Union and Union All:

Used to combine the results of two or more 'Select' statements

Union	Union All
* Combines the result of two or more 'Select' statements and removes duplicate rows from the result set.	* Combines the results of two or more 'Select' statements and includes all rows, including duplicates
* Performs sort operation to eliminate Duplicates.	* Doesn't perform sort operation to remove duplicates.

Example for union:

```
(Select *
from employee
order by empid asc
limit 5)
union
(Select *
from employee
order by empid desc
limit 5);
```

Example for union all:

```
(Select *
from employee
order by empid asc
limit 5)
union all
(Select *
from employee
order by empid desc
limit 5);
```

Limit: To restrict the result we can use the LIMIT in SQL. The LIMIT Clause in the SQL is used to restrict the number of records.

Offset: Offset is used to specify from which row we want the data to retrieve. Offset is used along with the LIMIT.

Syntax for OFFSET and LIMIT:

```
Select Column_Name
From Table_Name
Order by Column_Name
Limit Number_of_rows Offset offset_values;
```

Example:

To select the first 3 records

```
Select Empno
from employee
order by Empno
Limit 3;
```

To skip the first 3 records and select next 2 records

Select Empno

from employee

order by Empno

Limit 3 Offset 3;

Intersect:

Used to return common records (intersection) of two 'Select' statements. This means it will return only the rows that are present in both result sets.

Syntax:

Select Column_Name

From Table_Name

Intersect

Select Column_Name

From Table_Name

Example:

Select Empno,Empname

from employee

Intersect

Select Empno,Empname

From employee;

Except: Used to return the rows from the first 'Select' statement that are not present in the second 'Select' statement.

Syntax:

Select column_name

from Table_name

Except

Select Column_Name

From Table_Name;

Example:

Select Empno, Empname

from employee

Except

Select Empno, Empname

from employee;

*It returns the data's of first table which are not present in second table.

Minus: Minus compares two SELECT statements and returns rows from the first SELECT statement that are absent in the second.

Example:

Select Empno, Empname

from employee

Minus

Select Empno, Empname

from employee;

Indexes: Index is a database object that improves the speed of data retrieval operations on a table.

Purposes of Indexes:

1.Speed of retrieval

2.Enhance query performance

3.Enforce uniqueness

4.Facilitate sorting and ordering

5.Improve performance of aggregate functions

6.Optimize joins

If your assigning the column as index fetching the data will be fast compare to without index.

What is Sql injection?

Entering some code into the webserver so that it can be manipulated

Attackers gain access to the database and steal the valuable information from database and destroy or alter data in database

SQL injection occurs when malicious sql statement are inserted into fields

for example with sql injection a hacker can change the price of the product or can gain the customer information

What is Stored procedures?

A stored procedure is a prepared sql code that you can save, so that the code can be reused over and over again

So if you have an sql query that you write over and over again, save it as a stored procedure and the just call and execute it.

Aggregate Functions:

SUM(): Calculates the sum of values within the window.

AVG(): Calculates the average of values within the window.

MIN(): Returns the minimum value within the window.

MAX(): Returns the maximum value within the window.

COUNT() : Return the number of rows

Date Functions:

Select Curdate(); → Returns Current / today's date

Select current_date(); → Returns Current date

Select current_time(); → Returns Current Time

Select current_user(); → Returns Current user

Select current_timestamp(); → returns the current date and time.

SELECT DATEDIFF("2017-06-25", "2017-06-15"); → Difference between number of days

SELECT DATEDIFF("2017-06-25 09:34:21", "2017-06-15 15:25:35"); → Difference between number of days