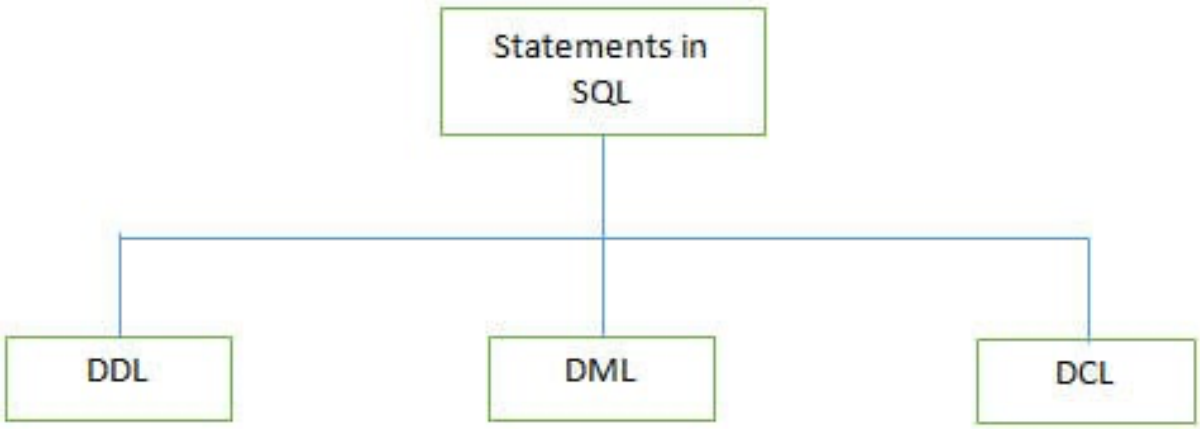
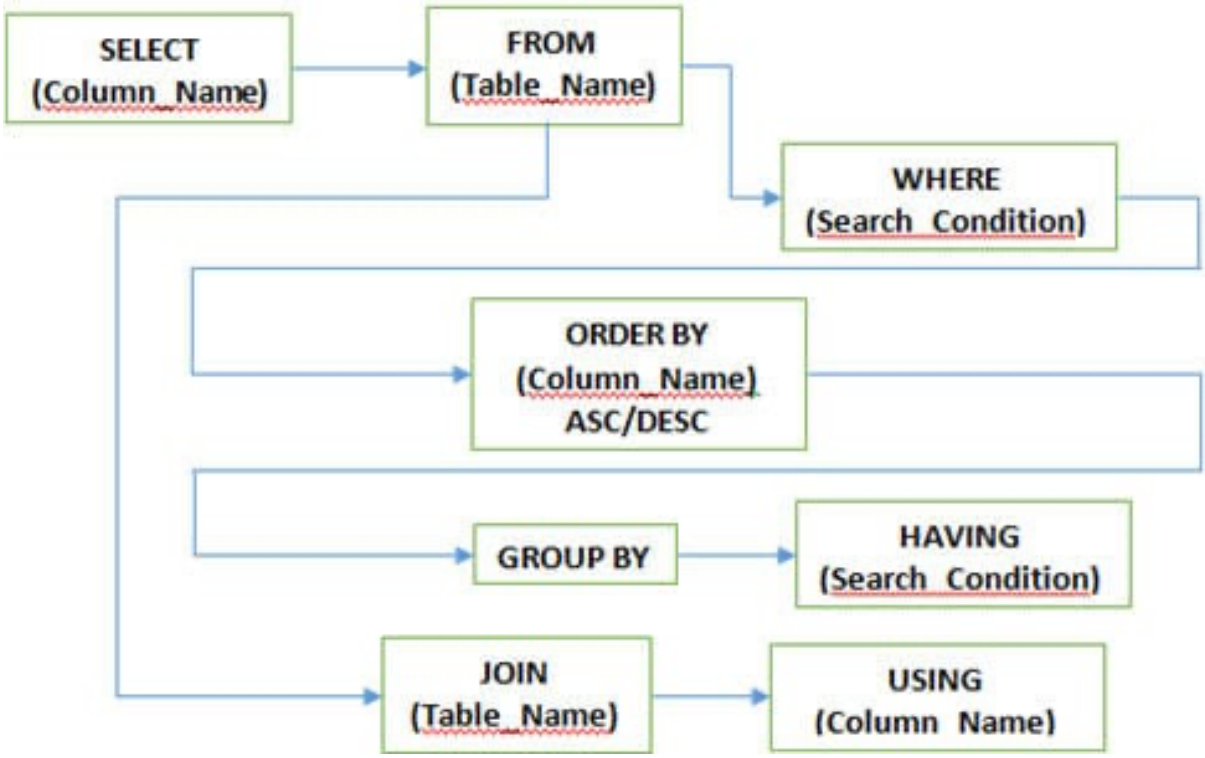
1) What is SQL?
Answer: Structured Query Language is a database tool which is used to create and access database to support software application.
2) What are tables in SQL?
Answer: The table is a collection of record and its information at a single view.
3) What are different types of statements supported by SQL?
Answer:



There are 3 types of SQL statements
1) DDL (Data Definition Language): It is used to define the database structure such as tables. It includes three statements such as Create, Alter, and Drop.
Some of the DDL Commands are listed below
CREATE: It is used for creating the table.
CREATE TABLE table_name
column_name1 data_type(size),
column_name2 data_type(size),
column_name3 data_type(size),
ALTER: The ALTER table is used for modifying the existing table object in the database.
ALTER TABLE table_name
ADD column_name datatype
OR
ALTER TABLE table_name
DROP COLUMN column_name
2) DML (Data Manipulation Language): These statements are used to manipulate the data in records. Commonly used DML statements are Insert, Update, and Delete.
The Select statement is used as partial DML statement that is used to select all or relevant records in the table.
3) DCL (Data Control Language): These statements are used to set privileges such as Grant and Revoke database access permission to the specific user.
4) How do we use DISTINCT statement? What is its use?
Answer:
The DISTINCT statement is used with the SELECT statement. If the records contain duplicate values then DISTINCT is used to select different values among duplicate records.
Syntax:
SELECT DISTINCT column_name(s)
FROM table_name;
5) What are different Clauses used in SQL?

Answer:



WHERE Clause: This clause is used to define the condition, extract and display only those records which fulfill the given condition

Syntax:

```
SELECT column_name(s)
 FROM table_name
 WHERE condition;
```

GROUP BY Clause: It is used with SELECT statement to group the result of the executed query using the value specified in it. It matches the value with the column name in tables and groups the end result accordingly.

Syntax:

```
SELECT column_name(s)
 FROM table_name
 GROUP BY column_name;
```

HAVING clause: This clause is used in association with the GROUP BY clause. It is applied to each group of result or the entire result as a single group and much similar as WHERE clause, the only difference is you cannot use it without GROUP BY clause

Syntax:

```
 SELECT column_name(s)
 FROM table_name
 GROUP BY column_name
 HAVING condition;
```

ORDER BY clause: This clause is to define the order of the query output either in ascending (ASC) or in descending (DESC) order. Ascending (ASC) is the default one but descending (DESC) is set explicitly.

Syntax:

```
SELECT column_name(s)
 FROM table_name
```

WHERE condition
ORDER BY column_name ASC|DESC;

USING clause: USING clause comes in use while working with SQL Joins. It is used to check equality based on columns when tables are joined. It can be used instead ON clause in Joins.

Syntax:

```
SELECT column_name(s)
FROM table_name
JOIN table_name
USING (column_name);
```

6) Why do we use SQL constraints? Which constraints we can use while creating a database in SQL?

Answer:

Constraints are used to set the rules for all records in the table. If any constraints get violated then it can abort the action that caused it.

Constraints are defined while creating the database itself with CREATE TABLE statement or even after the table is created once with ALTER TABLE statement.

There are 5 major constraints are used in SQL, such as

NOT NULL: That indicates that the column must have some value and cannot be left null

UNIQUE: This constraint is used to ensure that each row and column has unique value and no value is being repeated in any other row or column

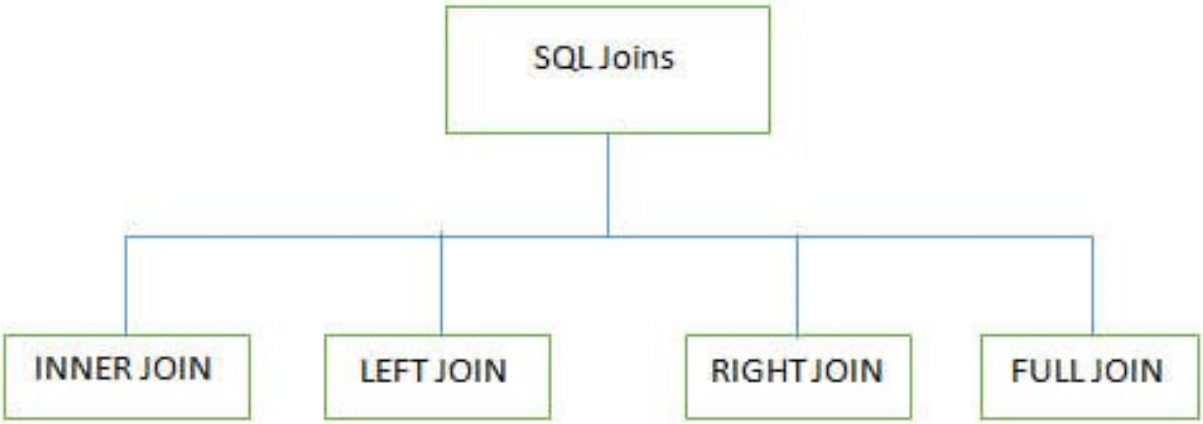
PRIMARY KEY: This constraint is used in association with NOT NULL and UNIQUE constraints such as on one or the combination of more than one column to identify the particular record with a unique identity.

FOREIGN KEY: It is used to ensure the referential integrity of data in the table and also matches the value in one table with another using Primary Key

CHECK: It is used to ensure whether the value in columns fulfills the specified condition

7) What are different JOINS used in SQL?

Answer:



There are 4 major types of joins made to use while working on multiple tables in SQL databases

INNER JOIN: It is also known as SIMPLE JOIN which returns all rows from BOTH tables when it has at least one column matched

Syntax:

```
SELECT column_name(s)
FROM table_name1
INNER JOIN table_name2
```

ON column_name1=column_name2;

Example

In this example, we have a table Employee with the following data

Emp_Id	Last_Name	First_Name	Job_Role
E0011	Verma	Akhil	Administration
E0012	Samson	Nikita	Asst. Manager
E0013	Jordan	Nil	In charge
E0014	Smith	Joe	Technician

The second Table is joining

Emp_Id	Last_Name	First_Name	Joining_Date
E0012	Verma	Akhil	2016/04/18
E0013	Samson	Nikita	2016/04/19
E0014	Jordan	Nil	2016/05/01

Enter the following SQL statement

```
1 SELECT Employee.Emp_id, Joining.Joining_Date
2   FROM Employee

3   INNER JOIN Joining
4   ON Employee.Emp_id = Joining.Emp_id

5   ORDER BY Employee.Emp_id;
```

There will be 4 records selected. These are the results that you should see

Emp_Id	Joining_Date
E0012	2016/04/18
E0013	2016/04/19
E0014	2016/05/01

Employee and orders tables where there is a matching customer_id value in both the Employee and orders tables

LEFT JOIN (LEFT OUTER JOIN): This join returns all rows from a LEFT table and its matched rows from a RIGHT table.

Syntax:

```
SELECT column_name(s)
 FROM table_name1
```

LEFT JOIN table_name2
ON column_name1=column_name2;

Example

In this example, we have a table Employee with the following data:

Emp_Id	Last_Name	First_Name	Job_Role
E0011	Verma	Akhil	Administration
E0012	Samson	Nikita	Asst. Manager
E0013	Jordan	Nil	In charge
E0014	Smith	Joe	Technician

Second Table is joining

Emp_Id	Last_Name	First_Name	Joining_Date
E0012	Verma	Akhil	2016/04/18
E0013	Samson	Nikita	2016/04/19
E0014	Jordan	Nil	2016/05/01
NULL	NULL	NULL	2016/03/01

Enter the following SQL statement

```
1 SELECT Employee.Emp_id, Joining.Joining_Date
2 FROM Employee

3 LEFT OUTER JOIN Joining
4 ON Employee.Emp_id = Joining.Emp_id

5 ORDER BY Employee.Emp_id;
```

There will be 4 records selected. These are the results that you should see:

Emp_Id	Joining_Date
NULL	*NULL*
E0012	2016/04/18
E0013	2016/04/19
E0014	2016/05/01

RIGHT JOIN (RIGHT OUTER JOIN): This joins returns all rows from the RIGHT table and its matched rows from a LEFT table.

Syntax:

```
SELECT column_name(s)
FROM table_name1
RIGHT JOIN table_name2
ON column_name1=column_name2;
```

Example

In this example, we have a table Employee with the following data

Emp_Id	Last_Name	First_Name	Job_Role
E0011	Verma	Akhil	Administration
E0012	Samson	Nikita	Asst. Manager
E0013	Jordan	Nil	In charge
E0014	Smith	Joe	Technician

The second Table is joining

Emp_Id	Last_Name	First_Name	Joining_Date
E0012	Verma	Akhil	2016/04/18
E0013	Samson	Nikita	2016/04/19
E0014	Jordan	Nil	2016/05/01
NULL	NULL	NULL	2016/03/01

Enter the following SQL statement

```
1 SELECT Employee.Emp_id, Joining.Joining_Date
2 FROM Employee

3 LEFT OUTER JOIN Joining
4 ON Employee.Emp_id = Joining.Emp_id

5 ORDER BY Employee.Emp_id;
```

There will be 4 records selected. These are the results that you should see

Emp_Id	Joining_Date
NULL	2016/03/01
E0012	2016/04/18
E0013	2016/04/19
E0014	2016/05/01

FULL JOIN (FULL OUTER JOIN): This joins returns all when there is a match either in the RIGHT table or in the LEFT table.

Syntax:

```
SELECT column_name(s)
 FROM table_name1
 FULL OUTER JOIN table_name2
 ON column_name1=column_name2;
```

Example

In this example, we have a table Employee with the following data:

Emp_Id	Last_Name	First_Name	Job_Role
E0011	Verma	Akhil	Administration
E0012	Samson	Nikita	Asst. Manager
E0013	Jordan	Nil	In charge
E0014	Smith	Joe	Technician

Second Table is joining

Emp_Id	Last_Name	First_Name	Joining_Date
E0012	Verma	Akhil	2016/04/18
E0013	Samson	Nikita	2016/04/19
E0014	Jordan	Nil	2016/05/01
NULL	NULL	NULL	2016/03/01

Enter the following SQL statement:

```
1 SELECT Employee.Emp_id, Joining.Joining_Date
2 FROM Employee

3 FULL OUTER JOIN Joining
4 ON Employee.Emp_id = Joining.Emp_id
```

5 ORDER BY Employee.Emp_id;

There will be 8 records selected. These are the results that you should see

Emp_Id	Joining_Date
NULL	*NULL*
E0012	2016/04/18
E0013	2016/04/19
E0014	2016/05/01
NULL	2016/03/01
E0012	2016/04/18
E0013	2016/04/19
E0014	2016/05/01

8) What are transactions and their controls?

Answer:

A transaction can be defined as the sequence task that is performed on databases in a logical manner to gain certain results. Operations performed like Creating, updating, deleting records in the database come from transactions.

In simple words, we can say that a transaction means a group of SQL queries executed on database records.

There are 4 transaction controls such as

COMMIT: It is used to save all changes made through the transaction

ROLLBACK: It is used to roll back the transaction such as all changes made by the transaction are reverted back and database remains as before

SET TRANSACTION: Set the name of transaction

SAVEPOINT: It is used to set the point from where the transaction is to be rolled back

9) What are properties of the transaction?

Answer:

Properties of the transaction are known as ACID properties, such as

Atomicity: Ensures the completeness of all transactions performed. Checks whether every transaction is completed successfully if not then transaction is aborted at the failure point and the previous transaction is rolled back to its initial state as changes undone

Consistency: Ensures that all changes made through successful transaction are reflected properly on database

Isolation: Ensures that all transactions are performed independently and changes made by one transaction are not reflected on other

Durability: Ensures that the changes made in the database with committed transactions persist as it is even after a system failure

10) How many Aggregate Functions are available there in SQL?

Answer:

SQL Aggregate Functions calculates values from multiple columns in a table and returns a single value.

There are 7 aggregate functions we use in SQL

AVG(): Returns the average value from specified columns

COUNT(): Returns number of table rows

MAX(): Returns largest value among the records

MIN(): Returns smallest value among the records

SUM(): Returns the sum of specified column values

FIRST(): Returns the first value

LAST(): Returns Last value

11) What are Scalar Functions in SQL?

Answer:

Scalar Functions are used to return a single value based on the input values.

Scalar Functions are as follows

UCASE(): Converts the specified field in upper case

LCASE(): Converts the specified field in lower case

MID(): Extracts and returns character from the text field

FORMAT(): Specifies the display format

LEN(): Specifies the length of the text field

ROUND(): Rounds up the decimal field value to a number

12) What are triggers?

Answer:

Triggers in SQL is kind of stored procedures used to create a response to a specific action performed on the table such as Insert, Update or Delete. You can invoke triggers explicitly on the table in the database.

Action and Event are two main components of SQL triggers when certain actions are performed the event occurs in response to that action.

Syntax:

```
CREATE TRIGGER name {BEFORE|AFTER} (event [OR..]}
ON table_name [FOR [EACH] {ROW|STATEMENT}]
EXECUTE PROCEDURE functionname {arguments}
```

13) What is View in SQL?

Answer:

A View can be defined as a virtual table that contains rows and columns with fields from one or more table.

Syntax:

```
CREATE VIEW view_name AS
 SELECT column_name(s)
 FROM table_name
 WHERE condition
```

14) How we can update the view?

Answer:

SQL CREATE and REPLACE can be used for updating the view.

Following query syntax is to be executed to update the created view
Syntax:

```
CREATE OR REPLACE VIEW view_name AS
 SELECT column_name(s)
 FROM table_name
 WHERE condition
```

15) Explain the working of SQL Privileges?
Answer:
SQL GRANT and REVOKE commands are used to implement privileges in SQL multiple user environments. The administrator of the database can grant or revoke privileges to or from users of database object like SELECT, INSERT, UPDATE, DELETE, ALL etc.
GRANT Command: This command is used provide database access to user apart from an administrator.
Syntax:

```
GRANT privilege_name
 ON object_name
 TO {user_name|PUBLIC|role_name}
 [WITH GRANT OPTION];
```

In above syntax WITH GRANT OPTIONS indicates that the user can grant access to another user too.
REVOKE command: This command is used provide database deny or remove access to database objects.
Syntax:

```
REVOKE privilege_name
 ON object_name
 FROM {user_name|PUBLIC|role_name};
```

16) How many types of Privileges are available in SQL?
Answer:
There are two types of privileges used in SQL, such as
System Privilege: System privileges deal with an object of a particular type and specifies the right to perform one or more actions on it which include Admin allows a user to perform administrative tasks, ALTER ANY INDEX, ALTER ANY CACHE GROUP CREATE/ALTER/DELETE TABLE, CREATE/ALTER/DELETE VIEW etc.
Object Privilege: This allows to perform actions on an object or object of another user(s) viz. table, view, indexes etc. Some of the object privileges are EXECUTE, INSERT, UPDATE, DELETE, SELECT, FLUSH, LOAD, INDEX, REFERENCES etc.
17) What is SQL Injection?
Answer:
SQL Injection is a type of database attack technique where malicious SQL statements are inserted into an entry field of database such that once it is executed the database is opened for an attacker. This technique is usually used for attacking Data-Driven Applications to have access to sensitive data and perform administrative tasks on databases.
For Example: SELECT column_name(s) FROM table_name WHERE condition;
18) What is SQL Sandbox in SQL Server?
Answer:
SQL Sandbox is the safe place in SQL Server Environment where untrusted scripts are executed. There are 3 types of SQL sandbox, such as

Safe Access Sandbox: Here a user can perform SQL operations such as creating stored procedures, triggers etc. but cannot have access to the memory and cannot create files.
External Access Sandbox: User can have access to files without having a right to manipulate the memory allocation.
Unsafe Access Sandbox: This contains untrusted codes where a user can have access to memory.

19) What is the difference between SQL and PL/SQL?

Answer:

SQL is a structured query language to create and access databases whereas PL/SQL comes with procedural concepts of programming languages.

20) What is the difference between SQL and MySQL?

Answer:

SQL is a structured query language that is used for manipulating and accessing the relational database, on the other hand, MySQL itself is a relational database that uses SQL as the standard database language.

21) What is the use of NVL function?

Answer:

NVL function is used to convert the null value to its actual value.

22) What is the Cartesian product of the table?

Answer:

The output of Cross Join is called as a Cartesian product. It returns rows combining each row from the first table with each row of the second table. For Example, if we join two tables having 15 and 20 columns the Cartesian product of two tables will be 15×20=300 Rows.

23) What do you mean by Subquery?

Answer:

Query within another query is called as Subquery. A subquery is called inner query which returns output that is to be used by another query.

24) How many row comparison operators are used while working with a subquery?

Answer:

There are 3-row comparison operators that are used in subqueries such as IN, ANY and ALL.

25) What is the difference between clustered and non-clustered indexes?

Answer:

One table can have only one clustered index but multiple nonclustered indexes.
Clustered indexes can be read rapidly rather than non-clustered indexes.
Clustered indexes store data physically in the table or view and non-clustered indexes do not store data in the table as it has separate structure from the data row

26) What is the difference between DELETE and TRUNCATE?

Answer:

The basic difference in both is DELETE is DML command and TRUNCATE is DDL
DELETE is used to delete a specific row from the table whereas TRUNCATE is used to remove all rows from the table
We can use DELETE with WHERE clause but cannot use TRUNCATE with it

27) What is the difference between DROP and TRUNCATE?

Answer:

TRUNCATE removes all rows from the table which cannot be retrieved back, DROP removes the entire table from the database and it cannot be retrieved back.

28) How to write a query to show the details of a student from Students table whose name starts with K?
Answer:
SELECT * FROM Student WHERE Student_Name like 'K%';
Here 'like' operator is used for pattern matching.
29) What is the difference between Nested Subquery and Correlated Subquery?
Answer:
Subquery within another subquery is called as Nested Subquery. If the output of a subquery is depending on column values of the parent query table then the query is called Correlated Subquery.
SELECT adminid(SELEC Firstname+' '+Lastname FROM Employee WHERE
empid=emp. adminid)AS EmpAdminId FROM Employee
This query gets details of an employee from the Employee table.
30) What is Normalization? How many Normalization forms are there?
Answer:
Normalization is used to organize the data in such a manner that data redundancy will never occur in the database and avoid insert, update and delete anomalies.
There are 5 forms of Normalization
First Normal Form (1NF): It removes all duplicate columns from the table. Creates a table for related data and identifies unique column values
First Normal Form (2NF): Follows 1NF and creates and places data subsets in an individual table and defines the relationship between tables using the primary key
Third Normal Form (3NF): Follows 2NF and removes those columns which are not related through primary key
Fourth Normal Form (4NF): Follows 3NF and do not define multi-valued dependencies. 4NF also known as BCNF
31) What is a Relationship? How many types of Relationships are there?
Answer:
The relationship can be defined as the connection between more than one tables in the database.
There are 4 types of relationships
One to One Relationship
Many to One Relationship
Many to Many Relationship
One to Many Relationship
32) What do you mean by Stored Procedures? How do we use it?
Answer:
A stored procedure is a collection of SQL statements that can be used as a function to access the database. We can create these stored procedures previously before using it and can execute these them wherever we require and also apply some conditional logic to it. Stored procedures are also used to reduce network traffic and improve performance.
Syntax:
```
CREATE Procedure Procedure_Name
 (
 //Parameters
 )
 AS
```

BEGIN
SQL statements in stored procedures to update/retrieve records
END

33) State some properties of Relational databases?

Answer:

In relational databases, each column should have a unique name
The sequence of rows and columns in relational databases are insignificant
All values are atomic and each row is unique

34) What are Nested Triggers?

Answer:

Triggers may implement data modification logic by using INSERT, UPDATE, and DELETE statements. These triggers that contain data modification logic and find other triggers for data modification are called Nested Triggers.

35) What is a Cursor?

Answer:

A cursor is a database object which is used to manipulate data in a row-to-row manner.
Cursor follows steps as given below
Declare Cursor
Open Cursor
Retrieve row from the Cursor
Process the row
Close Cursor
Deallocate Cursor

36) What is Collation?

Answer:

Collation is a set of rules that check how the data is sorted by comparing it. Such as Character data is stored using correct character sequence along with case sensitivity, type, and accent.

37) What do we need to check in Database Testing?

Answer:

Generally, in Database Testing following thing is need to be tested
Database Connectivity
Constraint Check
Required Application Field and its size
Data Retrieval and Processing With DML operations
Stored Procedures
Functional flow

38) What is Database White Box Testing?

Answer:

Database White Box Testing involves
Database Consistency and ACID properties
Database triggers and logical views
Decision Coverage, Condition Coverage, and Statement Coverage
Database Tables, Data Model, and Database Schema
Referential integrity rules

39) What is Database Black Box Testing?

Answer:

Database Black Box Testing involves
Data Mapping
Data stored and retrieved
Use of Black Box techniques such as Equivalence Partitioning and Boundary Value Analysis (BVA)

40) What are Indexes in SQL?
Answer: The index can be defined as the way to retrieve the data more quickly. We can define indexes using CREATE statements.
Syntax:

```
CREATE INDEX index_name
 ON table_name (column_name)
```

Further, we can also create Unique Index using following syntax;
Syntax:

```
CREATE UNIQUE INDEX index_name
 ON table_name (column_name)
```

****
UPDATE: We have added few more short questions for practice.
41) What does SQL stand for?
Answer: SQL stands for Structured Query Language.
42) How to select all records from the table?
Answer: To select all the records from the table we need to use the following syntax:

```
Select * from table_name;
```

43) Define join and name different types of joins?
Answer: Join keyword is used to fetch data from related two or more tables. It returns rows where there is at least one match in both the tables included in the join. Read more here.
Type of joins are:
Right Join
Outer Join
Full Join
Cross Join
Self Join.
44) What is the syntax to add a record to a table?
Answer: To add a record in a table INSERT syntax is used.
Example: INSERT into table_name VALUES (value1, value2..);
45) How do you add a column to a table?
Answer: To add another column in the table following command has been used.

```
ALTER TABLE table_name ADD (column_name);
```

46) Define SQL Delete statement.
Answer: Delete is used to delete a row or rows from a table based on the specified condition.
The basic syntax is as follows:

```
DELETE FROM table_name
WHERE <Condition>
```

47) Define COMMIT?
Answer: COMMIT saves all changes made by DML statements.

48) What is the Primary key?
Answer: A Primary key is a column whose values uniquely identify every row in a table. Primary key values can never be reused.
49) What are Foreign keys?
Answer: When a one table's primary key field is added to related tables in order to create the common field which relates the two tables, it called a foreign key in other tables. Foreign Key constraints enforce referential integrity.
50) What is CHECK Constraint?
Answer: A CHECK constraint is used to limit the values or type of data that can be stored in a column. They are used to enforce domain integrity.
51) Is it possible for a table to have more than one foreign key?
Answer: Yes, a table can have many foreign keys and only one primary key.
52) What are the possible values for the BOOLEAN data field?
Answer: For a BOOLEAN data field, two values are possible: -1(true) and 0(false).
53) What is a stored procedure?
Answer: A stored procedure is a set of SQL queries which can take input and send back output.
54) What is identity in SQL?
Answer: An identity column in the SQL automatically generates numeric values. We can define a start and increment value of the identity column.
55) What is Normalization?
Answer: The process of table design to minimize the data redundancy is called normalization. We need to divide a database into two or more table and define relationships between them.
56) What is a Trigger?
Answer: The Trigger allows us to execute a batch of SQL code when a table event occurs (Insert, update or delete command executed against a specific table)
57) How to select random rows from a table?
Answer: Using a SAMPLE clause we can select random rows.
Example:
SELECT * FROM table_name SAMPLE(10);
58) Which TCP/IP port does SQL Server run?
Answer: By default SQL Server runs on port 1433.
59) Write a SQL SELECT query that only returns each name only once from a table?
Answer: To get each name only once, we need to use the DISTINCT keyword.
SELECT DISTINCT name FROM table_name;
60) Explain DML and DDL?
Answer: DML stands for Data Manipulation Language. INSERT, UPDATE and DELETE are DML statements.
DDL stands for Data Definition Language. CREATE, ALTER, DROP, RENAME are DDL statements.
61) Can we rename a column in the output of SQL query?
Answer: Yes using the following syntax we can do this.
SELECT column_name AS new_name FROM table_name;
62) Give the order of SQL SELECT?
Answer: Order of SQL SELECT clauses is: SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY. Only the SELECT and FROM clause are mandatory.

63) Suppose a Student column has two columns, Name and Marks. How to get name and marks of the top three students.
Answer: SELECT Name, Marks FROM Student s1 where 3 <= (SELECT COUNT(*) FROM Students s2 WHERE s1.marks = s2.marks)
64) What is SQL comments?
Answer: SQL comments can be put by two consecutive hyphens (–).
65) Difference between TRUNCATE, DELETE and DROP commands?
Answer: DELETE removes some or all rows from a table based on the condition. It can be rolled back.
TRUNCATE removes ALL rows from a table by de-allocating the memory pages. The operation cannot be rolled back
DROP command removes a table from the database completely.
66) What are the properties of a transaction?
Answer: Generally, these properties are referred to as ACID properties. They are:
Atomicity
Consistency
Isolation
Durability.
67) What do you mean by ROWID?
Answer: It's an 18 character long pseudo column attached with each row of a table.
68) Define UNION, MINUS, UNION ALL, INTERSECT ?
Answer: MINUS – returns all distinct rows selected by the first query but not by the second.
UNION – returns all distinct rows selected by either query
UNION ALL – returns all rows selected by either query, including all duplicates.
INTERSECT – returns all distinct rows selected by both queries.
69) What is a transaction?
Answer: A transaction is a sequence of code that runs against a database. It takes the database from one consistent state to another.
70) What is the difference between UNIQUE and PRIMARY KEY constraints?
Answer: A table can have only one PRIMARY KEY whereas there can be any number of UNIQUE keys.
The primary key cannot contain Null values whereas Unique key can contain Null values.
71) What is a composite primary key?
Answer: Primary key created on more than one column is called composite primary key.
72) What is an Index?
Answer: An Index is a special structure associated with a table speed up the performance of queries. The index can be created on one or more columns of a table.
73) What is the Subquery?
Answer: A Subquery is a subset of select statements whose return values are used in filtering conditions of the main query.
74) What do you mean by query optimization?
Answer: Query optimization is a process in which a database system compares different query strategies and select the query with the least cost.
75) What is Collation?
Answer: Set of rules that define how data is stored, how case sensitivity and Kana character can be treated etc.

76) What is Referential Integrity?
Answer: Set of rules that restrict the values of one or more columns of the tables based on the values of the primary key or unique key of the referenced table.
77) What is Case Function?
Answer: Case facilitates if-then-else type of logic in SQL. It evaluates a list of conditions and returns one of the multiple possible result expressions.
78) Define a temp table?
Answer: A temp table is a temporary storage structure to store the data temporarily.
79) How can we avoid duplicating records in a query?
Answer: By using DISTINCT keyword duplicating records in a query can be avoided.
80) Explain the difference between Rename and Alias?
Answer: Rename is a permanent name given to a table or column whereas Alias is a temporary name given to a table or column.
81) What is a View?
Answer: A view is a virtual table which contains data from one or more tables. Views restrict data access of the table by selecting only required values and make complex queries easy.
8). What are the advantages of Views?
Answer: Advantages of Views:
Views restrict access to the data because the view can display selective columns from the table.
Views can be used to make simple queries to retrieve the results of complicated queries. For example, views can be used to query information from multiple tables without the user knowing.
83) List the various privileges that a user can grant to another user?
Answer: SELECT, CONNECT, RESOURCES.
84) What is schema?
Answer: A schema is a collection of database objects of a User.
85) What is a Table?
Answer: A table is the basic unit of data storage in the database management system. Table data is stored in rows and columns.
86) Does View contain Data?
Answer: No, Views are virtual structure.
87) Can a View based on another View?
Answer: Yes, A View is based on another View.
88) What is the difference between Having clause and Where clause?
Answer: Both specify a search condition but Having clause is used only with the SELECT statement and typically used with GROUP BY clause.
If GROUP BY clause is not used then Having behaved like WHERE clause only.
89) What is the difference between Local and Global temporary table?
Answer: If defined in inside a compound statement a local temporary table exists only for the duration of that statement but a global temporary table exists permanently in the DB but its rows disappear when the connection is closed.
90) What is CTE?
Answer: A CTE or common table expression is an expression which contains temporary result set which is defined in a SQL statement.