

## Quel est le nombre de retours clients sur la livraison ?

```
select count(*) AS 'nombre de retours sur la livraison'
```

```
from retour_client
```

```
where libelle_categorie = 'livraison'
```

```
1 • select count(*) AS 'nombre de retours sur la livraison'
2   from retour_client
3   where libelle_categorie = 'livraison'
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	nombre de retours sur la livraison			
▶	639			

## Quelle est la liste des notes des clients sur les réseaux sociaux sur les TV ?

```
SELECT r.note, r.libelle_source, p.titre_produit
```

```
FROM retour_client AS r
```

```
JOIN produit as p ON r.cle_produit = p.cle_produit
```

```
WHERE r.libelle_source = 'réseaux sociaux'
```

```
AND p.titre_produit = 'TV'
```

```
1  SELECT r.note, r.libelle_source, p.titre_produit
2  FROM retour_client AS r
3  JOIN produit as p ON r.cle_produit = p.cle_produit
4  WHERE r.libelle_source = 'réseaux sociaux'
5  AND p.titre_produit = 'TV'
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	note	libelle_source	titre_produit	
▶	10	réseaux sociaux	TV	
	10	réseaux sociaux	TV	
	9	réseaux sociaux	TV	
	8	réseaux sociaux	TV	

### 3) Quelle est la note moyenne pour chaque catégorie de produit ? (Classé de la meilleure à la moins bonne)

```
SELECT p.typologie_produit, AVG(r.note) AS note_moyenne
```

```
FROM retour_client as r
```

```
JOIN produit as p ON r.cle_produit = p.cle_produit
```

```
GROUP BY p.typologie_produit
```

```
ORDER BY note_moyenne DESC
```

#### Ou bien

```
SELECT p.titre_produit, AVG(r.note) AS note_moyenne
```

```
FROM retour_client as r
```

```
JOIN produit as p ON r.cle_produit = p.cle_produit
```

```
GROUP BY p.titre_produit
```

```
ORDER BY note_moyenne DESC;
```

```
1 • SELECT p.typologie_produit, AVG(r.note) AS note_moyenne
2 FROM retour_client as r
3 JOIN produit as p ON r.cle_produit = p.cle_produit
4 GROUP BY p.typologie_produit
5 ORDER BY note_moyenne DESC;
6
```

	typologie_produit	note_moyenne
►	High-Tech	8.1607
	Loisirs	8.0904
	Alimentaire	8.0418
	Maison	7.8507

```
1 SELECT p.titre_produit, AVG(r.note) AS note_moyenne
2 FROM retour_client as r
3 JOIN produit as p ON r.cle_produit = p.cle_produit
4 GROUP BY p.titre_produit
5 ORDER BY note_moyenne DESC;
```

	titre_produit	note_moyenne
►	Plantes aromatiques surgelées	9.2000
	Sodas	9.0000
	Boissons alcoolisées	8.7895
	Sauces au soja	8.7727
	TV	8.7143
	Aliments à base de plantes frais	8.7143

#### 4) Quels sont les 5 magasins avec les meilleurs notes moyennes ?

```
select m.ref_magasin, m.libelle_de_commune, avg(r.note) as note_moyenne
```

```
from ref_magasin_2 as m
```

```
join retour_client as r on m.ref_magasin = r.ref_magasin
```

```
group by m.ref_magasin
```

```
order by note_moyenne desc
```

```
limit 5
```

```
1  select m.ref_magasin, m.libelle_de_commune, avg(r.note) as note_moyenne
2  from ref_magasin_2 as m
3  join retour_client as r on m.ref_magasin = r.ref_magasin
4  group by m.ref_magasin
5  order by note_moyenne desc
6  limit 5
```

	ref_magasin	libelle_de_commune	note_moyenne
▶	75	Paris 14e Arrondissement	8.7273
	78	Saint-Pierre-du-Perray	8.5484
	62	Paris 19e Arrondissement	8.5000
	23	Paris 11e Arrondissement	8.4839
	19	Coulommiers	8.4524

#### 5) Quels sont les magasins qui ont plus de 12 feedbacks sur le Drive ?

```
select count(*) as nombre_de_feedback_drive, m.ref_magasin, m.libelle_de_commune as magasin, m.departement
```

```
from retour_client as r
```

```
join ref_magasin_2 as m on r.ref_magasin = m.ref_magasin
```

```
where r.libelle_categorie = 'drive'
```

```
group by ref_magasin
```

```
having nombre_de_feedback_drive >= 12
```

```
1  select count(*) as nombre_de_feedback_drive, m.ref_magasin, m.libelle_de_commune as magasin, m.departement
2  from retour_client as r
3  join ref_magasin_2 as m on r.ref_magasin = m.ref_magasin
4  where r.libelle_categorie = 'drive'
5  group by ref_magasin
6  having nombre_de_feedback_drive > 12
```

	nombre_de_feedback_drive	ref_magasin	magasin	departement
▶	14	67	Éragny	95
	13	63	Ivry-sur-Seine	94
	13	45	Paris 12e Arrondissement	75

## 6) Quel est le classement des départements par note ?

```
select avg(r.note) as note_moyenne, m.departement
```

```
from retour_client as r
```

```
join ref_magasin_2 as m on r.ref_magasin = m.ref_magasin
```

```
group by m.departement
```

```
order by note_moyenne desc
```

```
1  select avg(r.note) as note_moyenne, m.departement
2  from retour_client as r
3  join ref_magasin_2 as m on r.ref_magasin = m.ref_magasin
4  group by m.departement
5  order by note_moyenne desc
```

note_moyenne	departement
8.1388	95
8.1076	75
8.0567	94
8.0466	91
8.0420	77
8.0278	92
8.0169	78
7.9377	93

## 7) Quelle est la typologie de produit qui apporte le meilleur service après-vente ?

```
select p.typologie_produit, avg(r.note) as note
```

```
from produit as p
```

```
join retour_client as r on p.cle_produit = r.cle_produit
```

```
where libelle_categorie = 'service après-vente'
```

```
group by p.typologie_produit
```

```
order by note desc
```

```
limit 1
```

```
1  select p.typologie_produit, avg(r.note) as note
2  from produit as p
3  join retour_client as r on p.cle_produit = r.cle_produit
4  where libelle_categorie = 'service après-vente'
5  group by p.typologie_produit
6  order by note desc
7  limit 1
8
```

typologie_produit	note
Logiciels	8.5135

## 8) Quelle est la note moyenne sur l'ensemble des boissons ?

```
select avg(r.note) as note_moyenne_sur_les_boissons
```

```
from retour_client as r
```

```
join produit as p on r.cle_produit = p.cle_produit
```

```
where p.titre_produit like 'Boisson%'
```

```
1  select avg(r.note) as note_moyenne_sur_les_boissons
2  from retour_client as r
3  join produit as p on r.cle_produit = p.cle_produit
4  where p.titre_produit like 'Boisson%'
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
note_moyenne_sur_les_boissons			
8.3208			

## 9) Quel est le classement des jours de la semaine où l'expérience client est la meilleure expérience en magasin ?

```
SELECT (DAYNAME(date_achat)) AS jour_semaine, avg(note) as note_moyenne,  
count(libelle_categorie) as nombre_de_retours
```

```
FROM retour_client
```

```
where libelle_categorie = 'expérience en magasin'
```

```
group by jour_semaine
```

```
ORDER by note_moyenne desc
```

```
1 • SELECT
2     (DAYNAME(date_achat)) AS jour_semaine,
3     AVG(note) AS note_moyenne,
4     COUNT(libelle_categorie) AS nombre_de_retours
5 FROM
6     retour_client
7 WHERE
8     libelle_categorie = 'expérience en magasin'
9 GROUP BY jour_semaine
10 ORDER BY note_moyenne DESC
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
jour_semaine	note_moyenne	nombre_de_retours	
Saturday	8.3425	73	
Sunday	8.1795	78	
Friday	8.0690	87	
Thursday	8.0400	75	
Wednesday	7.9868	76	
Tuesday	7.9535	86	
Monday	7.7407	81	

## 10) Sur quel mois a-t-on le plus de retour sur le service après-vente ?

SELECT

(MONTHNAME(date\_achat)) AS mois,

COUNT(libelle\_categorie) AS nombre\_de\_retours\_sav

FROM

retour\_client

WHERE

libelle\_categorie = 'service après-vente'

GROUP BY mois

ORDER BY nombre\_de\_retours\_sav DESC

Limit 1

```
1  SELECT
2      (MONTHNAME(date_achat)) AS mois,
3      COUNT(libelle_categorie) AS nombre_de_retours_sav
4  FROM
5      retour_client
6  WHERE
7      libelle_categorie = 'service après-vente'
8  GROUP BY mois
9  ORDER BY nombre_de_retours_sav DESC
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	mois	nombre_de_retours_sav			
▶	October	55			
	September	53			
	June	53			
	August	52			
	March	52			
	November	52			
	January	52			
	May	52			
	April	52			
	July	48			
	February	44			
	December	38			

11) Quel est le pourcentage de recommandations client ? Comptabiliser le nombre de retours client qui ont répondu « Oui » divisé par le nombre de retours total

SELECT

(COUNT(CASE WHEN recommandation = '1' THEN 1 END) / SUM(CASE WHEN recommandation IN ('1', '0') THEN 1 ELSE 0 END) \* 100) AS pourcentage\_recommandation

FROM

retour\_client

```
1 • SELECT
2 (COUNT(CASE WHEN recommandation = '1' THEN 1 END) / SUM(CASE WHEN recommandation IN ('1', '0') THEN 1 ELSE 0 END) * 100) AS pourcentage_recommandation
3 FROM
4 retour_client
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
pourcentage_recommandation			
90.9286			

## 12) Quels sont les magasins qui ont une note inférieure à la moyenne ?

SELECT

m.ref\_magasin,

m.libelle\_de\_commune,

AVG(r.note) AS moyenne\_note\_magasins,

(SELECT AVG(note) FROM retour\_client) AS moyenne\_globale

FROM

retour\_client AS r

JOIN

ref\_magasin\_2 AS m ON r.ref\_magasin = m.ref\_magasin

GROUP BY

m.ref\_magasin, m.libelle\_de\_commune

HAVING

AVG(r.note) < (SELECT AVG(note) FROM retour\_client)

Order by moyenne\_note\_magasin asc

```
1 • SELECT
2     m.ref_magasin,
3     m.libelle_de_commune,
4     AVG(r.note) AS moyenne_note_magasins,
5     (SELECT AVG(note) FROM retour_client) AS moyenne_globale
6 FROM
7     retour_client AS r
8 JOIN
9     ref_magasin_2 AS m ON r.ref_magasin = m.ref_magasin
10 GROUP BY
11     m.ref_magasin, m.libelle_de_commune
12 HAVING
13     AVG(r.note) < (SELECT AVG(note) FROM retour_client);
```

Result Grid				
Filter Rows:		Export:	Wrap Cell Content:	
	ref_magasin	libelle_de_commune	moyenne_note_magasins	moyenne_globale
▶	47	Freneuse	8.0286	8.0527
	6	Osny	7.8980	8.0527
	54	Paris 2e Arrondissement	7.8378	8.0527
	3	Paris 5e Arrondissement	8.0357	8.0527
	18	Saint-Germain-lès-Arpajon	7.8293	8.0527
	76	Carrières-sur-Seine	7.7419	8.0527
	14	Paris 17e Arrondissement	7.9048	8.0527
	46	Paris 15e Arrondissement	7.5556	8.0527
	8	Aubervilliers	7.6571	8.0527
	64	Osny-la-Ferrière	7.9286	8.0527



### 13) Quelles sont les typologies produits qui ont amélioré leur moyenne entre le 1<sup>er</sup> et le 2<sup>ème</sup> trimestre 2021 ?

Faisable avec titre et typologie de produit

SELECT

p.typologie\_produit,

AVG(CASE WHEN r.date\_achat BETWEEN '2021-01-01' AND '2021-03-31' THEN r.note END) AS moyenne\_1er\_trimestre,

AVG(CASE WHEN r.date\_achat BETWEEN '2021-04-01' AND '2021-06-30' THEN r.note END) AS moyenne\_2eme\_trimestre,

AVG(CASE WHEN r.date\_achat BETWEEN '2021-04-01' AND '2021-06-30' THEN r.note END) -

AVG(CASE WHEN r.date\_achat BETWEEN '2021-01-01' AND '2021-03-31' THEN r.note END) AS delta

FROM

retour\_client AS r

JOIN

produit AS p ON r.cle\_produit = p.cle\_produit

WHERE

r.date\_achat BETWEEN '2021-01-01' AND '2021-06-30'



GROUP BY

p.typologie\_produit

having delta > 0

order by delta desc

```
1 • SELECT
2     p.titre_produit,
3     AVG(CASE WHEN r.date_achat BETWEEN '2021-01-01' AND '2021-03-31' THEN r.note END) AS moyenne_1er_trimestre,
4     AVG(CASE WHEN r.date_achat BETWEEN '2021-04-01' AND '2021-06-30' THEN r.note END) AS moyenne_2eme_trimestre,
5     AVG(CASE WHEN r.date_achat BETWEEN '2021-04-01' AND '2021-06-30' THEN r.note END) -
6     AVG(CASE WHEN r.date_achat BETWEEN '2021-01-01' AND '2021-03-31' THEN r.note END) AS delta
7 FROM
8     retour_client AS r
9 JOIN
10    produit AS p ON r.cle_produit = p.cle_produit
11 WHERE
12    r.date_achat BETWEEN '2021-01-01' AND '2021-06-30'
13 GROUP BY
14    p.titre_produit
15    order by delta desc
```

Result Grid				
Filter Rows:		Export:  Wrap Cell Content: 		
titre_produit	moyenne_1er_trimestre	moyenne_2eme_trimestre	delta	
Chips	5.8333	8.3750	2.5417	
Maquereaux	6.3333	8.8333	2.5000	
Sucres de canne	6.5000	9.0000	2.5000	
Tomates	6.4000	8.4000	2.0000	
Mangues au sirop	7.0000	9.0000	2.0000	
Son	6.5000	8.5000	2.0000	
...	...	...	...	

Result 22 x

#### 14) calcul du NPS

```
select promoteur*100/total - detracteur*100/total as NPS
```

```
FROM
```

```
(
```

```
SELECT
```

```
sum(case when note > 8
```

```
  then 1
```

```
  else 0 end) as promoteur,
```

```
sum(case when note <= 6
```

```
  then 1
```

```
  else 0 end) as detracteur,
```

```
count(note) as total
```

```
FROM retour_client ) as total
```

The screenshot shows a SQL query editor with a query that calculates the Net Promoter Score (NPS). The query is as follows:

```
1 • select promoteur*100/total - detracteur*100/total as NPS
2 FROM
3 (
4   SELECT
5     sum(case when note > 8
6       then 1
7       else 0 end) as promoteur,
8
9     sum(case when note <= 6
10      then 1
11      else 0 end) as detracteur,
12     count(note) as total
13   FROM retour_client ) as total
14
```

Below the query editor, there is a toolbar with the following options: Result Grid, Filter Rows, Export, and Wrap Cell Content. The Result Grid shows the following data:

	NPS
▶	30.9667

# Analyses supplémentaires

## Évaluation de la Satisfaction Client par Magasin (par trimestre, mesuré en NPS)

SELECT

m.ref\_magasin,

m.libelle\_de\_commune,

ROUND(

(

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-01-01' AND '2021-03-31' AND r.note >= 9  
THEN 1 ELSE 0 END) / COUNT(\*)) -

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-01-01' AND '2021-03-31' AND r.note <= 6  
THEN 1 ELSE 0 END) / COUNT(\*))

) \* 100

) AS NPS\_1er\_trimestre,

ROUND(

(

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-04-01' AND '2021-06-30' AND r.note >= 9  
THEN 1 ELSE 0 END) / COUNT(\*)) -

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-04-01' AND '2021-06-30' AND r.note <= 6  
THEN 1 ELSE 0 END) / COUNT(\*))

) \* 100

) AS NPS\_2eme\_trimestre,

ROUND(

(

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-07-01' AND '2021-09-30' AND r.note >= 9  
THEN 1 ELSE 0 END) / COUNT(\*)) -

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-07-01' AND '2021-09-30' AND r.note <= 6  
THEN 1 ELSE 0 END) / COUNT(\*))

) \* 100

) AS NPS\_3eme\_trimestre,

ROUND(

(

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-10-01' AND '2021-12-31' AND r.note >= 9  
THEN 1 ELSE 0 END) / COUNT(\*)) -

(SUM(CASE WHEN r.date\_achat BETWEEN '2021-10-01' AND '2021-12-31' AND r.note <= 6  
THEN 1 ELSE 0 END) / COUNT(\*))

```

    ) * 100
  ) AS NPS_4eme_trimestre
FROM
  retour_client AS r
JOIN
  ref_magasin_2 AS m ON r.ref_magasin = m.ref_magasin
WHERE
  r.date_achat BETWEEN '2021-01-01' AND '2021-12-31'
GROUP BY
  m.ref_magasin, m.libelle_de_commune

```

```

24      (SUM(CASE WHEN r.date_achat BETWEEN '2021-10-01' AND '2021-12-31' AND r.note >= 9 THEN 1 ELSE 0 END) / COUNT(*)) -
25      (SUM(CASE WHEN r.date_achat BETWEEN '2021-10-01' AND '2021-12-31' AND r.note <= 6 THEN 1 ELSE 0 END) / COUNT(*))
26    ) * 100
27  ) AS NPS_4eme_trimestre
28 FROM
29   retour_client AS r
30 JOIN
31   ref_magasin_2 AS m ON r.ref_magasin = m.ref_magasin
32 WHERE
33   r.date_achat BETWEEN '2021-01-01' AND '2021-12-31'
34 GROUP BY
35   m.ref_magasin, m.libelle_de_commune
36

```

Result Grid						
		Filter Rows:	Export:	Wrap Cell Content:		
	ref_magasin	libelle_de_commune	NPS_1er_trimestre	NPS_2eme_trimestre	NPS_3eme_trimestre	NPS_4eme_trimestre
47	Freneuse	6	11	0	9	
6	Osny	2	4	8	6	
69	Gonesse	18	0	13	11	
54	Paris 2e Arrondissement	8	14	8	-5	
71	Montreuil	8	19	8	0	
3	Paris 5e Arrondissement	18	4	0	11	
18	Saint-Germain-lès-Arpajon	7	0	7	7	
76	Carrières-sur-Seine	3	3	6	10	

## Popularité du magasin

SELECT

```
m.ref_magasin,  
m.libelle_de_commune,  
COUNT(*) AS nombre_achats,  
m.population,  
COUNT(*) / m.population AS popularite
```

FROM

```
retour_client AS r
```

JOIN

```
ref_magasin_2 AS m ON r.ref_magasin = m.ref_magasin
```

GROUP BY

```
m.ref_magasin, m.libelle_de_commune, m.population
```

```
1 • SELECT  
2     m.ref_magasin,  
3     m.libelle_de_commune,  
4     COUNT(*) AS nombre_achats,  
5     m.population,  
6     COUNT(*) / m.population AS popularite  
7 FROM  
8     retour_client AS r  
9 JOIN  
10    ref_magasin_2 AS m ON r.ref_magasin = m.ref_magasin  
11 GROUP BY  
12    m.ref_magasin, m.libelle_de_commune, m.population  
13
```

Result Grid					
Filter Rows:		Export:		Wrap Cell Content:	
	ref_magasin	libelle_de_commune	nombre_achats	population	popularite
▶	47	Freneuse	35	3930	0.0089
	6	Osny	49	16236	0.0030
	69	Gonesse	38	26356	0.0014
	54	Paris 2e Arrondissement	37	23009	0.0016
	71	Montreuil	37	102770	0.0004
	3	Paris 5e Arrondissement	28	60938	0.0005
	18	Saint-Germain-lès-Arpajon	41	9338	0.0044
	76	Carrières-sur-Seine	31	15473	0.0020

## Meilleures catégories de produit sur l'année

SELECT

p.titre\_produit,

AVG(r.note) AS moyenne\_note,

(SELECT AVG(note) FROM retour\_client) AS moyenne\_globale

FROM

retour\_client AS r

JOIN

produit AS p ON r.cle\_produit = p.cle\_produit

WHERE

r.date\_achat BETWEEN '2021-01-01' AND '2021-12-31'

GROUP BY

p.titre\_produit

ORDER BY

moyenne\_note DESC

LIMIT 5

```
1 • SELECT
2     p.titre_produit,
3     AVG(r.note) AS moyenne_note,
4     (SELECT AVG(note) FROM retour_client) AS moyenne_globale
5 FROM
6     retour_client AS r
7 JOIN
8     produit AS p ON r.cle_produit = p.cle_produit
9 WHERE
10    r.date_achat BETWEEN '2021-01-01' AND '2021-12-31'
11 GROUP BY
12    p.titre_produit
13 ORDER BY
14    moyenne_note DESC
15 LIMIT 5
```

Result Grid			
Filter Rows:		Export:	Wrap Cell Content:
	titre_produit	moyenne_note	moyenne_globale
▶	Plantes aromatiques surgelées	9.2000	8.0527
	Sodas	9.0000	8.0527
	Boissons alcoolisées	8.7895	8.0527
	Sauces au soja	8.7727	8.0527
	TV	8.7143	8.0527