SQL For Testers

By - Amol Reddy

SQL Introduction

SQL- **S**tructured   **Q**uery   **L**anguage

➢ SQL is a standard language for storing, manipulating and retrieving data in databases

➢ SQL keywords are **NOT case sensitive**   Ex. SELECT as select

➢ Semicolon is the standard way to separate each SQL statement in database systems   ex. Select * from employee;

➢ SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987

What SQL do?

➢ SQL can execute quires, retrieve data , insert records, update records, delete records, create new tables in a database, create database, create stored procedures (SP) in a database, create views in a database.
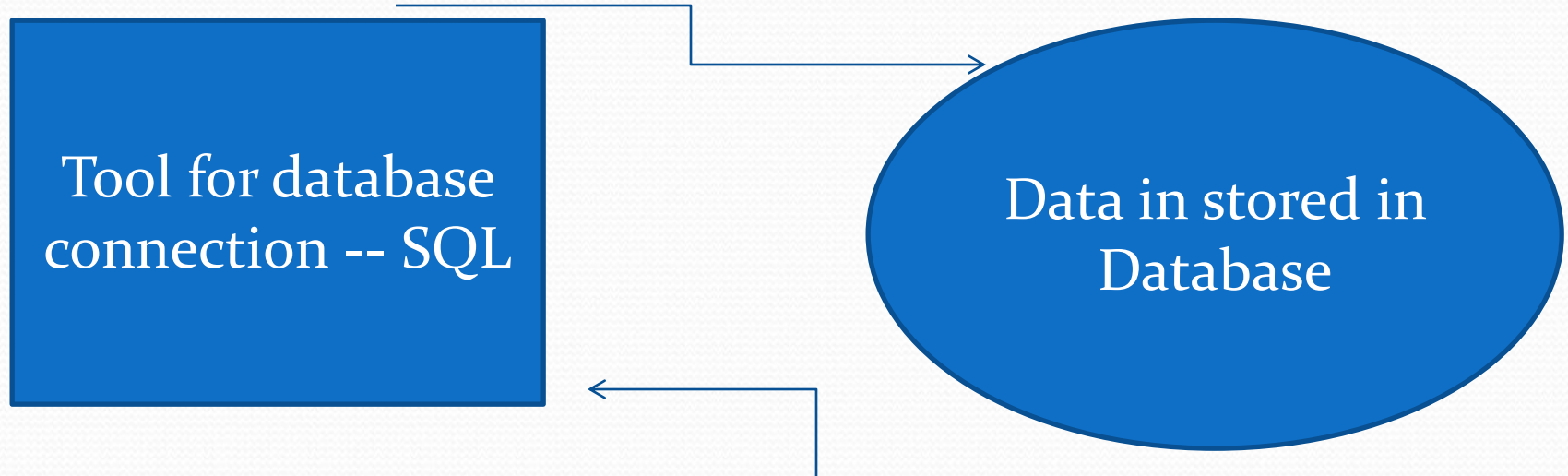
 RDBMS-

➢RDBMS stands for Relational Database Management System.

➢ RDBMS is the basis for SQL, and for all modern database systems such as MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.
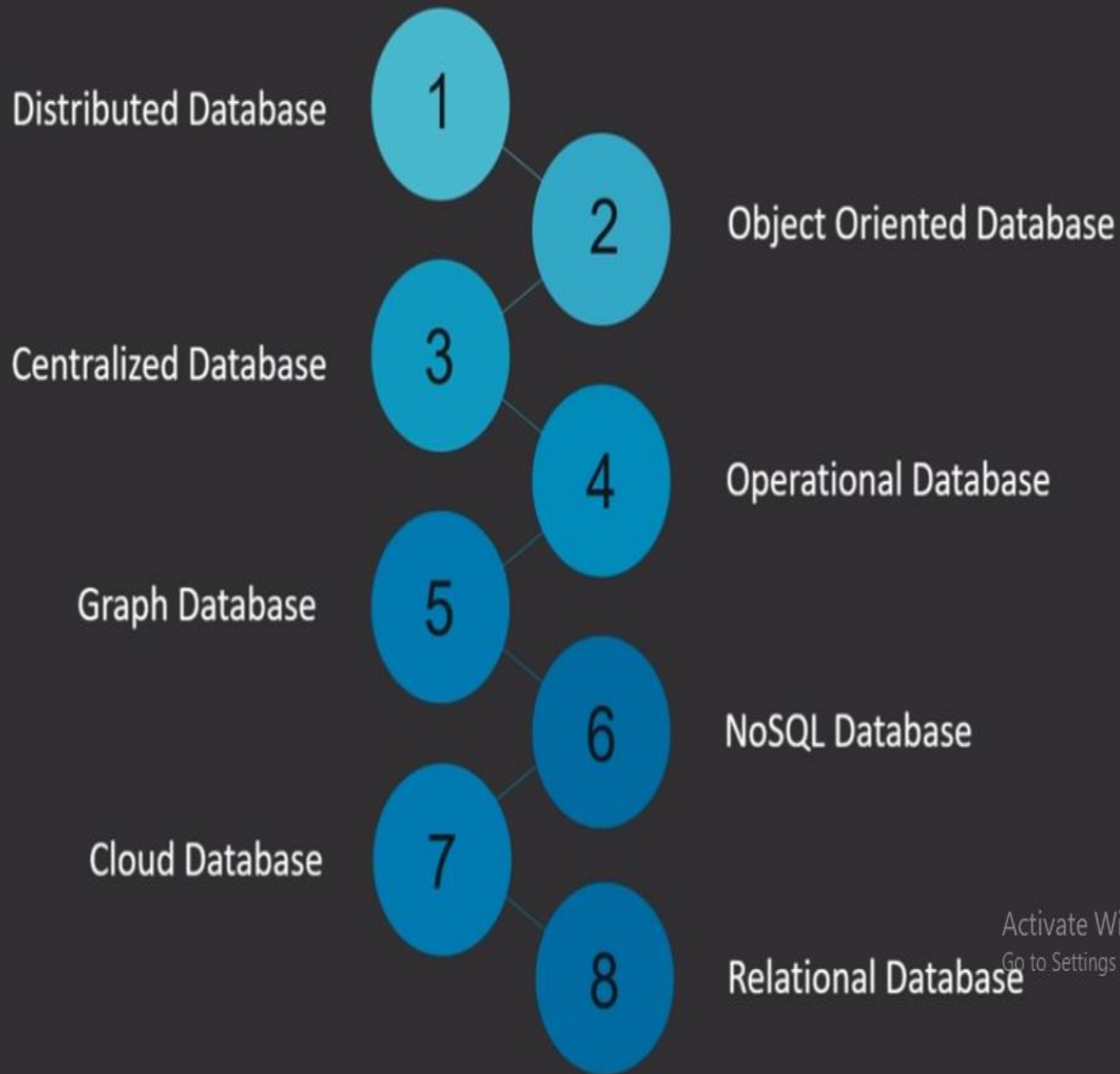
Creating DB, Creating table, updating data in table, Delete

Tool for database connection -- SQL

Data in stored in Database

Retrieve data from db

TYPES OF DATABASE

Distributed Database — **1**

2 — Object Oriented Database

Centralized Database — **3**

4 — Operational Database

Graph Database — **5**

6 — NoSQL Database

Cloud Database — **7**

8 — Relational Database

Few Popular Databases Management Studio

Mongo DB

Postgre SQL

Microsoft Access

SQL Server

MySQL

ORACLE DB

Activate Windows

SQL Command Types

➤ **DDL(Data Definition Language)**: - Allows to work with the Structure or Definition of the database or tables

• SQL commands are as follows in DDL: CREATE, ALTER, DROP, TRUNCATE

➤ **DML(Data Manipulation Language): -** To deal with the data itself directly

• SQL Commands come under DML are as follows: INSERT, UPDATE, DELETE, SELECT

➤ **DQL(Data Query Language)**: - Deals with the data but to retrieve the data

• SQL Commands come under DQL are as follows: SELECT

Tables in SQL

Product

Sr no./ PName	Price	Category	Manufacturer
Gizmo	$50	Gadgets	GizmoWorks
Powergizmo	$40	Gadgets	GizmoWorks
SingleTouch	$10	Photography	Canon
MultiTouch	$200	Household	Hitachi

Table name

Attribute names/ header

Tuples

Row

Data Types in SQL

- Characters / String:
  - CHAR(20)   -- fixed length
  - VARCHAR(255)   -- variable length
- Numbers:
  - INT
  - REAL, FLOAT   (10%)   -- differ in precision
  - Decimal (4.9403)
- Times and dates:
  - DATE
  - DATETIME   -- SQL Server
- Binary data type
  - Binary

SQL operators

Type	Operator	Description	Example
Arithmetic Operators	+ (Addition)	Adds values on either side of the operator.	10+ 20 will give 30
	- (Subtraction)	Subtracts right hand operand from left hand operand.	20- 10 will give 10
	* (Multiplication)	Multiplies values on either side of the operator.	5* 5 will give 25
	/ (Division)	Divides left hand operand by right hand operand.	20 / 5 will give 4
	% (Modulus)	Divides left hand operand by right hand operand and returns remainder.	22 % 5 will give 2

SQL operators

Type	Operator	Description	Example
Comparison Operators	=, !=	Checks, values of two operands are equal or not	(3 = 6) is not true.
	<>	Checks, values of two operands are equal or not	(10 <> 4) is true.
	>, <	Checks, value of left operand with greater/lesser value with right operand	(10 > 35) is not true.
	>=, <=	Checks, value of left operand with greater/lesser than or equal to the value of right operand	(a >= b) is not true.
	!<	Checks if the value of left operand is not less than the value of right operand	(a !< b) is false.
	!>	Checks if the value of left operand is not greater than the value of right operand	(a !> b) is true.

SQL operators

Type	Operator	Description	Example
Logical Operators	**AND**	The AND operator allows the existence of multiple conditions in an SQL statement's WHERE clause.	Cond AND cond
	OR	The OR operator is used to combine multiple conditions in an SQL statement's WHERE clause.	Cond OR cond
	NOT	The NOT operator reverses the meaning of the logical operator with which it is used. Eg: NOT EXISTS, NOT BETWEEN, NOT IN, etc. **This is a negate operator.**	NOT Cond

SQL **DDL** command-

❑ **Create – Table & db**

      Syntax- CREATE DATABASE *databasename*;
            Ex. CREATE DATABASE velocity;

      Syntax-

            CREATE TABLE *table_name* (
              column1 datatype,
              column2 datatype,
              column3 datatype, ...);

            Ex. CREATE TABLE Persons (
              PersonID int,
              LastName varchar(255),
              FirstName varchar(255),
              Address varchar(255),
              City varchar(255));

SQL **DDL** command-

❑ **Drop–** The DROP DATABASE, DROP TABLE statement is used to **drop an existing SQL database, existing table** in a database **with structure of table** of database.

       Syntax- DROP DATABASE *databasename*;
             Ex. DROP DATABASE testDB;

       Syntax- DROP TABLE *table_name*;
             Ex. DROP TABLE VelocityVV;

❑ **Alter** - The ALTER TABLE statement is used to add, drop, or alter **columns in an existing table.**

       Syntax- ALTER TABLE *table_name*
           ADD *column_name datatype*;
             Ex. ALTER TABLE VelocitVV
                alter Email varchar(20);

SQL **DDL** command-

❑ **TRUNCATE–** The SQL TRUNCATE TABLE command is **used to delete complete data** from an existing table **but structure of table still remaining**
❑ You can also use DROP TABLE command to delete complete table but it would remove complete table structure form the database

      Syntax- TRUNCATE TABLE table_name;
              Ex. TRUNCATE TABLE CUSTOMERS;

Tables in SQL

VCTC

Fname	Lname	Gender	Mockresult	Location
Aditya	Patki	M	7	Nagpur
Atish	Jain	M	8	Nagpur
Deepka	Roy	F	7	Pune
Jason	Borges	M	7	Bider
Neha	Bedre	F	8	Pune
Neha	Joshi	F	9	Pune
Pooja	Patil	F	8	Abad
Shri	Patil			Goa

SQL **DML** command-

❑ **INSERT-** INSERT statement is used to insert a single record or multiple records into a table in SQL Server.

       Syntax- INSERT INTO TN. (CN1, CN2, ...)
             VALUES (value1, value2, ...);

       Syntax- INSERT INTO *T.N.*
             VALUES (*value1*, *value2*, *value3*, *...*);

       Ex.   INSERT INTO Velocity
             VALUES(101,'Atul', 'Patil', 9921374751, 16, 'Pune');

  INSERT INTO VCTC VALUES('Atish','Jain','M',7,'Ngapur');

       Ex.   INSERT INTO VCTC (Fname, Lname, Gender)
             VALUES('Amol','Reddy','M');

SQL **DML** command-

❑ **SELECT–** Select statement is used to fetch the data from a database table.

    Syntax- SELECT * FROM TN.;

            SELECT CNS. FROM   TN.;

TN.– indicates table_name
CNS– indicates Coloumn_names

        Ex. SELECT * FROM VCTC;

        SELECT Fname, Lname FROM VCTC;

SQL **DML** command-

❑ **DISTINCT** clause/keyword is used in conjunction with the SELECT statement to eliminate all the duplicate records and fetching only unique records.

       Syntax- SELECT DISTINCT C.N.S. FROM   T.N
                     SELECT DISTINCT * FROM   T.N

       Ex. SELECT DISTINCT Fname FROM VCTC

❑ **TOP** clause is used to fetch a TOP N number or X percent records from a table.

       Syntax- SELECT TOP no./ percent C.N.S FROM T.N.

       Ex. SELECT TOP 10 * FROM VCTC;

SQL **DML** command-

❑ **Aggregate Function-** Aggregate used with select statements, they will return some numeric values

❑ **COUNT**() function returns the number of rows that matches.
❑ **AVG**() function returns the average value of a numeric column.
❑ **SUM**() function returns the total sum of a numeric column.
❑ **MAX**() function returns the max number of rows that matches a specified criterion.
❑ **MIN**() function returns the min value of a numeric column.

Syntax- SELECT COUNT (C.N.) FROM   *T.N.*
SELECT AVG (C.N.) FROM   *T.N.*
SELECT SUM (C.N.) FROM   *T.N.*
SELECT MAX (C.N.) FROM *T.N.*
SELECT MIN (C.N.) FROM   *T.N.*

Ex. SELECT COUNT (Mockresult) FROM VCTC;
SELECT AVG (Mockresult) FROM VCTC;
SELECT SUM (Mockresult) FROM VCTC;

SQL **DML** command-

❑ **WHERE** clause is used to specify a condition while fetching the data from a single table

> **Syntax**- SELECT * FROM T.N.
>   WHERE   Condition

❑ Condition we can write as ---- (C.N. operators value)

> **Ex.**   SELECT * FROM VCTC
>   WHERE   FName = 'Neha';

SQL **DML** command-

❑ **AND, OR, NOT-** operators are used to combine multiple conditions to narrow data in an SQL statement.

> **Syntax-** SELECT * FROM   T.N.
> WHERE Condition AND   Condition
> **Syntax-** SELECT * FROM   T.N.
> WHERE Condition   OR   Condition
> **Syntax-** SELECT * FROM   T.N.
> WHERE NOT   Condition

Ex. SELECT * FROM   VCTC
Where Fname = 'Aditya'   AND   Lname = 'Patki' ;
Ex. SELECT * FROM   VCTC
Where Fname = 'Aditya'   OR   Lname = 'Shabe' ;
Ex. SELECT * FROM   VCTC
Where   NOT   Fname = 'Aditya' ;

SQL **DML** command-

❑ **LIKE** clause is used to find a value to similar values or pattern using **wildcard operators.** There are two wildcards used in conjunction with the LIKE operator.

❑ **Wildcard operates** – 1. **percent sign (%)** ➔ Matches one or more characters.
2. **underscore (_)** ➔ Matches one character.
3. **[charlist]%** ➔ Matches more characters in charlist
4. **[! charlist]%** ➔ Not matches characters sequence

Syntax- SELECT C.N.S FROM T.N.
WHERE C.N. LIKE 'Pattern'

Ex. SELECT * FROM VCTC   A% = AXXXXXX,
where Fname LIKE 'A%';   %A = XXXXXXA
Ex. SELECT * FROM VCTC   %A% = XXXAXXX
where Fname LIKE 'Adi_y';   AM_N = AMXN
Ex. SELECT * FROM VCTC   _ATIL = XATIL
where Fname LIKE '[AN]% ';   [AN]% = AXXXX , NXXXX

SQL **DML** command-

❑ **BETWEEN** keyword / operator used to selects values within a given range with a WHERE clause

      Syntax- SELECT C.N.S FROM T.N.
            WHERE C.N. BETWEEN Value1 AND Value2

      Ex. SELECT * FROM VCTC
      WHERE Mockresult BETWEEN 6 AND 9;   RESULT- 6,7,8,9

❑ **IN** keyword / operator allows you to specify multiple values in a WHERE clause

      Syntax- SELECT C.N.S FROM T.N.
            WHERE C.N. IN (Value1, Value2, Value3)

      Ex. SELECT * FROM VCTC
      WHERE Mockresult IN (6 , 9);   RESULT- 6, 9

SQL **DML** command-

❑ **IS NULL & IS NOT NULL -** keyword / operator used with WHERE clause.
❑ To test NULL values, We will have to use the IS NULL and IS NOT NULL operators

        Syntax- SELECT C.N.S FROM T.N.
                 WHERE C.N.   IS NULL

        Ex. SELECT * FROM VCTC
        WHERE Mockresult IS NULL ;

    FNAME= AMAN, SARIKA, NULL, POOJA, AMAR

SQL **DML** command-

❑ **Order By-** ORDER BY clause is used to sort the records in your result set

      Syntax- SELECT C.N.S FROM T.N.
             ORDER BY C.N. ASC|DESC;

      Ex. SELECT * FROM VCTC
     ORDER BY *Fname* ASC;

❑ **Alise -** COLUMN ALIASES are used to make changes column headings in your result set easier to read & same as TABLE ALIASES are used to shorten your SQL to make it easier to read.

      Syntax- C.N. [AS] alias_name
             T. N. [AS] alias_name

     Ex. SELECT * FROM VCTC [AS] VV
     Ex. SELECT Fanme as FN FROM VCTC WHERE FN = "Pooja"

SQL **DML** command-

❑ **Update-** UPDATE statement is used to update existing records in a table in a SQL Server database.

        Syntax- UPDATE T.N.
                    SET C.N.1 = Value1,   C.N. 2= Value2,   ...
                    WHERE *condition*;

        Ex.   UPDATE VCTC
                    SET Fname = 'Pooja', Lname= ' Patil'
                    WHERE Fname = ' Neha'

❑ **Delete-** DELETE statement is used to delete a single record or multiple records from a table in SQL Server.

        Syntax- DELETE   FROM T.N. WHERE *condition*;

        Ex.   DELETE   FROM VCTC WHERE   Fname = 'Amol'

SQL **DML** command-

❑ **Union -** UNION operator is used to **combine the result sets of 2 or more SELECT** statements. It **removes duplicate rows** between the various SELECT statements.

❑ Each SELECT statement within the UNION operator must have the same number of columns in the result sets with similar data types.

> **Syntax**- SELECT C.N.S FROM T.N 1.
>   UNION
>   SELECT C.N.S FROM T.N 2;

> **Ex.**   SELECT * FROM VCTC.
>   UNION
>   SELECT * FROM VKVP;

❑ **Union All -** UNION ALL operator is used to combine the result sets of 2 or more SELECT statements. It **returns all rows** from the query and it does not remove duplicate rows between the various SELECT statements.

Each SELECT statement within the SQL Server UNION ALL operator must have the same number of fields in the result sets with similar data types.

SQL **DML** command-

❑ **SELECT INTO-** SELECT INTO statement is used to **create a table from an existing table** by copying the existing table's columns.

       Syntax- SELECT C.N.S
                  INTO new T. N.
                  FROM old T. N. ;

       Ex.   SELECT *
                  INTO VCTCViman
                  FROM VCTC;

SQL **DML** command-

Question –

1. What is DDL & DML in SQL
2. What is syntax of select statement
3. What is the Aggregate function in SQL
4. What are wildcard operates present in SQL & use of that
5. What is syntax of update & insert command in SQL

Simple SQL Query

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Select distinct (Category) from product
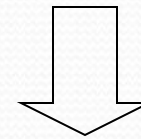
⇩
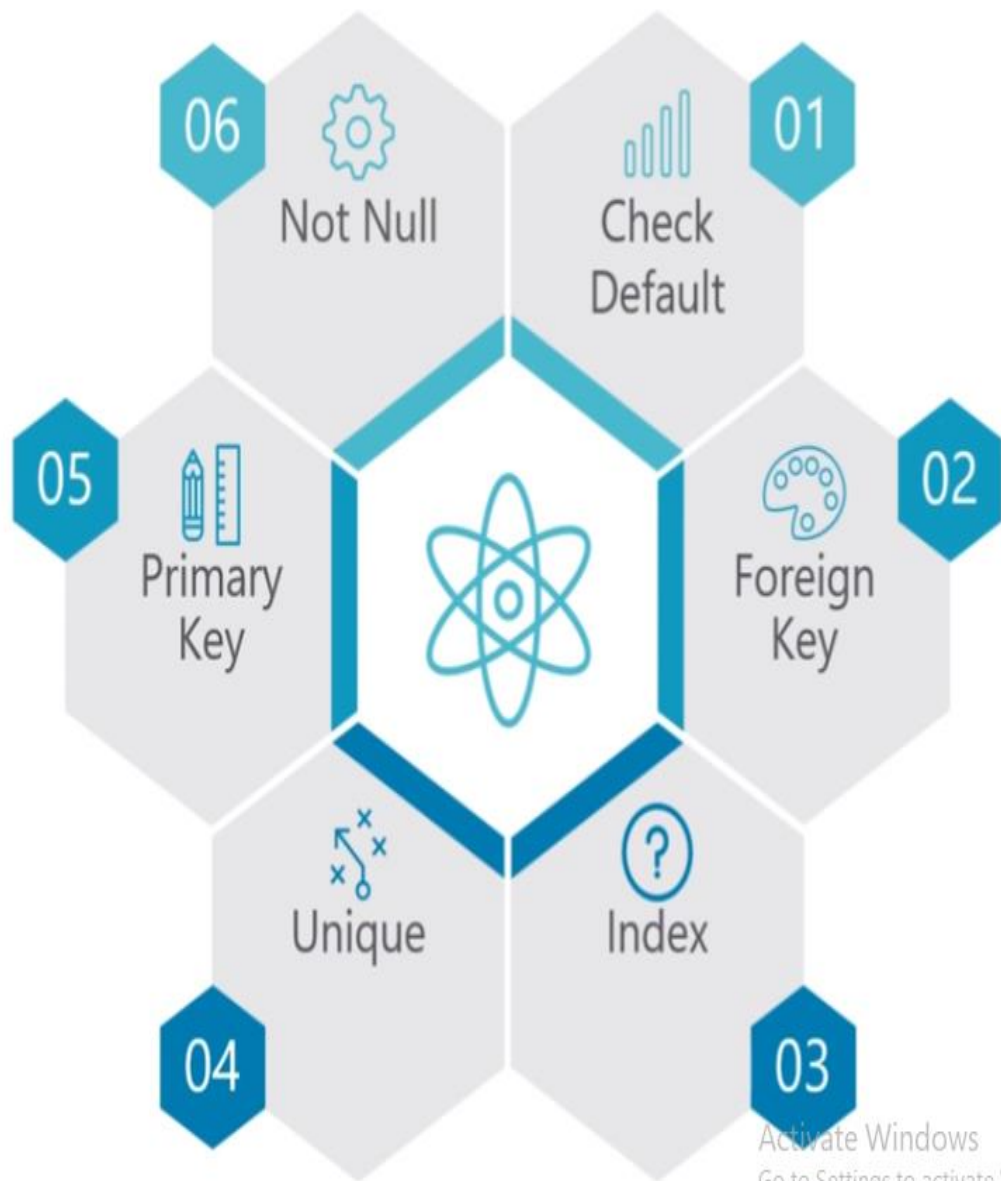
PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks

SQL Table constrains-

❑ SQL constraints are used to **specify rules for data in a table**. Constraints can be used when the table is created with the CREATE TABLE statement

❑ **NOT NULL -** The NOT NULL constraint forces a column to **NOT accept NULL values**. Becz while creating table if value not present it take NULL value in C.N.

❑ **UNIQUE -** The UNIQUE constraint ensures, **all values** in a column are **different.**

❑ **PRIMARY KEY -** The PRIMARY KEY constraint uniquely identifies each record in a table. Primary keys must contain **UNIQUE values**, and cannot contain **NULL** values.

❑ **FOREIGN KEY -** A FOREIGN KEY is a key used to **link two tables together**. A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table.

❑ **CHECK -** The CHECK constraint is used to **limit the value range** that can be placed in a column. If you define a CHECK constraint on a single column it allows only certain values for this column.

❑ **DEFAULT -** The DEFAULT constraint is used to provide a **default value** for a column.

34

Tables in SQL

VCTC

Fname	Lname	Gender	Mockresult	Location
Aditya	Patki	M	7	Nagpur
Atish	Jain	M	8	Nagpur
Deepka	Roy	F	7	Pune
Jason	Borges	M	7	Bider
Neha	Bedre	F	8	Pune
Neha	Joshi	F	9	Pune
Pooja	Patil	F	8	Abad
Shri	Patil			Goa

SQL Table constrains-

❑ SQL constraints are used to **specify rules for data in a table**. Constraints can be used when the table is created with the CREATE TABLE statement

    Syntax- CREATE TABLE T.N. (
          C.N.1 *datatype constraint*,
          C.N 2 *datatype constraint*,
          C.N.3 *datatype constraint*, ...);

    Ex. CREATE TABLE VCTC (

StudID	int	Primary Key,
Fname	varchar(255)	Not Null,
Lname	varchar(255)	,
Mockresult	int	Check (Mockresult < 20) ,
Location	varchar(255)	Default 'Punevctc'
Mobileno	int	Unique);

Constrains Difference-

Primary Key	Foreign Key	Unique Key
The PRIMARY KEY constraint uniquely identifies each record in a table. Primary keys must contain UNIQUE values	A FOREIGN KEY is a key used to link two tables together. A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table.	The UNIQUE constraint ensures that all values in a column are different.
Primary key cannot have a NULL value.	Foreign key can accept multiple null value.	Unique Constraint may have a NULL value.
Each table can have only one primary key.	We can have more than one foreign key in a table.	Each table can have more than one Unique Constraint.
Primary key is clustered index	Foreign keys do not automatically create an index, clustered or non-clustered	Unique key is a unique non-clustered index

SQL Join-

❑ JOINS are used to **retrieve data from multiple tables**. A SQL Server JOIN is performed whenever two or more tables are joined in a SQL statement.

❑ There are 5 different types of SQL Server joins:
1. INNER JOIN   2. LEFT JOIN
3. RIGHT JOIN   4. FULL JOIN   5. SELF JOIN

Syntax -   SELECT TN1 . CN, TN1 . CN, TN2 . CN, TN2 . CN
                       FROM TN1
   INNER JOIN TN2
   ON TN1 . CN (Primary Key) = TN2 . CN (Foreign Key)

❑ **INNER JOIN** - It is the most common type of join. SQL Server INNER JOINS return all rows from multiple tables where two table contains common values in the join condition is met.
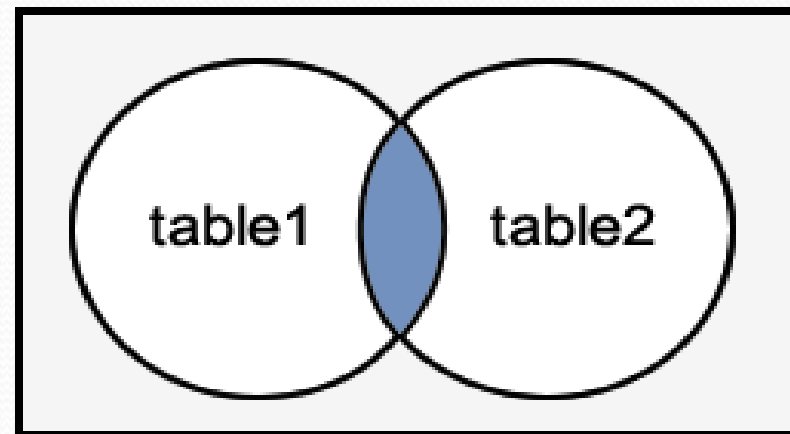
SQL Join-

❑ **LEFT JOIN** - join returns all rows from the LEFT-hand table specified in the ON condition and **only** those rows from the other table where the joined fields are equal (join condition is met).

❑ **RIGHT JOIN** - This type of join returns all rows from the RIGHT-hand table specified in the ON condition and **only** those rows from the other table where the joined fields are equal (join condition is met).
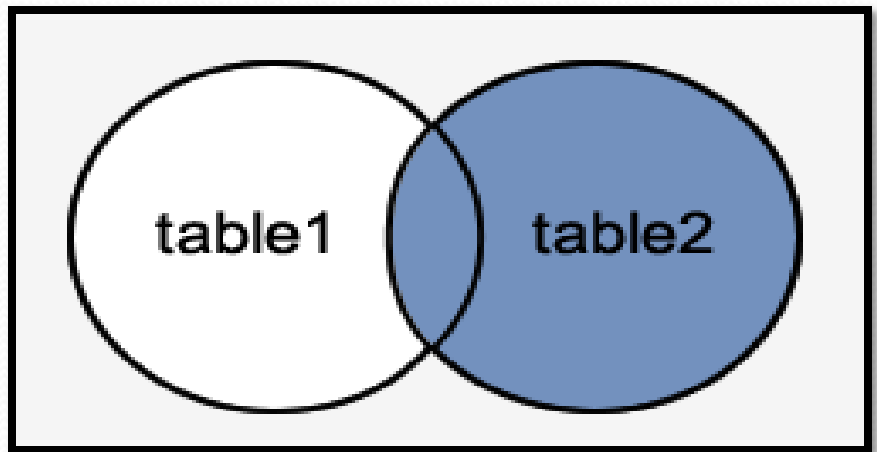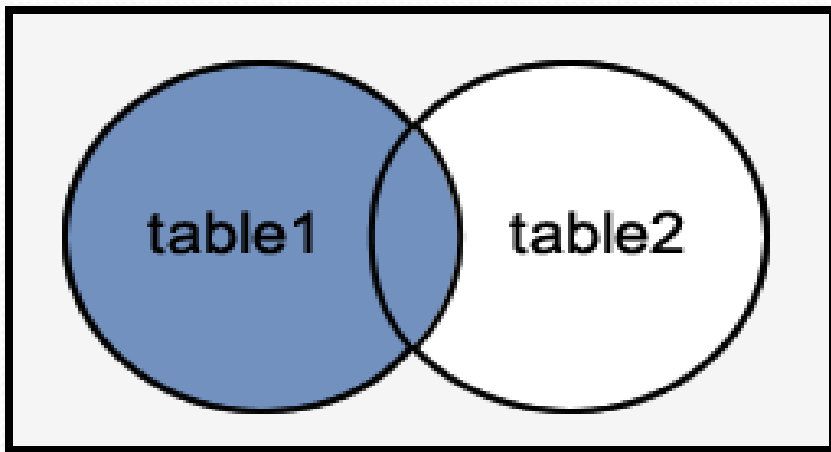
SQL Join-

❑ **FULL JOIN** - This type of join returns all rows from the LEFT-hand table and RIGHT-hand table with nulls in place where the join condition is not met.

❑ **SELF JOIN** - A self join is a join in which a table is joined with itself (which is also called Unary relationships). The self join can be viewed as a join of two copies of the same table. The table is not actually copied, but SQL performs the command as though it were.
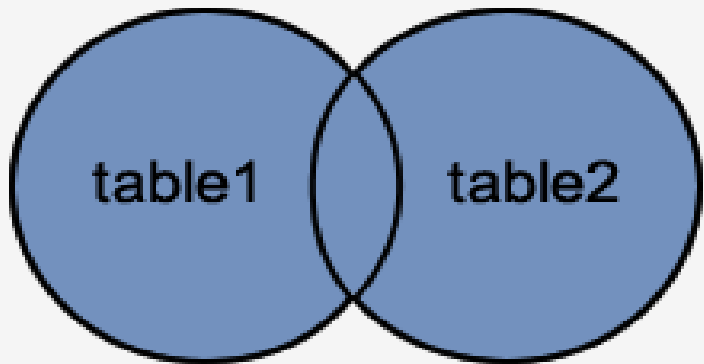
Syntax - SELECT TN1 . CN , TN2 . CN
FROM TN1 , TN2
WHERE T1 . CN = T2 . CN

SQL Group By-

❑ **Group By** - The GROUP BY statement is often used with **aggregate functions** (COUNT, MAX, MIN, SUM, AVG) to group the result-set by one or more columns.

    Syntax-   SELECT C.N.S, Aggregate_function (CN)
                             FROM   T.N
                             GROUP BY C.N.S

❑ **Having Clause-** HAVING clause is used in combination with the GROUP BY clause to provide a specific condition.

    Ex. -   SELECT Fname, Count (*)
                             FROM   VCTC
                             GROUP BY Fname
                             HAVING Fname > 1

SQL View & Index & SP -

❑ **View** –A view is a virtual table whose contents are defined by a query

    Syntax-   CREATE VIEW *view_name* AS
                      SELECT C.N.S
                      FROM   T.N.
                      WHERE *condition*;

❑ **Index** –Indexes are used to retrieve data from the database more quickly than otherwise. The users cannot see the indexes, they are just used to speed up searches/queries.
    Syntax-   CREATE INDEX *index_name*
                      ON   T.N. (C.N);

    Ex-   CREATE INDEX Vctclist
                      ON   VCTC (StudentID);

42

SQL Questions-

1. Explain DML and DDL?
2. How many Aggregate functions are available in SQL?
3. What is the difference in BETWEEN and IN condition operators?
4. What is the difference between the HAVING clause and WHERE clause?
5. What is the difference between DELETE, TRUNCATE & DROP ?
6. What are different Clauses used in SQL?
7. What are different SQL constraints?
8. What is the difference between UNIQUE key, PRIMARY KEY & FORGIN KEY constraints?
9. What are different JOINS used in SQL?

SQL Questions-

10. How to write a query to show the details of a student from Students table whose name start with K?

11. What is the syntax to add a record to a table?

12. What is the syntax of GROUP BY in SQL?

13. Define the SQL DELETE statement.

14. Write a SQL SELECT query that only returns each name only once from a table?

15. Write an SQL query to get the first maximum salary of an employee from a table named employee_table.

16. Write an SQL query to get the second maximum salary of an employee from a table named employee_table.

17. Write an SQL query to get the third maximum salary of an employee from a table named employee_table.

SQL Questions-

18. Write an SQL query to fetch unique values from a table?

19. Write an SQL query to fetch data from table whose name start with Vipul & Krishna?

20 Write an SQL query to print details of the Workers whose SALARY lies between 100000 and 500000.

21. Write an SQL query to print details of the Workers who have joined in Feb'2014.

22. Write an SQL query to fetch the count of employees working in the department 'Admin'.

23. Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.

24. What do you mean by Stored Procedures? How do we use it?

25. What are the Indexes & Views in SQL?

26. What is schema?

SQL Questions-

For practice see SQL quires in below link

https://www.techbeamers.com/sql-query-questions-answers-for-practice/