NORMAL DATABASE (OLTP)	DATA WAREHOUSE (OLAP)
OLTP (Online Transaction Processing)	OLAP (Online Analysis Process)
• It records the user current transaction data	• It maintain the historical data
• Tables and Joins are complex since they are normalized. (Avoid duplicates)	• Tables and Joins are simple since they are de-normalized.
• Optimised for keeping small amount of data	• Optimised for keeping large amount of data
• MB to GB	• GB to TB
• Small to Large	• Large to Very Large
• Entity relationship Modelling technique is used	• Dimension modelling technique is used
• Analyse the query time is less sub second to second	• Analyse the query time is more seconds to minutes
• Optimized (used) for write operations like insert, update, Delete	• Primarily Read only operation
• Application oriented	• Subject oriented
• Only single stream	• Integrity
• Volatile	• Non-volatile
• Handling single record at a time	• Handling multiple records at a time

DIMENSION TABLE	FACT TABLE
• It contains details descriptive about the business	• It contains measures about the business
• It is having primary key	• It is having foreign key
• It contains textual data	• It contains measures in numeric
• In OLTP dimension table are normalised • In OPAP dimension table are de-normalised	• It is always de-normalised
• Dimension table contains less data	• Fact table contains more data
• It contents Horizontal Table	• It contents Vertical Table

STAR SCHEMA	SNOW FLAKE SCHEMA
A centralised located fact table which surrounded by multiple dimension table.	t is the extension star schema, in which dimension table exploded into another dimension table
It is top down model	It is bottom up model
Dimension table are normalised	Dimension table are de-normalised
Simple in structure hence joins and query are simple	Complex or complicated in structure hence joins and tables are complicated
Execution of the SQL query is high Performance	Performance of the query is less

CHAR	VARCHAR2
• Fixed Length	• Flexible Length
• Name char(10)	• Name varchar2(10)
• Raju ------ (‘-‘ empty is unused)	• Raju------ (‘-‘ send back to DB)
• Maximum size of length is 0 - 255	• Maximum size of length is 0 - 255

DROP	TRUNCATE
We delete the table in a single shot permanently from DB	We delete the table records in a single shot permanently from DB but structure remains same.
Performance is high	Performance is low

DELETE	DROP
It is a DML command	It is DDL command
Used to delete the records from the table	Used to delete structure of table
No Auto Commit	Auto Commit
Data can be roll back before commit	No roll back

IN	EXSIST
Low Performance	High Performance
Maintains column names	Does not Maintains column names
It is effective when most of filter criteria in sub query	It is effective when most of filter criteria in main query

SUB or NESTED or INNER QUERY	CORELATE SUB QUERY
Inner and outer query are independent	Inner and outer query are interdependent
Outer query executed once	Outer query executed twice
High Performance	Low Performance

NVL	NVL2
Replace only null column	Replace both not null and null column
Ex: (comm,’5000’) null	Ex. (comm.,’5000’,’2000’) not null null
It passes two parameters	It passes three parameters

DELETE	TRUNCATE
Similarities: In these both commands the we delete data from the table but structure remains same.	
• We delete the specific row or whole data from the table.	• We delete all the data from the table at a single shot.
• By using where clause we delete specific row	• Truncate doesn't support the where clause.
• Performance is low	• Performance is high
• It is a DML command	• It is DDL command
• These are not auto commit	• These are auto commit
• Can be roll-back	• Cannot be roll-back

TRANSLATE	REPLACE
It is used to translate the character of string by new	It is used to replace the part of a string by new string
bcddef	Ddddef
Character is to be replaced	String is to be replaced
Ex: Raj - j to m ram	Ex: Raj to Sam

DECODE	CASE
It is an Oracle property	It is an a ANSI SQL
We use equal operator	We use any condition operator
Used with any select statement	Used with select & update commands

WHERE	HAVING
It is used when we want to specify the condition for the column	It is used when we want to specify the condition for the group function
It can be used with select, update, delete	It cannot be used
Used before group by clause	It used after group by clause
We cannot use where clause in aggregate function	We can use having clause in aggregate function

UNION	UNIONALL
It retrieves the data without Duplicate	It retrieves the data with Duplicate
Low performance	High performance
Sorts results in ascending order	No any sorting of result

DISTINCT	UNIQUE
Is a clame	Is a constrain
Used whenever we are retrieving the data	Used whenever we are in sorting data
It will not accept null value	It will accept null value

UNIONALL	EQUJOIN
It retrieves total number of records from table irrespective of condition	It retrieves total number of records from table with respect to condition
It combines two rows of same table	Here we need two tables or two columns of two different tables
Here column data type should be same	Here column data type may or maynot be same

JOIN	UNIOIN
It combines data from many tables based on matched condition between them	It combines result set of two or more select statement
Combines data into new column	It combines data into new rows
May not return distinct column	It return distinct rows

PRIMARY KEY (UNIQUE+NOT NULL)	**UNIQUE KEY**
Whenever we define primary key for column the system will generate index automatically	Whenever we define unique key for column then we have to define the index
Primary key is only for column in most of cases	Unique key is no of column
It will not accept null values	It will accept null values

PRIMARY KEY	FOREIGN KEY
Each table can have only one primary key	The table can have multiple foreign key
PK=unique + not null	Null & duplicate
Cluster index	Non cluster index
Relation between fact table and dimension table	In foreign key relation between some dimension & fact table

PRIMARY KEY	SURROGATE KEY
It is used for maintaining the unique record in OLTP data base	It is used to maintain records in OLAP data base or data warehouse
Primary Key is always alphanumeric	Surrogate key is always numeric number
Primary Key is a business attribute	S.K is technical attribute
It is table data	It is not table data
It is given by user sometimes automatically generated	S.K is generated Automatically

VIEW	TABLE
View stores the record logically	Will store the record physically
Will not accept any space in DB	Table will occupy the space
Store SQL quarries	Table stores data

VIEW	MATERIALIZED VIEW
View stores the data logically	It stores the data physically
After the dropping the table view cannot be accessed	Even after dropping the MV is accessible
Purpose is for security	For performance purpose
Performance is low	Performance is more
We can update the record in view	We can't update the record

NOT NULL	UNIQUE
Not null will not accept the null values	Accept the null values
It accept the duplicate record	It will not accept the duplicate records

STORE PROCEDURE	FUNCTION
It may or may not return values	It must return a values
We can use all DML statement	It cannot use DML statement
It cannot be called from select statement	It can be called from select statement

ETL Testing	Manual Resting
The process of extracting data from multiple sources (Flat file, XML, SAP) more simpler with help of tools	Loading the data other than flat files and oracle table need more effort
High & clear visibility of logic	Complex and not so user friendly visibility of logic
Contains Meta data and changes can be done easily	No Meta data concept and changes need more effort
Can handle historic data very well	Maximum effort from maintenance point of view