**RDBMS** stands for **Relational Database Management System**.

Machine generated alternative text:
Some of The Most Important SQL Commands 
• SELECT - extracts data from a database 
• UPDATE - updates data in a database 
• DELETE - deletes data from a database 
• INSERT INTO - inserts new data into a database 
• CREATE DATABASE - creates a new database 
• ALTER DATABASE - modifies a database 
• CREATE TABLE - creates a new table 
• ALTER TABLE - modifies a table 
• DROP TABLE - deletes a table 
• CREATE INDEX - creates an index (search key) 
• DROP INDEX - deletes an index 

1. T - sql language

Machine generated alternative text:
The four main categories of SQL statements are as follows: 
1. DML (Data Manipulation Language) 
2. DDL (Data Definition Language) 
3. DCL (Data Control Language) 
4. TCL (Transaction Control Language) 
SQL Language Statements 
DML 
SELECT 
INSERT 
UPDATE 
DELETE 
DDL 
CREATE 
ALTER 
DROP 
DCL 
GRANT 
REVOKE 
TCL 
BEGIN 
TRAN 
COMMIT 
TRAN 
ROLLBACK 
tech-recipes.com 

Machine generated alternative text:
DCL (Data Control Language) 
DCL statements control the level of access that users have on database objects. 
GRANT — allows users to read/write on certain database objects 
REVOKE — keeps users from read/write permission on database objects 
TCL (Transaction Control Language) 
TCL statements allow you to control and manage transactions to maintain the integrity of data within 
SQL statements. 
BEGIN Transaction — opens a transaction 
COMMIT Transaction — commits a transaction 
ROLLBACK Transaction — ROLLBACK a transaction in case of any error 

1. Normalization(范式)

Machine generated alternative text:
Normalization is used for mainly two purpose, 
• Eliminating reduntant(useless) data. 
• Ensuring data dependencies make sense i.e data is logically stored. 

Machine generated alternative text:
Here are the most commonly used normal forms: 
First normal form(1NF) 
Second normal form(2NF) 
Third normal form(3NF) 
Boyce & Codd normal form (BCNF) 

Machine generated alternative text:
First normal form (INF) 
As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values. It 
should hold only atomic values. 

Machine generated alternative text:
Second normal form (2NF) 
A table is said to be in 2NF if both the following conditions hold: 
• Table is in INF (First normal form) 
• No non-prime attribute is dependent on the proper subset of any candidate key of table. 
An attribute that is not part of any candidate key is known as non-prime attribute. 

Machine generated alternative text:
Third Normal form (3NF) 
A table design is said to be in 3NF if both the following conditions hold: 
• Table must be in 2NF 
Transitive functional dependency of non-prime attribute on any super key should be removed. 
An attribute that is not part of any candidate key is known as non-prime attribute. 
In other words 3NF can be explained like this: A table is in 3NF if it is in 2NF and for each functional 
dependency X-> Y at least one of the following conditions hold: 
IS a super key of table 
Y is a prime attribute of table 
An attribute that is a part of one of the candidate keys is known as prime attribute. 

1. Drop or Truncate Table

DROP TABLE *table\_name*;

TRUNCATE TABLE *table\_name*;

DELETE FROM *table\_name* WHERE *condition*;

**Truncate** table is used to delete the data inside a table, but not the table itself

**Drop** table is used to delete a table include the data and table itself structure

**Delete** table is used to delete existing records in a table, if there isn't where clause, all records will be deleted like Truncate

1. SP VS Function

**Stored Procedures** are **pre-compile objects** which are compiled for first time and its compiled format is saved which executes (compiled code) whenever it is called.

But **Function** is **compiled** and executed every time when it is called

Machine generated alternative text:
Basic Difference 
01. Function must return a value but in Stored Procedure it is optional( Procedure can return zero or n 
values). 
02. Functions can have only input parameters for it whereas Procedures can have input/output 
parameters . 
03. Functions can be called from Procedure whereas Procedures cannot be called from Function. 

Machine generated alternative text:
Advance Difference 
01. Procedure allows SELECT as well as DML(INSERT/UPDATE/DELETE) statement in it whereas Function 
allows only SELECT statement in it. 
02. Procedures can not be utilized in a SELECT statement whereas Function can be embedded in a 
SELECT statement. 
03. Stored Procedures cannot be used in the SQL statements anywhere in the WHERE/HAVING/SELECT 
section whereas Function can be. 
04. Functions that return tables can be treated as another rowset. This can be used in JOINs with other 
tables. 
05. Inline Function can be though of as views that take parameters and can be used in JOINs and other 
Rowset operations. 

Machine generated alternative text:
06. Exception can be handled by try-catch block in a Procedure whereas try-catch block cannot be used 
in a Function. 
07. We can go for Transaction Management in Procedure whereas we can't go in Function. 

1. Index

Indexes are special lookup tables that the database search engine can **use to speed up data retrieval**. Simply put, an index is a pointer to

data in a table. **An index in a database is very similar to an index in the back of a book**.

An index helps to speed up SELECT queries and WHERE clauses, but **it slows down data input, with the UPDATE and the INSERT statements**. Indexes can be created or dropped with no effect on the data.

Machine generated alternative text:
Single-Column Indexes 
A single-column index is created based on only one table column. The basic 
syntax is as follows. 
CREATE INDEX index name 
ON table name (column _ name); 
Unique Indexes 
Unique indexes are used not only for performance, but also for data integrity. 
A unique index does not allow any duplicate values to be inserted into the 
table. The basic syntax is as follows. 
CREATE UNIQUE INDEX index_name 
on table name (column _ name); 
Composite Indexes 
A composite index is an index on two or more columns of a table. 
syntax is as follows. 
CREATE INDEX index name 
on table name (columnl, column2); 
Its basic 
Whether to create a single-column index or a composite index, take into 
consideration the column(s) that you may use very frequently in a query's 
WHERE clause as filter conditions. 

Machine generated alternative text:
When should indexes be avoided? 
Although indexes are intended to enhance a database's performance, there 
are times when they should be avoided. 
The following guidelines indicate when the use of an index should be 
reconsidered. 
Indexes should not be used on small tables. 
Tables that have frequent, large batch updates or insert operations. 
Indexes should not be used on columns that contain a high number of 
NULL values. 
Columns that are frequently manipulated should not be indexed. 

1. View

a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database

A view always shows **up-to-date data**

Machine generated alternative text:
The view "Current Product List" lists all active products (products that are not discontinued) from the 
"Products" table. The view is created with the following SQL: 
CREATE VIEW [Current Product List] AS 
SELECT Product1D, ProductName 
FROM Products 
WHERE Discontinued 
- No; 
Then, we can query the view as follows: 
SELECT * FROM [Current Product List], 

1. Transaction

Machine generated alternative text:
Transactions have the following four standard properties, usually referred to 
by the acronym ACID. 
Atomicity — ensures that all operations within the work unit are 
completed successfully. Otherwise, the transaction is aborted at the 
point of failure and all the previous operations are rolled back to their 
former state. 
Consistency — ensures that the database properly changes states 
upon a successfully committed transaction. 
Isolation 
enables transactions to operate independently of and 
a 
transparent to each other. 
Durability — ensures that the result or effect of a committed 
transaction persists in case of a system failure. 
Transaction Control 
The following commands are used to control transactions. 
COMMIT — to save the changes. 
ROLLBACK — to roll back the changes. 
SAVEPOINT — creates points within the groups of transactions in 
which to ROLLBACK. 
SET TRANSACTION - Places a name on a transaction. 
a 

1. Join

Machine generated alternative text:
Different Types of SQL JOINs 
Here are the different types of the JOINs in SQL: 
• (INNER) JOIN: Returns records that have matching values in both tables 
• LEFT (OUTER) JOIN: Return all records from the left table, and the matched records from the right 
table 
• RIGHT (OUTER) JOIN: Return all records from the right table, and the matched records from the left 
table 
• FULL (OUTER) JOIN: Return all records when there is a match in either left or right table 
INNER JOIN 
LEFT JOIN 
RIGHT JOIN 
FULL OUTER JOIN 
tablel 
table2 

1. Union and Union All

Machine generated alternative text:
The SQL UNION Operator 
The UNION operator is used to combine the result-set of two or more SELECT statements. 
• Each SELECT statement within UNION must have the same number of columns 
• The columns must also have similar data types 
• The columns in each SELECT statement must also be in the same order 
UNION syntax 
SELECT FROM tablel 
UNION 
SELECT FROM table2; 

Machine generated alternative text:
UNION ALL syntax 
The UNION operator selects only distinct values by default. To allow duplicate values, use UNION ALL: 
SELECT FROM tablel 
UNION ALL 
SELECT FROM table2; 
Note: The column names in the result-set are usually equal to the column names in the first SELECT 
statement in the UNION. 