Rechercher les lignes :

* Utilisateur ou nombre achat supérieur uniquement à 10.
* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `utilisateur` WHERE `nombre\_achat` > 10
* Utilisateur nom finissant par « ne » et date naissance supérieure à 2000.
* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `utilisateur` WHERE `nom` [like](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-comparison-functions.html#operator_like) '%ne' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) `date\_naissance` > '2000-01-01'
* Utilisateur nombre achat = 50 ou 60 ou 80 et en plus email finissant par hotmail.fr
* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `utilisateur` WHERE `nombre\_achat` [in](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in)(50,60,80) [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) `email` [like](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-comparison-functions.html#operator_like) '%@hotmail.fr'
* Utilisateur ayant réalisé des achats et qu’il soit née après 2000 ou qu’il vive à lille trié par id plus petit au plus grand.
* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `utilisateur` WHERE `nombre\_achat` > 0 [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) (`date\_naissance` > '2000-01-01' [or](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_or) `ville` = 'Lille') order by id asc
* Utilisateur vivant en France et à lille ou simplement ayant réalisé des achats.
* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `utilisateur` WHERE (`pays` = 'France' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) `ville` = 'Lille') [OR](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_or) `nombre\_achat` > 0

Regrouper par villes le nombre d’utilisateur et ayant un nom different de « paris »

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [count](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_count)(\*), ville FROM `utilisateur` where nom != 'Paris' GROUP BY ville

HAVING s’utilise uniquement avec les fonctions d’aggrégations (count, sum, avg,….)

MEMO, elle s’applique sur le tableau déjà regroupé

WHERE s’utilise dans le cas contraire

MEMO, elle s’applique sur le tableau initial (non regroupé)

SELECT count(\*), ville, sum(nombre\_achat) FROM `utilisateur` GROUP BY ville having sum(nombre\_achat)>45

SELECT sum(nombre\_achat), ville from utilisateur where nombre\_achat>40 group by ville having sum(nombre\_achat)>50

WHERE EN PREMIER CAR CONDITION SUR TABLEAU INITIAL

GROUP BY REGROUPEMENT DES LIGNES PAR RAPPORT A UNE COLONNE

HAVING FILTRAGE SUR TABLEAU REGROUPE

Il est possible de coupler :

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) round([avg](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html" \l "function_avg" \t "mysql_doc)(nombre\_achat),1) FROM `utilisateur`

Sélectionner le nom et le prénom de l’utilisateur ayant acheté le moins d’objet

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nom, prenom FROM utilisateur where `nombre\_achat` = ([select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [MIN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_min)(nombre\_achat) from utilisateur)

Sélectionner le nom et le prénom de l’utilisateur ayant acheté le moins d’objet tout en étant née entre 1990 et 2005

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nom, prenom FROM utilisateur where `nombre\_achat` = ([select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [MIN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_min)(nombre\_achat) from utilisateur where [year](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/date-and-time-types.html)(date\_naissance) BETWEEN 1990 [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) 2005 )

Sélectionner le nom en minuscule et prénom en majuscule des personnes née les mois de Janvier, février, ou avril et ayant réalisé au moins une commande

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) lower(nom), upper(prenom), date\_naissance FROM utilisateur where month(date\_naissance) [IN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in) (1,2,4) [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) nombre\_achat > 0

Requete jointure avec alias

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) u.nom as nom\_user FROM utilisateur as u INNER JOIN adresse\_livraison as al on u.id = al.id\_user

Requete jointure avec alias et fonction agrégation

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [count](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_count)(\*) as nb\_adresse\_livraison FROM utilisateur as u INNER JOIN adresse\_livraison as al on u.id = al.id\_user

Equivalent de full outer join sur mysql serveur phpmyadmin

SELECT \* FROM `utilisateur` as u

left join adresse\_livraison as al on u.id = al.id\_user

UNION

SELECT \* FROM `utilisateur` as u

right join adresse\_livraison as al on u.id = al.id\_user

Selection utilisateur avec badges en fonction de certaines colonnes

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) u.nom, u.prenom, b.nom as nom\_du\_badge, ub.date as date\_obtention\_badge from utilisateur as u inner join utilisateur\_badge as ub ON u.id = ub.id\_user inner join badge as b ON b.id = ub.id\_badge

Selection avec jointure + condition + order by

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) u.nom, u.prenom, b.nom as nom\_du\_badge, ub.date as date\_obtention\_badge from utilisateur as u inner join utilisateur\_badge as ub ON u.id = ub.id\_user inner join badge as b ON b.id = ub.id\_badge WHERE b.nom = "Premium" order by nom desc

Récupérer que les lignes utilisateurs sans liaison avec adresse de livraison

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* from utilisateur as u [left](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-functions.html#function_left) join adresse\_livraison as al ON al.id\_user = u.id where al.id\_user [is](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#operator_is) null

Me fournir pour chaque utilisateur le nombre d'adresse de livraison

select count(\*) as nb\_adresse, u.nom, u.prenom from utilisateur as u inner join adresse\_livraison as al on u.id = al.id\_user group by u.id;

Me fournir l'utilisateur ayant le plus de badge (nom, prenom mail nb badge)

select count(\*) as nb\_badge, u.prenom, u.nom from utilisateur as u  
INNER join utilisateur\_badge as ub  
on u.id = ub.id\_user  
group by ub.id\_user  
order by nb\_badge DESC  
limit 1

Me donner la moyenne du nombre de badge sur l'ensemble des utilisateurs

SELECT AVG(nbbadge) as moyenne FROM (  
SELECT COUNT(\*) nbbadge FROM `utilisateurs\_badge`  
GROUP BY id\_user) as tmp;