CONCEPTUAL QUESTIONS
For
RDBMS -SQL

Website: www.analytixlabs.co.in

Email: info@analytixlabs.co.in

Disclaimer: This material is protected under copyright act AnalytixLabs©, 2011-2023. Unauthorized use and/ or duplication of this material or any part of this material including data, in any form without explicit and written permission from AnalytixLabs is strictly prohibited. Any violation of this copyright will attract legal actions.

SQL NOTES

The **select** statement is used to query the database and retrieve selected data that match the criteria that you specify. The column names that follow the **select** keyword determine which columns will be returned in the results. You can select as many column names that you'd like, or you can use a "*" to select all columns. The table name that follows the keyword **from** specifies the table that will be queried to retrieve the desired results.

The **where** clause (optional) specifies which data values or rows will be returned or displayed, based on the criteria described after the keyword **where** Conditional selections used in the **where** clause:

=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal to
LIKE	*See note below

The **LIKE** pattern matching operator can also be used in the conditional selection of the **where** clause. Like is a very powerful operator that allows you to select only rows that are "like" what you specify. The percent sign "%" can be used as a wild card to match any possible character that might appear before or after the characters specified. The **create table** statement is used to create a new table.

Example:

```
create table employee
(first varchar(15), last varchar(20), age number(3), address varchar(30),
 city varchar(20),
 state varchar(20));
```

The **insert** statement is used to insert or add a row of data into the table.

Example:

```
insert into employee
  (first, last, age, address, city, state)
  values ('Luke', 'Duke', 45, '2130 Boars Nest',
          'Hazard Co', 'Georgia');
```

The **update** statement is used to update or change records that match specified criteria. This is accomplished by carefully constructing a where clause.

Examples:

```
update phone_book
  set area_code = 623
 where prefix = 979;
```

The **delete** statement is used to delete records or rows from the table.

Examples:

```
delete from employee;
   where firstname = 'Mike' or firstname = 'Eric';
```

The **drop table** command is used to delete a table and all rows in the table.

Example:

```
drop table myemployees_ts0211;
```

SELECT Statement

The SELECT statement is used to query the database and retrieve selected data that match the criteria that you specify. The SELECT statement has five main clauses to choose from, although, FROM is the only required clause. Each of the clauses has a vast selection of options, parameters, etc. The clauses will be listed below.

Here is the format of the SELECT statement:

SELECT [ALL | DISTINCT] *column1*[,*column2*]
FROM *table1*[,*table2*]
[WHERE "conditions"]
[GROUP BY "column-list"]
[HAVING "conditions]
[ORDER BY "column-list" [ASC | DESC]]

Example:

```
SELECT name, age, salary
FROM employee
WHERE age > 50;
```

Aggregate Functions

MIN	returns the smallest value in a given column
MAX	returns the largest value in a given column
SUM	returns the sum of the numeric values in a given column
AVG	returns the average value of a given column
COUNT	returns the total number of values in a given column
COUNT(*)	returns the number of rows in a table

Example:

```
SELECT AVG(salary)
FROM employee;
WHERE title = 'Programmer';
```

GROUP BY clause:

The GROUP BY clause will gather all of the rows together that contain data in the specified column(s) and will allow aggregate functions to be performed on the one or more columns.

This can best be explained by an example:

```
SELECT max(salary), dept FROM employee GROUP BY dept;
```

Exercise #1

```
SELECT state, count(state)
FROM customers
GROUP BY state;
```

HAVING clause

The HAVING clause allows you to specify conditions on the rows for each group - in other words, which rows should be selected will be based on the conditions you specify. The HAVING clause should follow the GROUP BY clause if you are going to use it.

```
SELECT dept, avg(salary)
FROM employee
GROUP BY dept
HAVING avg(salary) > 20000;
```

ORDER BY clause

ORDER BY is an optional clause, which will allow you to display the results of your query in a sorted order (either ascending order or descending order) based on the columns that you specify to order by.

```
SELECT employee_id, dept, name, age, salary
FROM employee_info

WHERE dept = 'Sales'
ORDER BY salary, age DESC;
```

Combining conditions and Boolean Operators:

The AND operator can be used to join two or more conditions in the WHERE clause. Both sides of the AND condition must be true in order for the condition to be met and for those rows to be displayed. The OR operator can be used to join two or more conditions in the WHERE clause also. However, **either** side of the OR operator can be true and the condition will be met - hence, the rows will be displayed. With the OR operator, either side can be true or both sides can be true.

For example:

```
SELECT employeeid, firstname, lastname, title, salary
FROM employee_info
WHERE salary >= 50000.00 AND title = 'Programmer';
```

Another Example:

```
SELECT firstname, lastname, title, salary
FROM employee_info
WHERE (title = 'Sales') OR (title = 'Programmer');
```

Q. What is SQL?
Ans: SQL stands for Structured Query Language, and it is used to communicate with the Database. This is a standard language used to perform tasks such as retrieval, updation, insertion and deletion of data from a database. Standard SQL Commands are Select statements. SQL is an ANSI (American National Standards Institute) standard computer language for accessing and manipulating database systems.

Q. What are the specific uses of SQL functions?
Ans: SQL functions have the following uses –
Performing calculations on data
Modifying individual data items
Manipulating the output
Formatting dates and numbers
Converting data types

Q. What is the purpose of the group functions in SQL? Give some examples of group functions.
Ans: Group functions in SQL work on sets of rows and returns one result per group. Examples of group functions are AVG, COUNT, MAX, MIN, STDDEV, SUM, and VARIANCE.

Q. What is the difference between SQL and MySQL or SQL Server?
Ans: SQL or Structured Query Language is a language; language that communicates with a relational database thus providing ways of manipulating and creating databases. Mysql and Microsoft's SQL Server both are relational database management systems that use SQL as their standard relational database language.

Q. What is the difference between SQL and PL/SQL?
Ans: PL/SQL is a dialect of SQL that adds procedural features of programming languages in SQL. It was developed by Oracle Corporation in the early 90's to enhance the capabilities of SQL.

Q. What is a Database?
Ans: Database is nothing but an organized form of data for easy access, storing, retrieval and managing of data. This is also known as structured form of data which can be accessed in many ways.
Example: School Management Database, Bank Management Database.

Q. What is a Database system?
Ans: The database and DBMS software together is called as Database system.

Q. What are the advantages of DBMS?
Ans: Redundancy is controlled.

1. Unauthorized access is restricted.
2. Providing multiple user interfaces.
3. Enforcing integrity constraints.
4. Providing backup and recovery.

Q. What is RDBMS?
Ans: RDBMS stands for Relational Database Management System. RDBMS store the data into the collection of tables, which is related by common fields between the columns of the table. It also provides relational operators to manipulate the data stored into the tables. **Example: SQL Server.**

Q. What's difference between DBMS and RDBMS?
Ans: DBMS provides a systematic and organized way of storing, managing and retrieving from collection of logically related information. RDBMS also provides what DBMS provides but above that it provides relationship integrity.

So in short we can say RDBMS = DBMS + REFERENTIAL INTEGRITY

These relations are defined by using “Foreign Keys” in any RDBMS. Many DBMS companies claimed there DBMS product was a RDBMS compliant, but according to industry rules and regulations if the DBMS fulfills the twelve CODD rules it’s truly a RDBMS. Almost all DBMS (SQLSERVER, ORACLE etc) fulfills all the twelve CODD rules and are considered as truly RDBMS.

Q. What is NOT NULL Constraint?
Ans: A NOT NULL constraint enforces that the column will not accept null values. The not null constraints are used to enforce domain integrity, as the check constraints.

Q. What is database or database management systems (DBMS)? What’s the difference between file and database? Can files qualify as a database?
Ans: Database provides a systematic and organized way of storing, managing and retrieving from collection of logically related information.
Secondly the information has to be persistent, that means even after the application is closed the information should be persisted.

Finally it should provide an independent way of accessing data and should not be dependent on the application to access the information.

Main difference between a simple file and database that database has independent way (SQL) of accessing information while simple files do not File meets the storing, managing and retrieving part of a database but not the independent way of accessing data. Many experienced programmers think that the main difference is that file cannot provide multi-user capabilities which a DBMS provides. But if we look at some old COBOL and C programs where file where the only means of storing data, we can see functionalities like locking, multi-user etc provided very efficiently. So it’s a matter of debate if some interviewers think this as a main difference between files and database accept it… going in to debate is probably losing a job.

Q. What are the disadvantages in File Processing System?
Ans: The disadvantages of File processing system are

1. Data redundancy and inconsistency.
2. Difficult in accessing data.
3. Data isolation.
4. Data integrity.
5. Concurrent access is not possible.
6. Security Problems

Q. Describe the three levels of data abstraction?
Ans: The are three levels of abstraction:

1. **Physical level:** The lowest level of abstraction describes how data are stored.
2. **Logical level:** The next higher level of abstraction, describes what data are stored in database and what relationship among those data.
3. **View level:** he highest level of abstraction describes only part of entire database.

Q. What are E-R diagrams?
Ans: E-R diagram also termed as Entity-Relationship diagram shows relationship between various tables in the database.

Q. What is normalization? What are different types of normalization?

Ans: There is a set of rules that has been established to aid in the design of tables that are meant to be connected through relationships. This set of rules is known as Normalization.

Benefits of normalizing your database include:
Ans: =>Avoiding repetitive entries
=>Reducing required storage space
=>Preventing the need to restructure existing tables to accommodate new data.
=>Increased speed and flexibility of queries, sorts, and summaries.

Following are the three normal forms:-
First Normal Form
For a table to be in first normal form, data must be broken up into the smallest possible. In addition to breaking data up into the smallest meaningful values, tables first normal form should not contain repetition of groups of fields.

Second Normal form
The second normal form states that each field in a multiple field primary key table must be directly related to the entire primary key. Or in other words, each non-key field should be a fact about all the fields in the primary key.

Third normal form
A non-key field should not depend on other Non-key field.

Q. Can you explain Fourth Normal Form and Fifth Normal Form?
Ans: In fourth normal form it should not contain two or more independent multi-v about an entity and it should satisfy "Third Normal form".
Fifth normal form deals with reconstructing information from smaller pieces of information. These smaller pieces of information can be maintained with less redundancy.

Q. Have you heard about sixth normal form?
Ans: If we want relational system in conjunction with time we use sixth normal form. At this moment SQL Server does not supports it directly.

Q. What is Denormalization?
Ans: Denormalization is the process of putting one fact in numerous places (its vice-versa of normalization). Only one valid reason exists for denormalization a relational design - to enhance performance. The sacrifice to performance is that you increase redundancy in database.

Q. What are the properties of the Relational tables?
Ans: Relational tables have six properties:

1. Values are atomic.
2. Column values are of the same kind.
3. Each row is unique.
4. The sequence of columns is insignificant.
5. The sequence of rows is insignificant.
6. Each column must have a unique name.

Q. What is Data Warehousing?

Ans: Data Warehousing is a process in which the data is stored and accessed from central location and is meant to support some strategic decisions. Data Warehousing is not a requirement for Data mining. But just makes your Data mining process more efficient.

Data warehouse is a collection of integrated, subject-oriented databases designed to support the decision-support functions (DSF), where each unit of data is relevant to some moment in time.

Q. What are Data Marts?
Ans: Data Marts are smaller section of Data Warehouses. They help data warehouses collect data. For example your company has lot of branches which are spanned across the globe. Head-office of the company decides to collect data from all these branches for anticipating market. So to achieve this IT department can setup data mart in all branch offices and a central data warehouse where all data will finally reside.

Q. What are Fact tables and Dimension Tables? What is Dimensional Modeling and Star Schema Design?
Ans: When we design transactional database we always think in terms of normalizing design to its least form. But when it comes to designing for Data warehouse we think more in terms of denormalization the database. Data warehousing, Databases are designed using Dimensional Modeling. Dimensional Modeling uses the existing relational database structure and builds on that.

There are two basic tables in dimensional modeling:-
Fact Tables.
Dimension Tables.
Fact tables are central tables in data warehousing. Fact tables have the actual aggregate values which will be needed in a business process. While dimension tables revolve around fact tables. They describe the attributes of the fact tables.

Q. What is Snow Flake Schema design in database? What's the difference between Star and Snow flake schema?
Ans: Star schema is good when you do not have big tables in data warehousing. But when tables start becoming really huge it is better to denormalize. When you denormalize star schema it is nothing but snow flake design. For instance below customer address table is been normalized and is a child table of Customer table. Same holds true for Salesperson table.

Q. What is Data mining?
Ans: Data mining is a concept by which we can analyze the current data from different perspectives and summarize the information in more useful manner. It's mostly used either to derive some valuable information from the existing data or to predict sales to increase customer market.
There are two basic aims of Data mining:-
Prediction: -
From the given data we can focus on how the customer or market will perform. For instance we are having a sale of 40000 $ per month in India, if the same product is to be sold with a discount how much sales can the company expect.
Summarization: -
To derive important information to analyze the current business scenario. For example a weekly sales report will give a picture to the top management how we are performing on a weekly basis?

Q. List the different types of relationships in SQL?

One-to-One - This can be defined as the relationship between two tables where each record in one table is associated with the maximum of one record in the other table.

One-to-Many & Many-to-One - This is the most commonly used relationship where a record in a table is associated with multiple records in the other table.

Many-to-Many - This is used in cases when multiple instances on both sides are needed for defining a relationship.

Self-Referencing Relationships - This is used when a table needs to define a relationship with itself.

Q. What is the purpose of the COMMIT statement?

Ans: The COMMIT statement is used to save the changes made within a transaction. It makes the changes permanent and releases any locks held by the transaction.

Q. What is the purpose of the ROLLBACK statement?

Ans: The ROLLBACK statement is used to undo the changes made within a transaction and restore the data to its previous state.

Q. What is the difference between a view and a table?

Ans: A table is a physical storage structure that contains data, whereas a view is a virtual table derived from one or more tables. Views do

Q. What is a Temporary Table?

Ans:

- Temporary tables are physical tables that are created and stored in the tempdb database. They are used to store temporary data that persists for the duration of a session or connection.
- Temporary tables are explicitly created using the CREATE TABLE statement, and they can have columns, indexes, constraints, and other attributes like regular tables.
- Temporary tables are accessible across multiple queries within the same session or connection. They can be modified, inserted into, updated, or deleted from, just like regular tables.
- Temporary tables need to be explicitly dropped or they will be automatically dropped when the session or connection is closed or terminated.

Q. What is the difference between a CTE and Temporary tables?

Ans:

- CTEs are virtual tables defined within a single query, while temporary tables are physical tables stored in the tempdb database.
- CTEs are not accessible outside the query where they are defined, while temporary tables can be accessed and modified across multiple queries within the same session.
- CTEs are not explicitly created or dropped, while temporary tables need to be explicitly created and dropped.
- CTEs are typically used for simplifying complex queries and recursive operations, while temporary tables are used for storing intermediate results or temporary data for more complex data manipulation.

Q. What are the different types of tables used in SQL?

Ans: The following are the table types used in SQL:

Partitioned tables, Temporary tables, System tables, Wide tables

Q. What are the Set Operators?

Ans: There are four types of set operators available in SQL. They are given as follows:

Union : This operator allows combining result sets of two or more SELECT statements.

Union All : This operator allows combining result sets of two or more SELECT statements along with duplicates.

Intersect : This operator returns the common records of the result sets of two or more SELECT statements.

Minus : This operator returns the exclusive records of the first table when two tables undergo this operation.

Q. What are indexes? What are B-Trees?
Ans: Index makes your search faster. So defining indexes to your database will make your search faster. Most of the indexing fundamentals use “B-Tree” or “Balanced-Tree” principle. It’s not a principle that is something is created by SQL Server or ORACLE but is a mathematical derived fundamental. In order that “B-tree” fundamental work properly both of the sides should be balanced.

The database server uses a B-tree structure to organize index information. B-Tree generally has following types of index pages or nodes:

1. **root node:** A root node contains node pointers to branch nodes which can be only one.
2. **branch node:** A branch node contains pointers to leaf nodes or other branch nodes which can be two or more.
3. **leaf nodes:** A leaf node contains index items and horizontal pointers to other leaf nodes which can be many.

Q. I have a table which has lot of inserts, is it a good database design to create indexes on that table? Insert’s are slower on tables which have indexes, justify it? or Why does page splitting happen?
Ans: All indexing fundamentals in database use “B-tree” fundamental. Now whenever there is new data inserted or deleted the tree tries to become unbalance.
Creates a new page to balance the tree.
Shuffle and move the data to pages.

So if your table is having heavy inserts that means it’s transactional, then you can visualize the amount of splits it will be doing. This will not only increase insert time but will also upset the end-user who is sitting on the screen. So when you forecast that a table has lot of inserts it’s not a good idea to create indexes.

Q. What are the two types of indexes and explain them in detail? or What’s the difference between clustered and non-clustered indexes?
Ans: There are basically two types of indexes: Clustered Indexes & Non-Clustered Indexes.

In clustered index the non-leaf level actually points to the actual data. In Non-Clustered index the leaf nodes point to pointers (they are row id’s) which then point to actual data.

Q. What are CODD rules?
Ans: In 1969 Dr. E. F. Codd laid down some 12 rules which a DBMS should adhere in order to get the logo of a true RDBMS.

Rule 1: Information Rule.
"All information in a relational data base is represented explicitly at the logical level and inexactly one way - by values in tables."

Rule 2: Guaranteed access Rule.
"Each and every datum (atomic value) in a relational data base is guaranteed to be logically accessible by resorting to a combination of table name, primary key value and column name" In flat files we have to parse and know exact location of field values. But if a DBMS is truly RDBMS you can access the value by specifying the table name, field name, for instance

Customers.Fields [‘Customer Name’].

Rule 3: Systematic treatment of null values.

"Null values (distinct from the empty character string or a string of blank characters and distinct from zero or any other number) are supported in fully relational DBMS for representing missing information and inapplicable information in a systematic way, independent of data type."

Rule 4: Dynamic on-line catalog based on the relational model.
"The data base description is represented at the logical level in the same way as ordinary data, so that authorized users can apply the same relational language to its interrogation as they apply to the regular data." The Data Dictionary is held within the RDBMS, thus there is no-need for off-line volumes to tell you the structure of the database.

Rule 5: Comprehensive data sub-language Rule.
"A relational system may support several languages and various modes of terminal use (for example, the fill-in-the-blanks mode). However, there must be at least one language whose statements are expressible, per some well-defined syntax, as character strings and that is comprehensive in supporting all the following items

- Data Definition
- View Definition
- Data Manipulation (Interactive and by program).
- Integrity Constraints
- Authorization.
- Transaction boundaries (Begin , commit and rollback)

Rule 6: .View updating Rule
"All views that are theoretically updatable are also updatable by the system."

Rule 7: High-level insert, update and delete.
"The capability of handling a base relation or a derived relation as a single operand applies not only to the retrieval of data but also to the insertion, update and deletion of data."

Rule 8: Physical data independence.

"Application programs and terminal activities remain logically unimpaired whenever any changes are made in either storage representations or access methods."

Rule 9: Logical data independence.
"Application programs and terminal activities remain logically unimpaired when information preserving changes of any kind that theoretically permit un-impairment are made to the base tables."

Rule 10: Integrity independence.
"Integrity constraints specific to a particular relational data base must be definable in the relational data sub-language and storable in the catalog, not in the application programs."

Rule11: Distribution independence.
"A relational DBMS has distribution independence."

Rule 12: Non-subversion Rule.

"If a relational system has a low-level (single-record-at-a-time) language, that low level cannot be used to subvert or bypass the integrity Rules and constraints expressed in the higher level relational language (multiple-records-at-a-time)."

Q. What is ETL process in Data warehousing? What are the different stages in "Data warehousing"?
Ans: ETL (Extraction, Transformation and Loading) are different stages in Data warehousing. Like when we do software development we follow different stages like requirement gathering, designing, coding and testing. In the similar fashion we have for data warehousing.

Extraction:-
In this process we extract data from the source. In actual scenarios data source can be in many forms EXCEL, ACCESS, Delimited text, CSV (Comma Separated Files) etc. So extraction process handle's the complexity of understanding the data source and loading it in a structure of data warehouse.

Transformation:-
This process can also be called as cleaning up process. It's not necessary that after the extraction process data is clean and valid. For instance all the financial figures have NULL values but you want it to be ZERO for better analysis. So you can have some kind of stored procedure which runs through all extracted records and sets the value to zero.

Loading:-
After transformation you are ready to load the information in to your final data warehouse database.
Q. Compare Data mining and Data Warehousing?
Ans: "Data Warehousing" is technical process where we are making our data centralized while "Data mining" is more of business activity which will analyze how good your business is doing or predict how it will do in the future coming times using the current data. As said before "Data Warehousing" is not a need for "Data mining". It's good if you are doing "Data mining" on a "Data Warehouse" rather than on an actual production database. "Data Warehousing" is essential when we want to consolidate data from different sources, so it's like a cleaner and matured data which sits in between the various data sources and brings then in to one format.

"Data Warehouses" are normally physical entities which are meant to improve accuracy of "Data mining" process. For example you have 10 companies sending data in different format, so you create one physical database for consolidating all the data from different company sources, while "Data mining" can be a physical model or logical model. You can create a database in "Data mining" which gives you reports of net sales for this year for all companies. This need not be a physical database as such but a simple query.

Q. What is Online Transaction Processing (OLTP)?
Ans: Online Transaction Processing or OLTP manages transaction based applications which can be used for data entry and easy retrieval processing of data. This processing makes like easier on simplicity and efficiency. It is faster, more accurate results and expenses with respect to OTLP.

Example – Bank Transactions on a daily basis.

Q. What are various DDL commands in SQL? Give brief description of their purposes.
Ans: Following are various DDL or Data Definition Language commands in SQL −
CREATE − it creates a new table, a view of a table, or other object in database.
ALTER − it modifies an existing database object, such as a table.
DROP − it deletes an entire table, a view of a table or other object in the database.

Q. What are DDL statements?
Ans: DDL stands for Data definition Language. They change structure of the database objects like table, index etc. Most important DDL statements are as shown below:-
=>CREATE TABLE - creates a new table in the database.
=>ALTER TABLE – changes table structure in database.
=>DROP TABLE - deletes a table from database
=> CREATE INDEX - creates an index

=> DROP INDEX - deletes an index

Q. What are DML statements?
Ans: DML stands for Data Manipulation Statements. They update data values in table. Below are the most important DDL statements:-
=>SELECT - gets data from a database table
=> UPDATE - updates data in a table
=> DELETE - deletes data from a database table
=> INSERT INTO - inserts new data into a database table

Q. What are various DML commands in SQL? Give brief description of their purposes.
Ans: Following are various DML or Data Manipulation Language commands in SQL −
SELECT − it retrieves certain records from one or more tables.
INSERT − it creates a record.

UPDATE − it modifies records.
DELETE − it deletes records.
Q. What is the purpose of DML statements in SQL?
Ans: The DML statements are used to add new rows to a table
update or modify data in existing rows, or remove existing rows from a table.

Q. What are various DCL commands in SQL? Give brief description of their purposes.
Ans: Following are various DCL or Data Control Language commands in SQL −
GRANT − it gives a privilege to user.
REVOKE − it takes back privileges granted from user.

Q. What's the difference between DELETE and TRUNCATE?
Ans: Following are difference between them:
=>>DELETE TABLE syntax logs the deletes thus making the delete operations low. TRUNCATE table does not log any n formation but it logs information about deal location of data page of the table. So TRUNCATE table is faster as compared to delete table.
=>>DELETE table can have criteria while TRUNCATE cannot.
=>> TRUNCATE table cannot have triggers.

Q. What is the difference between TRUNCATE and DROP statements?
Ans: TRUNCATE removes all the rows from the table, and it cannot be rolled back. DROP command removes a table from the database and operation cannot be rolled back.

Q. What are tables and Fields?
Ans: A table is a set of data that are organized in a model with Columns and Rows. Columns can be categorized as vertical, and Rows are horizontal. A table has specified number of column called fields but can have any number of rows which is called record.

Example:
Table: Employee.
Field: Emp ID, Emp Name, Date of Birth.
Data: 201456, David, 11/15/1960.

Q. What is a primary key?
Ans: A primary key is a combination of fields which uniquely specify a row. This is a special kind of unique key, and it has implicit NOT NULL constraint. It means, Primary key values cannot be NULL.

Q. What is a unique key?
Ans: A Unique key constraint uniquely identified each record in the database. This provides uniqueness for the column or set of columns. A Primary key constraint has automatic unique constraint defined on it. But not in the case of Unique Key. There can be many unique constraint defined per table, but only one Primary key constraint defined per table.

Q. What is a foreign key?
Ans: foreign key is one table which can be related to the primary key of another table. Relationship needs to be created between two tables by referencing foreign key with the primary key of another table.

Q. What is an Index?
Ans: An index is performance tuning method of allowing faster retrieval of records from the table. An index creates an entry for each value and it will be faster to retrieve data.

Q. What are all the different types of indexes?
Ans: There are three types of indexes:

Unique Index: This indexing does not allow the field to have duplicate values if the column is unique indexed. Unique index can be applied automatically when primary key is defined.

Clustered Index: Clustered Index is a special type of index that reorders the way records in the table are physically stored. Therefore table can have only one clustered index. The leaf nodes of a clustered index contain the data pages. This type of index reorders the physical order of the table and search based on the key values. Each table can have only one clustered index.

NonClustered Index: NonClustered Index is a special type of index in which the logical order of the index does not match the physical stored order of the rows on disk. The leaf node of a non clustered index does not consist of the data pages. Instead, the leaf nodes contain index rows. NonClustered Index does not alter the physical order of the table and maintains logical order of data. Each table can have 999 nonclustered indexes.

Q. What are the different index configurations a table can have?
Ans: A table can have one of the following index configurations:

1. No indexes
2. A clustered index
3. A clustered index and many nonclustered indexes
4. A nonclustered index
5. Many nonclustered indexes

Q. What is a Cursor?
Ans: A database Cursor is a control which enables traversal over the rows or records in the table. This can be viewed as a pointer to one row in a set of rows. Cursor is very much useful for traversing such as retrieval, addition and removal of database records.

Q. What is a relationship and what are they?
Ans: Database Relationship is defined as the connection between the tables in a database. There are various data basing relationships, and they are as follows:

1. One-to-One Relationship.
2. One-to-Many Relationship.
3. Many-to-One Relationship.
4. Self-Referencing Relationship.

Q. What are local and global variables and their differences?
Ans: Local variables are the variables which can be used or exist inside the function. They are not known to the other functions and those variables cannot be referred or used. Variables can be created whenever that function is called.

Global variables are the variables which can be used or exist throughout the program. Same variable declared in global cannot be used in functions. Global variables cannot be created whenever that function is called.

Q. What is data Integrity?
Ans: Data Integrity defines the accuracy and consistency of data stored in a database. It can also define integrity constraints to enforce business rules on the data when it is entered into the application or database.

Q. What is a constraint?
Ans: Constraint can be used to specify the limit on the data type of table. Constraint can be specified while creating or altering the table statement.
Sample of constraint are:

1. NOT NULL
2. CHECK
3. DEFAULT
4. UNIQUE
5. PRIMARY KEY
6. FOREIGN KEY

Q. What is Auto Increment?
Ans: Auto increment keyword allows the user to create a unique number to be generated when a new record is inserted into the table. AUTO INCREMENT keyword can be used in Oracle and IDENTITY keyword can be used in SQL SERVER. Mostly this keyword can be used whenever PRIMARY KEY is used.

Q. What is a stored procedure?
Ans: Stored Procedure is a function consisting of many SQL statements to access the database system. Several SQL statements are consolidated into a stored procedure and execute them whenever and wherever required.

Q. Advantages and Disadvantages of Stored Procedure?
Ans: Stored procedure can be used as a modular programming – means create once, store and call for several times whenever required. This supports faster execution instead of executing multiple queries. This reduces network traffic and provides better security to the data. Disadvantage is that it can be executed only in the Database and utilizes more memory in the database server.

Q. What is recursive stored procedure?
Ans: A stored procedure which calls by itself until it reaches some boundary condition. This recursive function or procedure helps programmers to use the same set of code any number of times.

Q. What are the advantages of using Stored Procedures?
Ans:

1. Stored procedure can reduced network traffic and latency, boosting application performance.
2. Stored procedure execution plans can be reused, staying cached in SQL Server's memory, reducing server overhead.
3. Stored procedures help promote code reuse.
4. Stored procedures can encapsulate logic. You can change stored procedure code without affecting clients.
5. Stored procedures provide better security to your data.

Q. What is user defined functions?
Ans: User defined functions are the functions written to use that logic whenever required. It is not necessary to write the same logic several times. Instead, function can be called or executed whenever needed.

Q. What are all types of user defined functions?
Ans: Three types of user defined functions are:

- Scalar Functions
- Inline Table valued functions
- Multi statement valued functions

Scalar returns unit, variant defined the return clause. Other two types return table as a return.

Q. What is Difference between Function and Stored Procedure?
Ans: UDF can be used in the SQL statements anywhere in the WHERE/HAVING/SELECT section where as Stored procedures cannot be. UDFs that return tables can be treated as another rowset. This can be used in JOINs with other tables. Inline UDF's can be thought of as views that take parameters and can be used in JOINs and other Rowset operations.

Q. What is a trigger?
Ans: A DB trigger is a code or programs that automatically execute with response to some event on a table or view in a database. Mainly, trigger helps to maintain the integrity of the database. Example: When a new student is added to the student database, new records should be created in the related tables like Exam, Score and Attendance tables.

Q. What is Union, minus and Interact commands?
Ans: UNION operator is used to combine the results of two tables, and it eliminates duplicate rows from the tables. MINUS operator is used to return rows from the first query but not from the second query. Matching records of first and second query and other rows from the first query will be displayed as a result set.

INTERSECT operator is used to return rows returned by both the queries.

Q. What is CLAUSE?
Ans: SQL clause is defined to limit the result set by providing condition to the query. This usually filters some rows from the whole set of records.

Example – Query that has WHERE condition
Query that has HAVING condition

Q. What is an ALIAS command?
Ans: ALIAS name can be given to a table or column. This alias name can be referred in WHERE clause to identify the table or column.
Example: Select st.StudentID, Ex.Result from student st, Exam as Ex where st.studentID = Ex. StudentID

Here, st refers to alias name for student table and Ex refers to alias name for exam table.

Q. What are Aggregate and Scalar Functions?
Ans: Aggregate and Scalar functions are in built function for counting and calculations. Aggregate functions operate against a group of values but returns only one value.

AVG(column) :- Returns the average value of a column
COUNT(column) :- Returns the number of rows (without a NULL value) of a column
COUNT(*) :- Returns the number of selected rows

MAX(column) :- Returns the highest value of a column
MIN(column) :- Returns the lowest value of a column
Scalar functions operate against a single value and return value on basis of the single value.
UCASE(c) :- Converts a field to upper case
LCASE(c) :- Converts a field to lower case
MID(c,start[,end]) :- Extract characters from a text field
LEN(c) :- Returns the length of a text

Q. Name 3 ways to get an accurate count of the number of records in a table?
Ans: SELECT * FROM table1
SELECT COUNT(*) FROM table1
SELECT rows FROM sysindexes WHERE id = OBJECT_ID(table1) AND indid < 2

Q. How do we select distinct values from a table?
Ans: DISTINCT keyword is used to return only distinct values.
Below is syntax:-
Column age and Table pcdsEmp
SELECT DISTINCT age FROM pcdsEmp

Q. What is Like operator for and what are wild cards?
Ans: LIKE operator is used to match patterns. A "%" sign is used to define the pattern.
Below SQL statement will return all words with letter "S"
SELECT * FROM pcdsEmployee WHERE EmpName LIKE 'S%'
Below SQL statement will return all words which end with letter "S"
SELECT * FROM pcdsEmployee WHERE EmpName LIKE '%S'
Below SQL statement will return all words having letter "S" in between
SELECT * FROM pcdsEmployee WHERE EmpName LIKE '%S%'

"_" operator (we can read as “Underscore Operator”). “_” operator is the character defined atthat point. In the below sample fired a query
Select name from pcdsEmployee where name like
'_s%' So all name where second letter is “s” is returned.

Q. Can you explain Insert, Update and Delete query?
Ans: Insert statement is used to insert new rows in to table. Update to update existing data in the table. Delete statement to delete a record from the table. Below code snippet for Insert, Update and Delete :-
INSERT INTO pcdsEmployee SET name='rohit',age='24';
UPDATE pcdsEmployee SET age='25' where name='rohit';
DELETE FROM pcdsEmployee WHERE name = 'sonia';

Q. What is order by clause?
A. ORDER BY clause helps to sort the data in either ascending order to descending order.
Ascending order sort query
SELECT name, age FROM pcdsEmployee ORDER BY age ASC
Descending order sort query
SELECT name FROM pcdsEmployee ORDER BY age DESC

Q. What is the SQL "IN" clause?
A. SQL IN operator is used to see if the value exists in a group of values. For instance the below SQL checks if the Name is either 'rohit' or 'Anuradha'
SELECT * FROM pcdsEmployee WHERE

name IN ('Rohit','Anuradha') Also you can specify a not clause with the same. SELECT * FROM pcdsEmployee WHERE age NOT IN (17,16)

Q. Can you explain the between clause?
Ans: Below SQL selects employees born between '01/01/1975' AND '01/01/1978' as per mysql
SELECT * FROM pcdsEmployee WHERE DOB BETWEEN '1975-01-01' AND '2011-09-28'

Q. We have an employee salary table how do we find the second highest from it?
Ans: Below Sql Query find the second highest salary
SELECT * FROM pcdsEmployeeSalary a WHERE (2=(SELECT COUNT(DISTINCT(b.salary)) FROM pcdsEmployeeSalary b WHERE b.salary>=a.salary))

Q. What is Join and types of joins and explain each?
Ans: This is a keyword used to query data from more tables based on the relationship between the fields of the tables. Keys play a major role when JOINs are used.

There are various types of join which can be used to retrieve data and it depends on the relationship between tables.

Inner join: Inner join return rows when there is at least one match of rows between the tables.

Right Join: Right join return rows which are common between the tables and all rows of Right hand side table. Simply, it returns all the rows from the right hand side table even though there are no matches in the left hand side table.

Left Join: Left join return rows which are common between the tables and all rows of Left hand side table. Simply, it returns all the rows from Left hand side table even though there are no matches in the Right hand side table.

Full Join: Full join return rows when there are matching rows in any one of the tables. This means, it returns all the rows from the left hand side table and all the rows from the right hand side table.
Q. What is “CROSS JOIN”? or What is Cartesian product?
Ans: “CROSS JOIN” or “CARTESIAN PRODUCT” combines all rows from both tables. Number of rows will be product of the number of rows in each table. In real life scenario I cannot imagine where we will want to use a Cartesian product. But there are scenarios where we would like permutation and combination probably Cartesian would be the easiest way to achieve it.

Q. What is the difference between cross joins and natural joins?
Ans: The cross join produces the cross product or Cartesian product of two tables. The natural join is based on all the columns having same name and data types in both the tables.

Q. How to select the first record in a given set of rows?
Ans: Select top 1 * from sales.salesperson

Q. What is the default “-SORT” order for SQL?
Ans: ASCENDING

Q. What is Self-Join?
A.Self-join is set to be query used to compare to itself. This is used to compare values in a column with other values in the same column in the same table. ALIAS ES can be used for the same table comparison.

Q. What is the difference between “UNION” and “UNION ALL”?
Ans: UNION SQL syntax is used to select information from two tables. But it selects only distinct records from both the table, while UNION ALL selects all records from both the tables.

Q. What are cursors and what are the situations you will use them?
Ans: SQL statements are good for set at a time operation. So it is good at handling set of data. But there are scenarios where we want to update row depending on certain criteria. We will loop through all rows and update data accordingly. There's where cursors come in to picture.

Q. What is "Group by" clause?
Ans: "Group by" clause group similar data so that aggregate values can be derived.

Q. What is the difference between "HAVING" and "WHERE" clause?
Ans: "HAVING" clause is used to specify filtering criteria for "GROUP BY", while "WHERE" clauseapplies on normal SQL.

Q. What is a query?
Ans: A DB query is a code written in order to get the information back from the database. Query can be designed in such a way that it matched with our expectation of the result set. Simply, a question to the Database.

Q. What is subquery & Types of subqueries?
Ans: A subquery is a query within another query. The outer query is called as main query, and inner query is called subquery. SubQuery is always executed first, and the result of subquery is passed on to the main query.

There are two types of subquery – **Correlated and Non-Correlated**. A correlated subquery cannot be considered as independent query, but it can refer the column in a table listed in the FROM the list of the main query. A Non-Correlated sub query can be considered as independent query and the output of subquery are substituted in the main query.

For example, to retrieve all EmployeeID and CustomerID records from the ORDERS table that have the EmployeeID greater than the average of the EmployeeID field, you can create a nested query, as shown:

```
SELECT DISTINCT EmployeeID, CustomerID FROM ORDERS WHERE EmployeeID> (SELECTAVG(EmployeeID) FROM ORDERS)
```

Q. What are the properties and different Types of Sub-Queries?
Ans: Properties of Sub-Query:

1. A sub-query must be enclosed in the parenthesis.
2. A sub-query must be put in the right hand of the comparison operator, and
3. A sub-query cannot contain an ORDER-BY clause.

B). A query can contain more than one sub-query.

1. Types of Sub-Query
2. Single-row sub-query, where the sub-query returns only one row.
3. Multiple-row sub-query, where the sub-query returns multiple rows,. and
4. Multiple column sub-query, where the sub-query returns multiple columns.

Q. What do you understand by a subquery? When is it used?
Ans: A subquery is a SELECT statement embedded in a clause of another SELECT statement. It is used when the inner query, or the subquery, returns a value that is used by the outer query. It is very useful in selecting some rows in a table with a condition that depends on some data which is contained in the same table.

Q. Can you explain the SELECT INTO Statement?
Ans: SELECT INTO statement is used mostly to create backups. The below SQL statement backs up the Employee table in to the EmployeeBackUp table. One point to be noted is that the structure of pcdsEmployeeBackup and pcdsEmployee table should be same.

```
SELECT * INTO pcdsEmployeeBackup FROM pcdsEmployee
```

Q. What is a View?
Ans: A view is a virtual table which consists of a subset of data contained in a table. Views are not virtually present, and it takes less space to store. View can have data of one or more tables combined, and it is depending on the relationship.

```
CREATE VIEW [MyView] AS SELECT * from pcdsEmployee where LastName = 'singh'
```

In order to query the view

```
SELECT * FROM [MyView]
```

Q. Why should you use a view?
Ans: A view is used for –
Restricting access to data;
Making complex queries simple;
Ensuring data independency;
Providing different views of same data.

Q. What is SQL injection?
Ans: It is a Form of attack on a database-driven Web site in which the attacker executes unauthorized SQL commands by taking advantage of insecure code on a system connected to the Internet, bypassing the firewall. SQL injection attacks are used to steal information from a database from which the data would normally not be available and/or to gain access to an organization's host computers through the computer that is hosting the database.
SQL injection attacks typically are easy to avoid by ensuring that a system has strong input validation. As name suggest we inject SQL which can be relatively dangerous for the database.

Example: this is a simple SQL

```
SELECT email, passwd, login_id, full_name
FROM members WHERE email = 'x'
```

Now somebody does not put "x" as the input but puts "x ; DROP TABLE members;".
So the actual SQL which will execute is :-

```
SELECT email, passwd, login_id, full_name FROM members WHERE email = 'x' ; DROP TABLE
members;
```

Think what will happen to your database.

Q. How to fetch common records from two tables?
Ans: Common records result set can be achieved by -.

```
Select studentID from student.
<strong>INTERSECT </strong>
Select StudentID from Exam.
```

Q. How to fetch alternate records from a table?
Ans: Records can be fetched for both Odd and Even row numbers -.
To display even numbers-.

```
Select studentId from (Select rowno, studentId from student) where mod(rowno,2)=0.
```

To display odd numbers-.

```
Select studentId from (Select rowno, studentId from student) where mod(rowno,2)=1.
```

Q. How to select unique records from a table?
Ans: Select unique records from a table by using DISTINCT keyword.

```
 Select DISTINCT StudentID, StudentName from Student.
```

Q. What is the command used to fetch first 5 characters of the string?

Ans: There are many ways to fetch first 5 characters of the string -.
Select SUBSTRING(StudentName,1,5) as studentname from student.
Select RIGHT(Studentname,5) as studentname from student.

Q. Which operator is used in query for pattern matching?
Ans: LIKE operator is used for pattern matching, and it can be used as –
% – Matches zero or more characters.
_(Underscore) – Matching exactly one character.
Example -
Select * from Student where studentname like 'a%'

Q. Can you sort a column using a column alias?
Ans: Yes. A column alias could be used in the ORDER BY clause.

Q. Is a NULL value same as zero or a blank space? If not then what is the difference?
Ans: A NULL value is not same as zero or a blank space. A NULL value is a value which is 'unavailable, unassigned, unknown or not applicable'. Whereas, zero is a number and blank space is a character.

Q. Say True or False. Give explanation if False. If a column value taking part in an arithmetic expression is NULL, then the result obtained would be NULLM.
Ans: True.

Q. If a table contains duplicate rows, does a query result display the duplicate values by default? How can you eliminate duplicate rows from a query result?
Ans: A query result displays all rows including the duplicate rows. To eliminate duplicate rows in the result, the DISTINCT keyword is used in the SELECT clause.

Q. What is the purpose of the condition operators BETWEEN and IN?
Ans: The BETWEEN operator displays rows based on a range of values. The IN condition operator checks for values contained in a specific set of values.

Q. How do you search for a value in a database table when you don't have the exact value to search for?
Ans: In such cases, the LIKE condition operator is used to select rows that match a character pattern. This is also called 'wildcard' search.

Q. What is the default ordering of data using the ORDER BY clause? How could it be changed?
Ans: The default sorting order is ascending. It can be changed using the DESC keyword, after the column name in the ORDER BY clause.

Q. What are the case manipulation functions of SQL?
Ans: LOWER, UPPER, INITCAP

Q. Which function returns the remainder in a division operation?
Ans: The MOD function returns the remainder in a division operation.

Q. What is the purpose of the NVL function?
Ans: The NVL function converts a NULL value to an actual value.

Q. What is the difference between the NVL and the NVL2 functions?
Ans: The NVL(exp1, exp2) function converts the source expression (or value) exp1 to the target expression (or value) exp2, if exp1 contains NULL. The return value has the same data type as that of exp1.

The NVL2(exp1, exp2, exp3) function checks the first expression exp1, if it is not null then, the second expression exp2 is returned. If the first expression exp1 is null, then the third expression exp3 is returned.

Q. What is the use of the NULLIF function?
Ans: The NULLIF function compares two expressions. If they are equal, the function returns null. If they are not equal, the first expression is returned.

Q. Discuss the syntax and use of the COALESCE function?
Ans: The COALESCE function has the expression COALESCE(exp1, exp2, …. Expn)

1. It returns the first non-null expression given in the parameter list.

Q. Which expressions or functions allow you to implement conditional processing in a SQL statement?
Ans: There are two ways to implement conditional processing or IF-THEN-ELSE logic in a SQL statement.
Using CASE expression

Using the DECODE function

- You want to display a result query from joining two tables with 20 and 10 rows respectively. Erroneously you forget to write the WHERE clause. What would be the result?
- The result would be the Cartesian product of two tables with 20 x 10 = 200 rows.

Q. You want to display a result query from joining two tables with 20 and 10 rows respectively. Erroneously you forget to write the WHERE clause. What would be the result?
Ans: The result would be the Cartesian product of two tables with 20 x 10 = 200 rows.

Q. What is the purpose of the MERGE statement in SQL?
Ans: The MERGE statement allows conditional update or insertion of data into a database table. It performs an UPDATE if the row exists, or an INSERT if the row does not exist.

Q. What is the difference between VARCHAR2 AND CHAR data types?
Ans: VARCHAR2 represents variable length character data, whereas CHAR represents fixed length character data.

Q. Say True or False. Give explanation if False.
Ans: A DROP TABLE statement can be rolled back.
Ans: False. A DROP TABLE statement cannot be rolled back.

Q. Which SQL statement is used to add, modify or drop columns in a database table?
Ans: The ALTER TABLE statement.

Q. What's wrong in the following query?
Ans: SELECT student_code, name
FROM students
WHERE marks =
(SELECT MAX(marks)
FROM students   GROUP BY subject_code);

Q. Say True or False. Give explanation if False.
Ans: By the default the group functions consider only distinct values in the set. By default, group functions consider all values including the duplicate values.

Q. Say True or False. Give explanation if False.
Ans: The DISTINCT keyword allows a function consider only non-duplicate values.

A. True.

Q. Say True or False. Give explanation if False.
All group functions ignore null values.
Ans: True.

Q. Say True or False. Give explanation if False.
COUNT(*) returns the number of columns in a table.
Ans: False. COUNT(*) returns the number of rows in a table.

Q. What's wrong in the following query?

```
SELECT subject_code, count(name)
FROM students;
```

Ans: It doesn't have a GROUP BY clause. The subject_code should be in the GROUP BY clause.

```
SELECT subject_code, count(name)
FROM students
GROUP BY subject_code;
```

Q. What's wrong in the following query?

```
SELECT subject_code, AVG (marks)
FROM students
WHERE AVG(marks) > 75
GROUP BY subject_code;
```

Ans: The WHERE clause cannot be used to restrict groups. The HAVING clause should be used.

```
SELECT subject_code, AVG (marks)
FROM students
HAVING AVG(marks) > 75
GROUP BY subject_code;
```

Q. Say True or False. Give explanation if False.
Group functions cannot be nested.
Ans: False. Group functions can be nested to a depth of two.

Q. Say True or False. Give explanation if False.
A single row subquery returns only one row from the outer SELECT statement
Ans: False. A single row subquery returns only one row from the inner SELECT statement.

Q. Say True or False. Give explanation if False.
A multiple row subquery returns more than one row from the inner SELECT statement.
Ans: True.

Q. Say True or False. Give explanation if False.
Multiple column subqueries return more than one column from the inner SELECT statement.
Ans: True.

Q. What are the various multiple row comparison operators in SQL?
Ans: IN, ANY, ALL.

Q. Which statement is used to add a new row in a database table?

Ans: The INSERT INTO statement.

Q. Say True or False. Give explanation if False.
While inserting new rows in a table you must list values in the default order of the columns.
Ans: True.

Q. How do you insert null values in a column while inserting data?
Ans: Null values can be inserted into a table by one of the following ways −
Implicitly by omitting the column from the column list.
Explicitly by specifying the NULL keyword in the VALUES clause.

Q. Say True or False. Give explanation if False.
INSERT statement does not allow copying rows from one table to another.
Ans: False. INSERT statement allows adding rows to a table copying rows from an existing table.

Q. How do you copy rows from one table to another?
Ans: The INSERT statement can be used to add rows to a table by copying from another table. In this case, a subquery is used in the place of the VALUES clause.

Q. What happens if you omit the WHERE clause in the UPDATE statement?
Ans: All the rows in the table are modified.

Q. Can you modify the rows in a table based on values from another table? Explain.
Ans: Yes. Use of subqueries in UPDATE statements allow you to update rows in a table based on values from another table.

Q. Say True or False. Give explanation if False.
The DELETE statement is used to delete a table from the database.
Ans: False. The DELETE statement is used for removing existing rows from a table.

Q. What happens if you omit the WHERE clause in a delete statement?
Ans: All the rows in the table are deleted.

Q. Can you remove rows from a table based on values from another table? Explain.
Ans: Yes, subqueries can be used to remove rows from a table based on values from another table.

Q. Say True or False. Give explanation if False.
Attempting to delete a record with a value attached to an integrity constraint, returns an error.
Ans: True.

Q. Say True or False. Give explanation if False.
You can use a subquery in an INSERT statement.
Ans: True.
Q. The condition in a WHERE clause can refer to only one value.
A. True
B. False
Answer: Option B

Q. The ADD command is used to enter one row of data or to add multiple rows as a result of a query.
A. True
B. False

Answer: Option B

Q. SQL provides the AS keyword, which can be used to assign meaningful column names to the results of queries using the SQL built-in functions.
A. True
B. False
Answer: Option A

Q. The SELECT command, with its various clauses, allows users to query the data contained in the tables and ask many different questions or ad hoc queries.
A. True
B. False
Answer: Option A

Q. A SELECT statement within another SELECT statement and enclosed in square brackets ([...]) is called a subquery.
A. True
B. False
Answer: Option B

Q. The rows of the result relation produced by a SELECT statement can be sorted, but only by one column.
A. True
B. False
Answer: Option B

Q. There is an equivalent join expression that can be substituted for all subquery expressions.
A. True
B. False
Answer: Option B

Q. A dynamic view is one whose contents materialize when referenced.
A. True
B. False
Answer: Option A

Q. SQL is a programming language.
A. True
B. False
Answer: Option B

Q. SELECT DISTINCT is used if a user wishes to see duplicate columns in a query.
A. True
B. False
Answer: Option B
Q. Indexes can usually be created for both primary and secondary keys.
A. True
B. False
Answer: Option A

Q. Each index consumes extra storage space and also requires overhead maintenance time whenever indexed data change value.
A. True

B. False
Answer: Option A

Q. The HAVING clause acts like a WHERE clause, but it identifies groups that meet a criterion, rather than rows.
A. True
B. False
Answer: Option A

Q. SQL is a data sublanguage.
A. True
B. False
Answer: Option A

Q. The qualifier DISTINCT must be used in an SQL statement when we want to eliminate duplicate rows.
A. True
B. False
Answer: Option A

Q. DISTINCT and its counterpart, ALL, can be used more than once in a SELECT statement.
A. True
B. False
Answer: Option B

Q. The result of every SQL query is a table.
A. True
B. False
Answer: Option A

Q. COUNT(field_name) tallies only those rows that contain a value; it ignores all null values.
A. True
B. False
Answer: Option A

Q. SUM, AVG, MIN, and MAX can only be used with numeric columns.
A. True
B. False
Answer: Option B

Q. Most companies keep at least two versions of any database they are using.
A. True
B. False
Answer: Option A

Q. The format SELECT-FROM-WHERE is the fundamental framework of SQL SELECT statements.
A. True
B. False
Answer: Option A

Q. Indexes may be created or dropped at any time.
A. True
B. False

Answer: Option A

Q. The SQL statement: SELECT Number1 + Number 2 AS Total FROM NUMBER_TABLE; adds two numbers from each row together and lists the results in a column named Total.

A. True

B. False

Answer: Option A

Q. ORDER BY can be combined with the SELECT statements.

A. True

B. False

Answer: Option A

Q. Data manipulation language (DML) commands are used to define a database, including creating, altering, and dropping tables and establishing constraints.

A. True

B. False

Answer: Option B

Q. Scalar aggregate are multiple values returned from an SQL query that includes an aggregate function.

A. True

B. False

Answer: Option B

Q. The keyword LIKE can be used in a WHERE clause to refer to a range of values.

A. True

B. False

Answer: Option B

Q. The SQL statement: SELECT Name, COUNT(*) FROM NAME_TABLE; counts the number of name rows and displays this total in a table with a single row and a single column.

A. True

B. False

Answer: Option B

Q. The SQL keyword GROUP BY instructs the DBMS to group together those rows that have the same value in a column.

A. True

B. False

Answer: Option A

Q. The wildcard asterisk (*) is the SQL-92 standard for indicating "any sequence of characters."

A. True

B. False

Answer: Option B

Q. Microsoft Access has become ubiquitous, and being able to program in Access is a critical skill.

A. True

B. False

Answer: Option B

Q. SQL provides five built-in functions: COUNT, SUM, AVG, MAX, MIN.

A. True
B. False
Answer: Option A

Q. The keyword BETWEEN can be used in a WHERE clause to refer to a range of values.
A. True
B. False
Answer: Option A

Q. To establish a range of values, < and > can be used.
A. True
B. False
Answer: Option A

Q. If you are going to use a combination of three or more AND and OR conditions, it is often easier to use the NOT and NOT IN operators.
A. True
B. False
Answer: Option A

Q. Say True or False. Give explanation if False.
A DDL statement or a DCL statement is automatically committed.
Answer. True.

Q. You can add a row using SQL in a database with which of the following?
A. ADD
B. CREATE
C. INSERT
D. MAKE
Answer: Option C

Q. The command to remove rows from a table 'CUSTOMER' is:
A.REMOVE FROM CUSTOMER ...
B.DROP FROM CUSTOMER ...
C.DELETE FROM CUSTOMER WHERE ...
D.UPDATE FROM CUSTOMER ...
Answer: Option C

Q. The SQL WHERE clause:
A.limits the column data that are returned.
B.limits the row data are returned.
C.Both A and B are correct.
D.Neither A nor B are correct.
Answer: Option B

Q. Which of the following is the original purpose of SQL?
A.To specify the syntax and semantics of SQL data definition language
B.To specify the syntax and semantics of SQL manipulation language
C.To define the data structures
D.All of the above.
Answer: Option D

Q. The wildcard in a WHERE clause is useful when?
A.An exact match is necessary in a SELECT statement.
B.An exact match is not possible in a SELECT statement.
C.An exact match is necessary in a CREATE statement.
D.An exact match is not possible in a CREATE statement.
Answer: Option B

Q. A view is which of the following?
A.A virtual table that can be accessed via SQL commands
B.A virtual table that cannot be accessed via SQL commands
C.A base table that can be accessed via SQL commands
D.A base table that cannot be accessed via SQL commands
Answer: Option A

Q. The command to eliminate a table from a database is:
A.REMOVE TABLE CUSTOMER;
B.DROP TABLE CUSTOMER;
C.DELETE TABLE CUSTOMER;
D.UPDATE TABLE CUSTOMER;
Answer: Option B

Q. ON UPDATE CASCADE ensures which of the following?
A.Normalization
B.Data Integrity
C.Materialized Views
D.All of the above.
Answer: Option B

Q. SQL data definition commands make up a(n) ________ .
A. DDL
B. DML
C. HTML
D. XML
Answer: Option A

Q. Which of the following is valid SQL for an Index?
A.CREATE INDEX ID;
B.CHANGE INDEX ID;
C.ADD INDEX ID;
D.REMOVE INDEX ID;
Answer: Option A

Q. The SQL keyword(s) ________ is used with wildcards.
A.LIKE only
B. IN only
C.NOT IN only
D.IN and NOT IN
Answer: Option A

Q. Which of the following is the correct order of keywords for SQL SELECT statements?

A.SELECT, FROM, WHERE
B.FROM, WHERE, SELECT
C.WHERE, FROM,SELECT
D.SELECT,WHERE,FROM
Answer: Option A

Q. A subquery in an SQL SELECT statement is enclosed in:
A.braces -- {...}.
B.CAPITAL LETTERS.
C.parenthesis -- (...) .
D.brackets -- [...].
Answer: Option C

Q. The result of a SQL SELECT statement is a(n) ________ .
A. report
B. form
C. file
D. table
Answer: Option D

Q. Which of the following are the five built-in functions provided by SQL?
A.COUNT, SUM, AVG, MAX, MIN
B.SUM, AVG, MIN, MAX, MULT
C.SUM, AVG, MULT, DIV, MIN
D.SUM, AVG, MIN, MAX, NAME
Answer: Option A

Q. In an SQL SELECT statement querying a single table, according to the SQL-92 standard the asterisk (*) means that:
A. all columns of the table are to be returned.
B. all records meeting the full criteria are to be returned.
C. all records with even partial criteria met are to be returned.
D.None of the above is correct.
Answer: Option A

Q. The HAVING clause does which of the following?
A. Acts like a WHERE clause but is used for groups rather than rows.
B. Acts like a WHERE clause but is used for rows rather than columns.
C. Acts like a WHERE clause but is used for columns rather than groups.
D. Acts EXACTLY like a WHERE clause.
Answer: Option A

Q. The SQL -92 wildcards are ____ and ____ .
A. asterisk (*); percent sign (%)
B. percent sign (%); underscore (_)
C. underscore(_); question mark (?)
D. question mark (?); asterisk (*)
Answer: Option B

Q. To remove duplicate rows from the results of an SQL SELECT statement, the ________ qualifier specified must be included.
A. ONLY

B. UNIQUE
C.DISTINCT
D. SINGLE
Answer: Option C

Q. The benefits of a standard relational language include which of the following?
A.Reduced training costs
B.Increased dependence on a single vendor
C.Applications are not needed.
D.All of the above.
Answer: Option A

Q. Which of the following do you need to consider when you make a table in SQL?
A.Data types
B.Primary keys
C.Default values
D.All of the above.
Answer: Option D

Q. SQL query and modification commands make up a(n) ________ .
A. DDL
B. DML
C. HTML
D. XML
Answer: Option B

Q. When three or more AND and OR conditions are combined, it is easier to use the SQL keyword(s):
A. LIKE only.
B. IN only.
C. NOT IN only.
D. Both IN and NOT IN.
Answer: Option D

Q. The Microsoft Access wildcards are ____ and ____ .
A. asterisk (*); percent sign (%)
B. percent sign (%); underscore (_)
C. underscore(_); question mark (?)
D. question mark (?); asterisk (*)
Answer: Option D

Q. Find the SQL statement below that is equal to the following: SELECT NAME FROM CUSTOMER WHERE STATE = 'VA';
A.SELECT NAME IN CUSTOMER WHERE STATE IN ('VA');
B.SELECT NAME IN CUSTOMER WHERE STATE = 'VA';
C.SELECT NAME IN CUSTOMER WHERE STATE = 'V';
D.SELECT NAME FROM CUSTOMER WHERE STATE IN ('VA');
Answer: Option D

Q. Which one of the following sorts rows in SQL?
A.SORT BY
B.ALIGN BY
C.ORDER BY

D.GROUP BY

Answer: Option C

Q. To sort the results of a query use:

A.SORT BY.

B.GROUP BY.

C.ORDER BY.

D.None of the above is correct.

Answer: Option C

Q. The SQL statement that queries or reads data from a table is ________ .

A. SELECT

B.READ

C. QUERY

D.None of the above is correct.

Answer: Option A

Q. Which one of the following sorts rows in SQL?

A.SORT BY

B.ALIGN BY

C.ORDER BY

D.GROUP BY

Answer: Option C

Q. To define what columns should be displayed in an SQL SELECT statement:

A. use FROM to name the source table(s) and list the columns to be shown after SELECT.

B. use USING to name the source table(s) and list the columns to be shown after SELECT.

C. use SELECT to name the source table(s) and list the columns to be shown after USING.

D. use USING to name the source table(s) and list the columns to be shown after WHERE.

Answer: Option A

Q. SQL can be used to:

A. create database structures only.

B. query database data only.

C. modify database data only.

D.All of the above can be done by SQL.

Answer: Option D

Q. The SQL keyword BETWEEN is used:

A.for ranges.

B.to limit the columns displayed.

C.as a wildcard.

D.None of the above is correct.

Answer: Option A

Q. A subquery in an SQL SELECT statement:

A.can only be used with two tables.

B.can always be duplicated by a join.

C.has a distinct form that cannot be duplicated by a join.

D.cannot have its results sorted using ORDER BY.

Answer: Option C

Q. ________ was adopted as a national standard by ANSI in 1992.

A. Oracle

B. SQL

C.Microsoft Access

D. DBase

Answer: Option B

Q. SQL is:

A.a programming language.

B.an operating system.

C.a data sublanguage.

D. a DBMS.

Answer: Option C