Important Concepts

Where vs Having

Where :

Where filter rows of a table

where works on rows data not on aggregated data

Important : Where = is a case sensitive clause

Example: Select * from gbpnihe_schema.employeeinfo where department='HR'

So we can cast the column data as select * from gbpnihe_schema.employeeinfo where lower(deptname)='hr'

to avoid problem of different case Having

It works on aggregated data

It should be used with GROUP By clause

It select rows after grouping

Union vs Union All

Both union and union all are applied bewteen two select query

Example:

select * from gbpnihe_schema.employeeinfo union all select * from gbpnihe_schema.employeedetails

Both the table should have same number of column and same data type column

Order of the respective column must also be same It is not neccessary that name of column should be same .

If name of the columns is different than in output result the data will be fetched with name of column as in Union ignore the duplicate rows and include duplicate rows only once

but Union all includes all the the rows

In vs Exists

Basic syntax: select * from gbpnihe_schema.employeeinfo where deptname in ('HR','ADMIN')

above query can also retrun as

select * from gbpnihe_schema.employeeinfo where (deptname='HR' OR deptname='ADMIN')

So IN clause is basically used to shorten the query with mui;ptple OR condition

We can also write query with IN clause as:

select * from gbpnihe_schema.employeeinfo where deptname in (select deptname from departmentlist)

Exists

Exists clause return true or false value

i.e it tells whether inner query /sub query returns any value or not

Exists retrun true if inner query return atleast one record if it returns no record it retruns zero Example: select * from gbpnihe_schema.employeeinfo where EXISTS (select * from gbpnihe_schema.countries where country_name='ad')

IN Clause	EXISTS Clause
In the IN-condition SQL Engine compares all the values in the IN Clause.	Once true is evaluated in the EXISTS condition then the SQL Engine will stop the process of further matching.
To check against only a single column, IN operator can be used.	For checking against more than one single column, you can use the EXISTS Operator.
IN works faster than the EXISTS Operator when If the sub-query result is small.	If the sub-query result is larger, then EXISTS works faster than the IN Operator.

ORDER BY vs GROUP BY

Order by is used to sort by a column in ascending or descending order
by default data is sorted in descending order
Example:
select * from gbpnihe_schema.employeeinfo order by name // will sort data by name in ascending order
select * from gbpnihe_schema.employeeinfo order by name asc // will sort data by name in ascending order
select * from gbpnihe_schema.employeeinfo order by name desc // will sort data by name in descending order

Group By
Group by is basically used with aggregate function and it group the aggregated data as per the given column Example:
select gender,count(gender) from gbpnihe_schema.employeeinfo group by(gender)
Important
Group by clause can be written after where clause in select statments, where clause cannot be used after group by clause
Example: select deptname,count(deptname) from gbpnihe_schema.employeeinfo where deptname='HR' group by (deptname)//No error
select deptname,count(deptname) from gbpnihe_schema.employeeinfo group by (deptname) where deptname='HR' //Errors

Having is used after group by
We can use order by with group by but order by should come after group by clause

Alter Command

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.
The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

Add Column:

Alter table employee add address text;// explicity place column before columnname or not both will work

Delete Column:

Alter table employee drop address;// explicity place column before columnname or not both will work

Change the data type -

In Mysql-MODIFY is used to change the data type of a column
Alter table employee modify COLUMN address integer// In mysql
In Pgsql - to change the data type alter command is used with alter
alter table gbpnihe_schema.employeeinfo alter COLUMN age type integer // explicity place column before columnname or not both will work

Rename the column name:

alter table gbpnihe_schema.employeeinfo rename column gender to gender1 // In mysql and pgsql both //// explicity place column before columnname or not both will work

SHow CONSTRAINSTS from employee ;

Above query can be used to display all the constraints of the given table;

Adding the Constraints

ALTER TABLE table_name MODIFY column_name datatype NOT NULL;
ALTER TABLE Persons ADD CONSTRAINT PK_Person PRIMARY KEY (ID,LastName);
ALTER TABLE table_name ADD CONSTRAINT MyUniqueConstraint CHECK (CONDITION);
ALTER TABLE table_name ADD CONSTRAINT MyUniqueConstraint UNIQUE(column1, column2...);

Droping the Constraints

ALTER TABLE table_name DROP CONSTRAINT MyUniqueConstraint; // In pgsql
ALTER TABLE table_name DROP INDEX MyUniqueConstraint; //In mysql

Droping the Primary key Constraints

ALTER TABLE table_name DROP CONSTRAINT MyPrimaryKey; //in Pgsql
ALTER TABLE table_name DROP PRIMARY KEY; //in mysql

Multiple Order by and Group by clause in query

Multiple Order by
Example 1:
SELECT * FROM Customers order by country,city,address
So data will be first sorted by country and within same country data will be sorted by city and then within same city data will be sorted by address

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
64	Rancho grande	Sergio Gutiérrez	Av. del Libertador 900	Buenos Aires	1010	Argentina
12	Cactus Comidas para llevar	Patricio Simpson	Cerrito 333	Buenos Aires	1010	Argentina
54	Océano Atlántico Ltda.	Yvonne Moncada	Ing. Gustavo Moncada 8585 Piso 20-A	Buenos Aires	1010	Argentina
20	Ernst Handel	Roland Mendel	Kirchgasse 6	Graz	8010	Austria
59	Piccolo und mehr	Georg Pipps	Geislweg 14	Salzburg	5020	Austria
50	Maison Dewey	Catherine Dewey	Rue Joseph-Bens 532	Bruxelles	B-1180	Belgium
76	Suprêmes délices	Pascale Cartrain	Boulevard Tirou, 255	Charleroi	B-6000	Belgium
31	Gourmet Lanchonetes	André Fonseca	Av. Brasil, 442	Campinas	04876-786	Brazil
88	Wellington Importadora	Paula Parente	Rua do Mercado, 12	Resende	08737-363	Brazil
67	Ricardo Adocicados	Janete Limeira	Av. Copacabana, 267	Rio de Janeiro	02389-890	Brazil
61	Que Delícia	Bernardo Batista	Rua da Panificadora, 12	Rio de Janeiro	02389-673	Brazil
34	Hanari Carnes	Mario Pontes	Rua do Paço, 67	Rio de Janeiro	05454-876	Brazil
62	Queen Cozinha	Lúcia Carvalho	Alameda dos Canàrios, 891	São Paulo	05487-020	Brazil
15	Comércio Mineiro	Pedro Afonso	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil

81	Tradição Hipermercados	Anabela Domingues	Av. Inês de Castro, 414	São Paulo	05634-030	Brazil
21	Familia Arquibaldo	Aria Cruz	Rua Orós, 92	São Paulo	05442-030	Brazil
51	Mère Paillarde	Jean Fresnière	43 rue St. Laurent	Montréal	H1J 1C3	Canada
10	Bottom-Dollar Marketse	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada
42	Laughing Bacchus Wine Cellars	Yoshi Tannamuri	1900 Oak St.	Vancouver	V3F 2K1	Canada
83	Vaffeljernet	Palle Ibsen	Smagsløget 45	Århus	8200	Denmark
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany

Muliple Group By
SELECT country,count(*) FROM Customers group by country;
Output will be different country names with count of each country i.e no. of entries of each country
Example: India ---10
Pakistan -9
Srilanka ---8
SELECT city,country,count(*) FROM Customers group by country; Output will same as above but city will be displayed it s agroup by function so first city will be displayed other cities within the same country will be ignored Example:
Delhi---India---10
Karachi---Pakistan---9
Ugana---Srilanka---8
SELECT city,country,count(*) FROM Customers group by country;

Constraints in SQL

NOT NULL - Ensures that a column cannot have a NULL value
UNIQUE - Ensures that all values in a column are different
PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
FOREIGN KEY - Prevents actions that would destroy links between tables
CHECK - Ensures that the values in a column satisfies a specific condition
DEFAULT - Sets a default value for a column if no value is specified
CREATE INDEX - Used to create and retrieve data from the database very quickly

LIKE vs ILIKE

Like is basically a case sensitive clause so if we want to ignore case we can use ilike

```
select deptname,count(*) from gbpnihe_schema.employeeinfo group by deptname having deptname ilike 'a%'
select deptname,count(*) from gbpnihe_schema.employeeinfo group by deptname having deptname like 'A%'
```

Ordering of output data using Select statement without order by clause

Output data is displayed using FIFO technique i.e row inserted first will appear first
But suppose we have updated data of a row so that row will displayed last in output result because to update that row first that row inserted with new updated data and old row gets deleted from the table it means in FIFO order updated row will be at last