestre
2  SELECT Nom_commune, count(vente.Id_vente) as nbs_vente
3  FROM Vente
4  INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
5  INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
6  WHERE Date BETWEEN '2020-01-01' AND '2020-03-31'
7  GROUP by Nom_commune
8  HAVING nbs_vente >= 50

```

	Nom_commune	nbs_vente
1	AJACCIO	54
2	ANGERS	64
3	ANTIBES	77
4	ASNIERES-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

L'exécution s'est terminée sans erreur.
Résultat : 48 enregistrements ramenés en 92ms

Requête 10 :

```

1  -- requete 10: différence en pourcentage du prix au mètre carré entre un appartement de 2 pièces
2  -- et un appartement de 3 pièces
3  WITH appt_2_piece as (
4      SELECT Bien.Id_bien, Vente.Id_bien, Surface_local, Valeur, Total_piece,
5          avg(Valeur/Surface_carrez) as prix_m2_2
6      FROM Vente
7      INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
8      WHERE Total_piece = 2),
9  appt_3_piece as (
10     SELECT Bien.Id_bien, Vente.Id_bien, Surface_local, Valeur, Total_piece,
11         avg(Valeur/Surface_carrez) as prix_m2_3
12     FROM Vente
13     INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
14     WHERE Total_piece = 3)
15 SELECT round(((prix_m2_3 - prix_m2_2)/prix_m2_2)*100.0, 2) as dif_en_poucentage
16 FROM appt_2_piece, appt_3_piece

```

	dif_en_poucentage
1	-12.77

Requête 11 :

```

1  -- requete 11: les moyennes de valeurs foncières pour le top 3 des communes des
2  -- département 6, 13, 33, 59 et 69
3  WITH tab1 as
4  (SELECT avg(Valeur) as moy_val_fonciere, Code_departement, Nom_commune
5   FROM Vente
6   INNER JOIN Bien on Bien.Id_bien = Vente.Id_bien
7   INNER JOIN Commune on Bien.Id_commune = Commune.Id_commune
8   WHERE Code_departement in (6,13,33,59,69)
9   GROUP by Code_departement, Nom_commune),
10 tab2 as
11 (SELECT
12  rank()OVER (PARTITION by Code_departement ORDER by moy_val_fonciere DESC) as classement,
13  Code_departement, Nom_commune, round(moy_val_fonciere, 2) as moyenne_valeur_fonciere
14  FROM tab1
15  GROUP by Nom_commune, Code_departement)
16 SELECT classement, Code_departement, Nom_commune, moyenne_valeur_fonciere
17 FROM tab2
18 WHERE classement < 4

```

	classement	Code_departement	Nom_commune	moyenne_valeur_fonciere
1	1	13	GIGNAC-LA-NERTHE	330000.0
2	2	13	SAINT SAVOURNIN	314425.0
3	3	13	CASSIS	313416.88
4	1	33	LEGE-CAP-FERRET	549500.64
5	2	33	VAYRES	335000.0
6	3	33	ARCACHON	307435.93
7	1	59	BERSEE	433202.0
8	2	59	CYSOING	408550.0
9	3	59	HALLUIN	322250.0
10	1	6	SAINT-JEAN-CAP-FERRAT	968750.0

L'exécution s'est terminée sans erreur.
Résultat : 15 enregistrements ramenés en 104ms