3/134/

Assignment 2

Title: Design and develop SQL DDL statements which demonstrate the use of SQL objects, such as creation of: Table, View, Index, Sequence, Synonym

Problem Statement: Implement DDL commands in context of view, index, sequence

Learning Objective:
- To understand & implement the various DDL Commands.
- To understand database concepts like view, index, sequence and synonym.

Learning Outcome: The students will be able to
- Use and Implement the DDL commands like Create the tables, Alter the table structure.
- Implement view, index (types of index), synonym and sequence concept.

Software packages and hardware apparatus used:
- MySQL
- 64-bit Linux based open source OS
- 8 GB RAM

Concept related theory:
DDL COMMANDS: DDL is short form of Data Definition Language, which deals with data schemas and description, of how data can reside in database
Various commands in DDL are:

Create:
Create table command defines each attribute uniquely. Each attribute has 3 mandatory things.
a. Attribute name
b. Attribute size
c. Data type

3/134/

Syntax:

Create table tablename (Attribute_name attribute_datatype(size),Attribute_name attribute_datatype(size),Attribute_name attribute_datatype(size).....n)

Alter:

By using ALTER command existing table can be modified.

Adding New Columns

Syntax: Alter table ADD ((), .n)

Dropping a Column from the Table

Syntax: Alter table DROP COLUMN

*This command will drop a particular column. *

Modifying Existing Table

Syntax: Alter table MODIFY (())

Restriction on the Alter table

Using the Alter table clause, the following tasks cannot be performed.

1. Change the name of the table
2. Change the name of the column
3. Decrease the size of a column if table data exists

Drop:

The Drop command will destroy table along with the data enteries in it.

Syntax: Drop table

Truncate:

The truncate command deletes all entries existing in tables but keep the structure of table

3/134/

Syntax: Truncate Table

Rename:
The rename command is used to rename the table
Syntax: Rename

Creating Views:
Database views are created using the CREATE VIEW statement. Views can be created from a single table, multiple tables, or another view. To create a view, a user must have the appropriate system privilege according to the specific implementation.

Syntax: Create View view_name As Select column1, column2.....From table_name Where condition;

Updating a view:
A view can be updated with a CREATE or REPLACE view command
Syntax: Create or Replace View view_name As Select column1, column2, ...From table_name Where condition

Dropping views:
Syntax: Drop View view_name

INDEX
An index can be created in a table to find data more quickly and efficiently.
The users cannot see the indexes, they are just used to speed up searches/queries.
Note: Updating a table with indexes takes more time than updating a table without (because the indexes also need an update). So, you should only create indexes on columns (and tables) that will be frequently searched against.

Creating index:

Syntax: Create Index index_name ON table_name (column1, column2 ...)

Drop index:

Syntax: Alter table table_name Drop Index index_name

SEQUENCES

Sequences are frequently used in the databases because many applications require each row in a table to contain a unique value and sequences provide an easy way to generate them. The simplest way to use sequences in mysql is to define as AUTO_INCREMENT and leave the remaining things to mysql

Generic syntax: Create Sequence sequence_name Start With initial-value Increment By increment-value maxvalue maximum-value cycle / nocycle

SYNONYMS

A synonym is an alternative name for objects such as tables, views, sequences, stored procedures and other database objects.

You generally use synonyms when you are granting access to an object from another schema and you don't want the users to have to worry about knowing which schema owns the object.

Syntax: Create Synonym For .

Note: Synonyms are not possible in mysql. The above syntax works in OracleSQL.

Conclusion:

Through this assignment, I have implemented all the DDL statements queries and also implemented views, indexes, sequences and synonyms