

Expt

Task 2
Date:
5/8/23

Implementation of DDL & DML commands

Aim: To implement the different queries in DDL & DML involved in the design & implementation of a database system in MySQL.

i) DDL Queries

- * Create table: helps create a new table

Query:

```
>>> Create table students( studid int, stuname  
                         varchar(10), rollno int, marks int);
```

Output:

Table created

- * Describe: shows fields & types of data.

Query:

```
>>> Desc students;
```

Output:

Field	Type
STUID	Number (38)
STUNAME	VARCHAR (10)
ROLL NO	NUMBER (38)
MARKS	NUMBER (38)

- * Drop Table: (deletes the table)

Query:

>> Drop table students;

Output:

Table deleted.

- * Alter Table: (Adds fields in a table)

Query:

>> ALTER TABLE Students ADD subject varchar(10);

Output:

subject field added.

- (ii) DML Query

- * Insert into => (inserts records into the table)

Query:

>> Insert into students values (3, 'shani',
643, 78);

- * SELECT - (Retrieves data from one or more tables)

Query:

>> Select * from students;

Output:

STUDID	STUNAME	ROLLNO	MARKS
1	Anubhav	128	84
2	Shami	190	92
3	Shekhar	112	88
4	Aman	001	56

★ UPDATE \Rightarrow (Modifies existing data)

>>> UPDATE students SET STUDID=20 WHERE ROLL NO =190 ;

Output:

1 row Updated.

★ Delete : (Deletes one or more rows from a table)

Query:

>>> Delete from students where stud=2;

Output

1 Row Deleted.

* SELECT (Retrieves Records that satisfies the condition)

Query:

>>> SELECT * From Students where studid = 1;

STUDID	STUNAME	ROLLNO	MARKS
1	Anubhav	12 8	84

C

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	3
VIVA VOCE (5)	5
RECORD (5)	2
TOTAL (20)	15
ON WITH DATE	

C
6/12/2015

Result:

Therefore, DDL & DML commands using MySQL has been implemented successfully.

DDL And DML Command with Query constraintsAim:

To design and implement a database for a sports event Management System that manages information about teams, coaches, players and matches using SQL DDL.

Steps : 1) Identify Entities

- Team
- Coach
- Player
- Match

2) Identify Attributes

- Team → (Team, Name, Homeground, Coach ID)
- Coach → (Coach ID, Name, Experience)
- Player → (Player ID, Name, Position, Team ID)
- Match → (Match ID, Match Date, Home Team ID)

3) Identify Relationship

- Team - Coach → one-to-one
- Team - Player → one-to-many
- Match - Team → many-to-one

4) Reframe Relation with Keys & constraints

→ Primary Key - Team ID, Coach ID, Player ID
→ Match ID.

- Foreign keys → Coach ID in Team, TeamID

1) DDL COMMANDS

* Table for Teams

CREATE Table Team (

Team ID INT Primary key,

, Team Name Varchar(50) NOT NULL,

Homeground Varchar(50),

CoachID INT UNIQUE,

Foreign Key (CoachID) References Coach
(CoachID))

);

Table for Coaches

Create Table Coach (

CoachID INT Primary key,

coachName Varchar(50) NOT NULL,

Experience INT,

TeamID Unique,

Foreign Key (TeamID) References

Team (TeamID)

);

Table for Players

Create Table Player (

Player ID INT Primary Key,

Player Name Varchar (50),

Position Varchar (30)

TeamID INT,

Foreign Key (TeamID) References

Team (TeamID)

);

SQL> DESC COACH;

Name	Null?	Type
COACHID	NOT NULL	NUMBER (38)
NAME	NOT NULL	VARCHAR (50)
EXPERIENCE		NUMBER (38)
TEAMID		NUMBER (38)

SQL> DESC PLAYER;

Name	NULL?	Type
PlayerID	NOT NULL	NUMBER (38)
Name	NOT NULL	VARCHAR (50)
Position		VARCHAR (50)
TeamID		NUMBER (38)

SQL> DESC Team;

Name	NULL?	Type
TeamID	NOT NULL	NUMBER(38)
Name		VARCHAR2(50)
HomeGround		VARCHAR2(50)
CoachID		NUMBER(38)

SQL> DESC Match;

Name	NULL?	TYPE
MatchID	NOT NULL	NUMBER(38)
Match Date	NOT NULL	DATE
HomeTeamID		NUMBER(38)
AwayTeamID		NUMBER(38)

2) DML COMMANDS

1) Insert

SQL> Insert into team (teamID, Name, Homeground, coachID) values (1, 'Warriors', 'Stadium A', Null);

1 row created 2) Select

SQL> Select * from team;

TEAMID	Name	Homeground	Coach ID
1	Warriors	Stadium A	NULL

3) UPDATE

SQL> UPDATE team
 2 set Homeground = 'Stadium B'
 3 WHERE TeamID = 1;

TeamID	Name	Homeground	CoachID
1	Warriors	Stadium B	NULL

4) Delete

SQL> ~~DELETE FROM team
 2 WHERE TEAMID = 1;~~

1 row deleted.

VEL TECH	
EX NO.	2-1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	65
SIGN WITH DATE	R

Result:

The the Sql command for DDL & DML query has been executed successfully.