



## **Créez et utilisez une base de données immobilière avec SQL**

## DEFINITION DES COMMANDES

COMMANDES	DEFINITIONS
<b>COUNT</b>	« En SQL, la fonction d'agrégation COUNT () permet de compter le nombre d'enregistrement dans une table »
<b>INNER JOIN</b>	« La commande INNER JOIN est un type de jointures très communes pour lier plusieurs tables entre-elle »
<b>WHERE</b>	« La commande WHERE dans une requête SQL permet d'extraire les lignes d'une base de données qui respectent une condition. Cela permet d'obtenir uniquement les informations désirées. »
<b>ORDER BY</b>	« La commande ORDER BY permet de trier les lignes dans un résultat d'une requête SQL. Il est possible de trier les données sur une ou plusieurs colonnes, par ordre ascendant ou descendant. »
<b>GROUP BY</b>	La commande GROUP BY est utilisée en SQL pour grouper plusieurs résultats et utiliser une fonction de totaux sur un groupe de résultat.
<b>DISTINCT</b>	« L'utilisation de la commande <a href="#">SELECT</a> en SQL permet de lire toutes les données d'une ou plusieurs colonnes. Cette commande peut potentiellement afficher des lignes en doubles. Pour éviter des redondances dans les résultats il faut simplement ajouter DISTINCT après le mot SELECT. »
<b>ROUND</b>	« La fonction ROUND () permet d'arrondir un résultat numérique. Cette fonction permet soit d'arrondir sans utiliser de décimal pour retourner un nombre entier (c'est-à-dire : aucun chiffre après la virgule), ou de choisir le nombre de chiffre après la virgule. »
<b>AVG</b>	« La fonction d'agrégation AVG () dans le langage SQL permet de calculer une valeur moyenne sur un ensemble d'enregistrement de type numérique et non nul. »
<b>SUM</b>	« La fonction d'agrégation SUM () permet de calculer la somme totale d'une colonne contenant des valeurs numériques. Cette fonction ne fonctionne que sur des colonnes de types numériques (INT, FLOAT ...) et n'additionne pas les valeurs NULL. »
<b>QUARTER</b>	« La fonction QUARTER () renvoie le trimestre de l'année pour une valeur de date donnée (un nombre de 1 à 4). »
<b>HAVING</b>	« La condition HAVING en SQL est presque similaire à WHERE à la seule différence que HAVING permet de filtrer en utilisant des fonctions telles que SUM (), COUNT (), AVG (), MIN () ou MAX(). »
<b>CASE</b>	« La commande "CASE ... WHEN ..." permet d'utiliser des conditions de type "si / sinon" (cf. if / else) similaire à un langage de programmation pour retourner un résultat disponible entre plusieurs possibilités. Le CASE peut être utilisé dans n'importe quelle instruction ou clause, telle que SELECT, UPDATE, DELETE, WHERE, ORDER BY ou HAVING. »
<b>SELECT</b>	« L'utilisation la plus courante de SQL consiste à lire des données issues de la base de données. Cela s'effectue grâce à la commande SELECT, qui retourne des enregistrements dans un tableau de résultat. Cette commande peut sélectionner une ou plusieurs colonnes d'une table. »

**Source :** <https://sql.sh/>

# REQUETES ET RESULTATS

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

```
SELECT COUNT(logement.TYPE_LOCAL='Appartement') as 'Total' FROM `logement`
INNER JOIN mutation ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT
WHERE logement.TYPE_LOCAL='Appartement' AND mutation.NATURE_MUTATION='Vente'
```

Total
31378

## 2. Proportion des ventes d'appartements par le nombre de pièces.

```
SELECT logement.NOMBRE_PIECES, ((COUNT(logement.NOMBRE_PIECES) / (SELECT COUNT(logement.NOMBRE_PIECES) FROM `logement` INNER JOIN mutation ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT WHERE `TYPE_LOCAL`='Appartement' AND `NATURE_MUTATION`='Vente')) * 100) AS 'Proportion'
FROM logement INNER JOIN mutation ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT
WHERE `TYPE_LOCAL`='Appartement' AND `NATURE_MUTATION`='Vente' GROUP BY logement.NOMBRE_PIECES
ORDER BY logement.NOMBRE_PIECES ASC
```

<u>NOMBRE_PIECES_1</u>	<u>Proportion</u>
0	0.0956
1	21.4768
2	31.1779
3	28.5742
4	14.2138
5	3.5503
6	0.6501
7	0.1721
8	0.0542
9	0.0255
10	0.0064
11	0.0032

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

```
SELECT adresse.CODE_DEPARTEMENT,AVG(mutation.VALEUR_FONCIERE/ logement.SURFACE_REELLE_BATI) as
'Prixm2'

FROM `adresse` inner JOIN logement ON adresse.ID_ADRESSE=logement.ID_ADRESSE inner JOIN mutati
on ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT

WHERE logement.SURFACE_REELLE_BATI!=0 AND adresse.CODE_DEPARTEMENT!=0

GROUP BY adresse.CODE_DEPARTEMENT ORDER BY Prixm2 DESC LIMIT 10
```

CODE DEPARTEMENT	Prixm2_1
75	12121.880704294932
92	7415.2848907089965
94	5395.977019856416
6	4681.760125263882
93	4363.013878565967
74	4149.556962802092
78	4126.185880300853
69	4063.8323873628415
33	3838.7434542483093
83	3792.7460811040296

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

```
SELECT AVG(mutation.VALEUR_FONCIERE/ logement.SURFACE_REELLE_BATI) AS 'Prixm2'

FROM adresse INNER JOIN logement ON adresse.ID_ADRESSE=logement.ID_ADRESSE inner JO
IN mutation ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT

WHERE logement.TYPE_LOCAL='Maison' AND mutation.NATURE_MUTATION='Vente' AND adresse
.CODE_DEPARTEMENT IN (75,77,78,91,92,93,94,95)
```

Prixm2
3997.711701859063

5. Liste des 10 appartements les plus chers avec le département et le nombre de mètres carrés.

```
SELECT adresse.CODE_DEPARTEMENT,mutation.VALEUR_FONCIERE,logement.SURFACE_REELLE_BATI
FROM `adresse` inner JOIN logement ON adresse.ID_ADRESSE=logement.ID_ADRESSE inner JOIN mutati
on ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT
WHERE logement.TYPE_LOCAL='Appartement' AND mutation.NATURE_MUTATION='Vente'
ORDER BY mutation.VALEUR_FONCIERE DESC LIMIT 10
```

CODE DEPARTEMENT	VALEUR FONCIERE _1	SURFACE REELLE BATI
75	9000000	10
91	8600000	62
75	8577713	289
75	7620000	42
75	7600000	200
75	7535000	143
75	7420000	357
75	7200000	241
75	7050000	310
75	6600000	76

6. Taux d'évolution du nombre de ventes entre le premier et le deuxième trimestre de 2020.

```
SELECT ( SELECT DISTINCT COUNT(mutation.ID_LOGEMENT) FROM `mutation`
INNER JOIN logement ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT
WHERE mutation.NATURE_MUTATION='Vente' AND QUARTER(mutation.DATE_MUTATION)=1) as 'Vente1',
(SELECT DISTINCT COUNT(mutation.ID_LOGEMENT) FROM mutation inner JOIN logement ON logement.ID_
LOGEMENT=mutation.ID_LOGEMENT
WHERE mutation.NATURE_MUTATION='Vente' AND QUARTER(mutation.DATE_MUTATION)=2) AS 'Vente2', (SEL
ECT (Vente2 - Vente1) / Vente1*100) AS 'Taux'
```

Vente1	Vente2	Taux
16776	17393	3.6779

**7. Liste des communes où le taux d'évolution des ventes est supérieur à 20 % entre le premier et le second semestre de 2020.**

```
SELECT adresse.COMMUNE,AVG(mutation.VALEUR_FONCIERE) as 'Moyenne'
FROM `adresse` INNER JOIN logement ON logement.ID_ADRESSE=adresse.ID_ADRESSE INNER JOIN mutati
on ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT
WHERE QUARTER(mutation.DATE_MUTATION)=2 GROUP BY adresse.COMMUNE
HAVING AVG(mutation.VALEUR_FONCIERE)>=ALL(SELECT 1.2*AVG(mutation.VALEUR_FONCIERE) FROM mutati
on
WHERE QUARTER(mutation.DATE_MUTATION)=1) ORDER BY adresse.COMMUNE
```

<b>COMMUNE_1</b>	<b>Moyenne</b>
ACHERES-LA-FORET	323500
AIME LA PLAGNE	535160
ARCHAMPS	315000
ASNIERES-SUR-SEINE	370250
ASSAS	508000
AUVERS-SUR-OISE	369666.6666666667
BAGNOLET	583000
BAILLY	349102.3333333333
BARROUX (LE)	325000
BEAULIEU SUR MER	411000
BERSEE	433202
BONDUES	318150
BOURG-LA-REINE	325604
BRIIS-SOUS-FORGES	352600
BRY-SUR-MARNE	321603.75
BUC	324000
CAGNES SUR MER	311185
CANTARON	387000
CAP-D AIL	485000
CARRIERES SUR SEINE	386625
CASSIS	309580
CELLE-SAINT-CLOUD (LA)	462792

<b><u>COMMUNE_1</u></b>	<b><u>Moyenne</u></b>
CERNAY-LA-VILLE	495000
CHAMBOURCY	337266.6666666667
CHARENTON-LE-PONT	452600.59375
CHATEAUFORT	390400
CHATENAY-MALABRY	570271.1538461539
CHATILLON	379445.3846153846
CHENEX	364000
CHEVILLY-LARUE	314200
CLAIX	361800
CLICHY	354589.73846153845
COLLONGES-SOUS-SALEVE	461215
COLOMBES	395912.5
COMBLOUX	311933.3333333333
COTI-CHIAVARI	473000
COURBEVOIE	380944.843373494
CRESPIERES	632065
CROISSY SUR SEINE	360426.5
CYSOING	408550
DIVONNE-LES-BAINS	353143.3333333333
DOMMARTIN	319000
DUINGT	610800
ECULLY	314349.4444444444
ETANG-LA-VILLE (L )	401307
ETIOLLES	330000
EVENOS	619900
EZE	330000
FEUCHEROLLES	540750
FLOURENS	316484
FONTENAY-SOUS-BOIS	319871.71794871794

<b><u>COMMUNE_1</u></b>	<b><u>Moyenne</u></b>
FREPILLON	354000
GENTILLY	311486.36363636365
GOMETZ-LE-CHATEL	355000
GRIMAUD	343166.6666666667
ISSY-LES-MOULINEAUX	409436.32352941175
IVRY-SUR-SEINE	310656.8965517241
JOUY EN JOSAS	420860.4285714286
JUVIGNAC	332000
LA ROQUETTE SUR SIAGNE	510037.5
LANGUIDIC	450000
LE CASTELLET	420000
LE CHESNAY-ROCQUENCOURT	370861.4117647059
LE KREMLIN-BICETRE	310726.6666666667
LE PLESSIS-ROBINSON	352966.2
LECCI	310179.28571428574
LEGE-CAP-FERRET	511744.54545454547
LEVALLOIS-PERRET	517230.2727272727
LYON 1ER	315588.8695652174
LYON 2EME	455209.4358974359
LYON 3EME	341525.1923076923
LYON 4EME	412550.73170731706
LYON 6EME	369262.9166666667
MAISONS-LAFFITTE	310253.3333333333
MANDRES-LES-ROSES	352512.5
MAREIL-MARLY	369512.5
MEGEVE	514585.85714285716
MENTHON-SAINT-BERNARD	330000
MESSY	360000
MONTLIGNON	323000



<u>COMMUNE_1</u>	<u>Moyenne</u>
MONTREUIL	313069.602739726
MONTRouGE	375795.2162162162
MOUANS-SARTOUX	807092.5
MOUVAUX	332500
NANTERRE	393261.21875
NEUILLY-SUR-SEINE	845679.3684210526
NOGENT-SUR-MARNE	354386.5625
NOISEAU	357733.3333333333
NOISY LE ROI	451725
NOISY-SUR-ECOLE	373000
NOZAY	323463.5
ORGEVAL	340090
ORNEX	410000
PARIS 01	1048644.3333333333
PARIS 02	553876.7727272727
PARIS 03	595616.256097561
PARIS 04	687730.6333333333
PARIS 05	559629.2666666667
PARIS 06	970099.8113207547
PARIS 07	1074643.8125
<u>COMMUNE_1</u>	<u>Moyenne</u>
PARIS 08	1023770.1948051949
PARIS 09	545135.931372549
PARIS 10	542897.9870967742
PARIS 11	447271.08411214955
PARIS 12	460409.0416666667
PARIS 13	535715.9459459459
PARIS 14	509105.4605263158
PARIS 15	489762.234375

<b><u>COMMUNE_1</u></b>	<b><u>Moyenne</u></b>
PARIS 16	998841.5414847161
PARIS 17	704924.652892562
PARIS 18	437982.6384364821
PARIS 19	397304.94701986754
PARIS 20	418395.8522727273
PERIGNY	333010
POISAT	326570
PORTO-VECCHIO	377884.8
PREVESSIN-MOENS	422000
PUTEAUX	350971.1666666667
ROCHETAILLEE-SUR-SAONE	540000
RUEIL-MALMAISON	330868.83720930235
SAINT TROPEZ	448875
SAINT-CYR-SOUS-DOURDAN	343000
SAINT-ETIENNE-DE-CROSSEY	332500
SAINT-GERMAIN-EN-LAYE	362989
SAINT-JEAN-CAP-FERRAT	622500
SAINT-JEAN-DE-BEAUREGARD	333400
SAINT-MANDE	405809.5
SAINT-MAUR-DES-FOSSES	311573.93333333335
SAINT-NAZAIRE-LES-EYMES	367000
SAINT-NOM-LA-BRETECHE	430130
SAINT-OUEN-SUR-SEINE	345302.96428571426
SAINT-SAUVEUR-SUR-ECOLE	490000
SAINTE FOY LES LYON	358283.3333333333
SAINTE-AGNES	345000
SANTA-MARIA-SICHE	530000
SATHONAY-CAMP	314950
SATHONAY-VILLAGE	312833.3333333333

<b><u>COMMUNE_1</u></b>	<b><u>Moyenne</u></b>
SCEAUX	332860
SERGY	472025
ST MAURICE	318206.27777777775
ST-GENIS-POUILLY	442667.5
SURESNES	457039.3243243243
TASSIN-LA-DEMI-LUNE	336861.3157894737
THOIRY	308750
TOURRETTES	348000
TOUSSIEU	339000
TRESSERVE	312000
VAL-D ISERE	720000
VANVES	390333.5714285714
VAULNAVEYS-LE-BAS	345000
VAYRES	335000
VERRIERES LE BUISSON	380936.8333333333
VERSAILLES	428917.7
VERTOU	330000
VESINET (LE)	330257.14285714284
VILLEFRANCHE-SUR-MER	427800
VILLENNES-SUR-SEINE	359416.6666666667
VINCENNES	447710.23287671234
VIRY	379350

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

```
SELECT (SELECT round(SUM(mutation.VALEUR_FONCIERE) / SUM(logement.SURFACE_REELLE_BATI),2) as 'P
M'
FROM `mutation` INNER JOIN logement ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT
WHERE mutation.NATURE_MUTATION='Vente' AND logement.TYPE_LOCAL='Appartement' AND logement.NOMB
RE_PIECES=2) AS P2,
(SELECT round(SUM(mutation.VALEUR_FONCIERE) / SUM(logement.SURFACE_REELLE_BATI),2) AS 'PM'
FROM mutation INNER JOIN logement ON mutation.ID_LOGEMENT=logement.ID_LOGEMENT
WHERE mutation.NATURE_MUTATION='Vente' AND logement.TYPE_LOCAL='Appartement' AND logement.NOMB
RE_PIECES=3) AS P3, (SELECT (P3-P2)/P2*100) AS 'Taux'
```

P2	P3	Taux
4763.41	4228.27	-11.234389

#### 9. Taux d'appartements qui ont été vendus à un prix du mètre carré deux fois plus élevé que le prix du mètre carré moyen du département.

```
SELECT adresse.CODE_DEPARTEMENT, AVG(mutation.VALEUR_FONCIERE/ logement.SURFACE_REELLE_BATI)
as PrixM
FROM adresse INNER JOIN logement ON adresse.ID_ADRESSE= logement.ID_ADRESSE INNER JOIN
mutation ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT
WHERE logement.SURFACE_REELLE_BATI != 0 GROUP BY adresse.CODE_DEPARTEMENT,
(SELECT logement.ID_LOGEMENT,adresse.COMMUNE, adresse.CODE_DEPARTEMENT,
(mutation.VALEUR_FONCIERE/ logement.SURFACE_REELLE_BATI) as PrixM2, PrixM, CASE WHEN
(mutation.VALEUR_FONCIERE/ logement.SURFACE_REELLE_BATI) > 2 * PrixM THEN 1 ELSE 0 END as
Prix2sup
FROM adresse INNER JOIN logement ON adresse.ID_ADRESSE= logement.ID_ADRESSE INNER JOIN
mutation ON logement.ID_LOGEMENT=mutation.ID_LOGEMENT WHERE (mutation.VALEUR_FONCIERE/
logement.SURFACE_REELLE_BATI) > 2 * PrixM AND logement.SURFACE_REELLE_BATI != 0)
```

10. Donnez les moyennes de valeurs foncières pour le top 20 des communes.

```
SELECT adresse.COMMUNE, AVG(mutation.VALEUR_FONCIERE) AS 'MoyenneVF' FROM `adresse`
INNER JOIN logement ON logement.ID_ADRESSE=adresse.ID_ADRESSE INNER JOIN mutation ON mutation.
ID_LOGEMENT=logement.ID_LOGEMENT
GROUP BY adresse.COMMUNE
ORDER BY MoyenneVF DESC limit 20
```

<u>COMMUNE</u>	<u>MoyenneVF_1</u>
GARRIGUES SAINTE EULALIE	1683000
LES PORTES-EN-RE	1400000
VERNEUIL EN HALATTE	1280000
PARIS 07	1251288.4
PARIS 16	1028243.692893401
PARIS 08	1010536.071942446
PARIS 06	1005955.5683453238
PARIS 01	1000674.7468354431
SAINT-JEAN-CAP-FERRAT	968750
VAL-D ISERE	880000
NEUILLY-SUR-SEINE	813661.0731707317
PARIS 17	720622.8957446809
BARBIZON	713500
PARIS 03	655953.7826086957
EZE	655000
PARIS 04	645484.525
RAMATUELLE	633000
CRESPIERES	632065
GARCHES	615348.5714285715
DUINGT	610800