

Ex: no: 1

Date: 3/8/22

## IMPLEMENTING DDL COMMANDS

Aim :-

To create a table 'emp' and perform implementing DDL commands with Create, alter, drop and modify.

Fields	Data type	Constraints
Empno	Number (4)	Primary key
ename	char (10)	

- a.) Create a table emp- with the above field.
- b.) Insert records into the table
- c.) Modify existing columns.
- d.) Modify the structure of the table
- e.) Drop a column from a table
- f.) Rename the table
- g.) Truncate the table
- h.) Destroy the table.

Algorithm:-

Step 1: Start the process

Step 2: Create a table using Create table command.

Step 3: Create table <table name>, <columns>, <data types>, <size>, <constraints>, <data types> <size>;

Step 4: Insert records into the table using  
insert into <table name> Value <fields>  
field 2000;

Step 5: Modify the structure of the table  
using alter table <table name> add  
<new cols>, <data>, <size>;

Step 6: Drop a column from a table using  
alter table <table name> drop column  
<column names>;

Step 7: Modify existing column using alter  
table <table name> modify <col name>  
<new data types><new size>;

Step 8: Rename the table using rename  
command rename <old table name>  
To <new table name>

Step 9: Truncate the table using truncate  
command truncate table <table name>

Step 10: Destroy the table using drop command  
drop table <table name>....

Result:- The above San programming queries  
for the implementation of RPN  
Commands were executed and  
Obtained the result successfully.

Ex: no: 2

Date: 12/8/09

## IMPLEMENTING DML COMMANDS

Aim:-

To create a table 'emp' and perform implementing DML Commands with 'Select', 'Insert', 'Update' and delete.

Field	Data type	Constraints
Eno	Varchar(3)	Primary key
Ename	Varchar(10)	Primary key
City	Varchar(10)	Primary key

Create a table 'emp' and insert 5 record into the table.

List all the record from the table.

Select the name of the employee from Calcutta.

~~Update existing data change the city name code instead of titchy~~

Update name of the employees from agathiyam to haseen.

Algorithm:-

Step 1: Start the process

Step 2: Create the table 'emp' using the create table command.

Step 3: List all the records from the table

- Step 4: Insert all the record into the table.
- Step 5: List the name of the employee from Calimbatore using Select and Update Command.
- Step 6: change the existing data by city name encode instead of tsicky.
- Step 7: change the employee name as naveen instead of agathiyam.
- Step 8: Stop the process.

Result: The above SQL programming queries for the implementation of DML commands were executed and obtained the result successfully.

Ex: no: 3

Date 16/3/22

### STUDENT MARKLIST

Aim:-

To create a table marklist with the following fields.

Fields	Data type
Sno	Number(2) Varchar(10)
Name	Varchar(8)
Reg no	Number(2)
tamil	Number(2)
english	Number(2)
Language	Number(2)
dbms	Number(2)
total	Number(3)
result	Number(4)

- a.) Insert marklist records into the table
- b.) List the name of the Student who passed in clanguage and DBMS.
- c.) List all the Student who passed in english
- d.) List the Student details whose total mark is greater than 300.

Algorithm:-

Step 1: Start the process..

Step 2: Insert a table "Marklist" by using Create Table Command.

Step 3: Insert Mark Records by using Insert Command.

Step 4: List the field DBMS by using Select Command and Where Clause.

Step 5: List the field English by using Select Command and Where Clause.

Step 6: List the Student Mark greater than 300 by using the Same above Command.

Step 7: List the Students details whose Result is Pass.

Step 8: Stop the Process.

Result: The above San Programming queries for a Student's marklist table were executed and obtained the result successfully.

Ex: no: 4

Date: - 25/8/02

## EMPLOYEE DETAILS

Aim:-

To create a table "Employee" with following fields

Fields	Data type	Constraints
Emp no	Vaachas 2	6
Ename	Vaachas 2	15
DOB	Date	
BR. No	Vaachas 2	10
Dept	Vaachas 2	10
Desin	Vaachas 2	10
Salary	Numbers	5

- a.) insert the employees record into the table.
- b.) list all the employee details of the company.
- c.) list the employee name Starts with "a".
- d.) list the employee who get Salary more than 10000.
- e.) list the name of the employees in marketing department.
- f.) list the name of employees financial department who gets Salary below 10000.

g) List the name of the employee who join after 2000.

Algorithm:-

Step 1: Start the process

Step 2: Create the table "employee" using create table command.

Step 3: Insert the employee records by using insert table command.

Step 4: List all the employee details of the company using Select Commands.

Step 5: List all the employees name the Starts with "a" using Select and Where Command.

Step 6: List the name of the employees who get Salary more than 10000 using Select and Where Command.

Step 7: List the name of the employee in marketing department.

Step 8: List the name of the employees in financial department

Step 9: List the name of the employees who joined after 2000.

Step 10: Stop the Process.

Result:- The above SQL programming queries  
for an ~~employee~~ table were executed  
and obtained the results successfully.

Ex: no: 5

Date: 2/9/22

### CUSTOMER DETAILS

Aim:-

To create a table 'customers' with the following details.

Fields	Data type
cust name	Varchar(20)
Gender	char(1D)
DOB	Date
TYPE	Varchar(5)
Address	Varchar(10)
city	Varchar(10)
Last Purchased	Date

- ① Insert customer records into the table
- ② List all male customers
- ③ List all female customers who live in  
Cambatore
- ④ List all customers who bought thing today  
for cascade.

Algorithm:-

Step 1: Start the process

Step 2: Create a customer table by using  
create table command.

Step 3: Insert records by using insert command

Step 4: List all male customers who live in  
Carmabase by using Select  
Command..

Step 5: List all female customers who live  
in Carmabase using Select Command.

Step 6: List all customers who bought things  
today for credit by using Select  
Command.

Step 7: Stop the process.

Result: The above SQL programming queries  
for customers table were executed  
and obtained the result successfully.

Page no: 6

Date: 9/9/22

## FIBONACCI SERIES AND FACTORIAL

Aim:-

To write a C++ console program to find the factorial and fibonacci series for given number.

Algorithm:-

Step 1: Start the process

Step 2: Declase the Variables

Step 3: Begin the factorial and fibonacci Series.

Step 4: Calculate the factorial and fibonacci series with the loop.

Step 5: Stop the Process.

Result: The above C++ program for finding the fibonacci Series and factorial with a loop was executed and obtained the result successfully.

Ex: no: 7

Date: 12/9/22

## STUDENT DATABASE

Aims:-

To create a table "Student" database with the following fields.

Field	Data type	constraints
Rno	Numbers	5
Name	Vaschad a	15
DOB	Date	
Course	Vaschad a	10
Dept	Vaschad a	10
Semester	Vaschad a	2
Percentage	Numbers	3

- a.) Insert Student database record into the table
- b.) Select all record
- c.) Select sno, name, Percentage, from Student.
- d.) Select details of all Student with percentage greater than 75
- e.) Select name, sno, of all in order by percentage
- f.) Select os Student in descending order
- g.) Select name of Student not in CS department

h) Select all record where name starts with the alphabet 'A'.

Algorithm :-

Step 1: Start the process

Step 2: Create a table 'Student' using create table command.

Step 3: Insert Student records by using insert command.

Step 4: Select sno, name, percentage from Student by using select command.

Step 5: Select details of all Student with percentage greater than 75 using select and ~~and~~ where clause.

Step 6: Select sno, name, of all Student stored in order by percentage.

Step 7: Select the name of all Student belonging to 3rd Semester Bsc CS

Step 8: Select name of the Student not in CS department

Step 9: Select all the records from Student where name starts with alphabet 'A'

Step 10: Stop the Process.

Result:- The above SQL programming Queries  
for Student database were executed  
and obtained the result successfully.

Ex: no: 8  
Date: - 19/9/22

## ITEM SALES

Aim:-

To Create the tables Item (itemCode, name, sell price, quality hand) and Sale (item code, day sold) with the required.

- a.) Change the name of the item from "System" to 'Computer System'.
- b.) Delete all the sales records if the name is 'mouse'
- c.) Insert the new table attribute 'tax' to sale tables.
- d.) Fill the tax table with the following values 4% of selling price \* sold quality.
- e.) Delete the records from the both table if the item have not been sold for the past 20 days..

Algorithm:-

Step 1: Start the process

Step 2: Create a item table

Step 3: Create a second table item Sales.

Step 4: Insert the values

Step 5: change the name of the item from 'System' to 'computer System'.

Step 6: Delete all the Sales records if the item is "mouse".

Step 7: Insert a new column "tax" to Sales.

Step 8: fill the text with the following 4% of selling price \* sold quantity.

Step 9: Delete the records from both table if the item have not been sold for the past 30 days.

Step 10: Display all the queries.

Step 11: Stop the process.

Result: The above SQL programming queries for the item and Sales table were executed and obtained the result successfully.

Ex: no: 9

Date: 20/9/20

## VEHICLE DETAILS

Aim:-

To write an SQL program to create a view "VehicleView" from the "Vehicle" table with the given fields and to execute the following queries.

Fields	Data types
Vcode	Number(5)
Vname	Varchar(10)
Type	Varchar(10)
company	Varchar(10)
Onroadprice	Number(7)
Colour	Varchar(5)
Date purchased	Date
model	Varchar(5)

a.) Create a table "Vehicle" with given fields.

b.) Insert the Vehicle records to the table

c.) Create a View "VehicleView" from Vehicle table from Vehicle table for Colour = "black" and type = 'Sedan'

d.) List the records of the Vehicle table.

- c.) Delete the vehicles which were purchased before the year from the view.
- d.) Select the records from the view having Onroadprice between 40,000 and 50,000.
- e.) Drop the view.
- f.) Select all the vehicles which in colour and observe the difference between view and table.

Algorithms -

Step 1: Create the process

Step 2: Create a table "Vehicle" by using create table command.

Step 3: Create the Sequence using the ~~Sequence Statement with the Start as one and increment by one.~~

Step 4: Enter the records of View using the Sequence.

Step 5: Create the View on Vehicle table only for Silver colour for four wheels.

Step 6: Delete the vehicle which were purchased before one year.

Step 7: Select the records from the View having onroad price between 3 and 7 Lakhs.

Step 8: Drop the View using drop View Statement.

Step 9: Select Silver colour Cars from View and table.

Step 10: Stop the process.



Result: The above SQL programming queries for the "View" and the Vehicle table were executed and obtained the result successfully.

Ex.no:10

Date 22/9/22

## SUPPLIER DETAILS

Aim:-

To write an SQL program to create a Suppliers table "Sup" and its child table "Supcode" with the given fields and to execute the following queries.

Main Table : Sup

Fields	Data types	Constraints
Sup code	Vaachar 2(3)	Primary Key
Name	Vaachar 2(10)	not null
Gender	char(1)	check (m/f)
TYPE	Vaachar 2(6)	check (credit/cash)
Address	Vaachar 2(50)	-
City	Vaachar 2(10)	-
Mobile	Number (10)	unique

Child Table : Supcode.

Fields	Data type	Constraints
Supcode	Vaachar 2(5)	Reference key
Itemname	Vaachar 2(10)	not null
Price	Number (3)	check
Delivered Within	Number (a)	

- a) Create a table "Sup" and a child table "Supcode" given fields.

- b.) Insert record into both the tables.
- c.) List the Suppliers name and city who supply Ice cream.
- d.) List the Suppliers name who supply biscuits within the Supplier name.
- e.) List the number of different item along with the Supplier name.
- f.) List the female Suppliers who supply for credit but do not Supply 'Cake' and delivered within 2 days.

Algorithm:-

Step 1:- Start the process.

Step 2:- Create a parent table as "Sup" by using create table command.

Step 3:- Create a child table as "Supply" "Items" using create table command.

Step 4:- ~~Insert~~ the Suppliers and Supply item details those table using insert command.

Step 5:- List the Supplier name and city who Supply Ice cream Select command.

Step 6:- List the total number of different item along with the Suppliers name.

Step 7:- List the Supplier name who supply Biscuit within a Day.

Step 8:- List the female Suppliers who Supply for credit but do not Supply 'Coke' delivered within 2 days.

Step 9:- Stop the process

Result:- The above San Programming queries for the tables ~~Sup~~ and Supcode were executed and obtained the result successfully.

## TRIGGER

Aim:-

To create a table "triggers" with the fields: id, distance, fare and observe the MySQL trigger events on update, insert and delete command.

Algorithm:-

Step 1: Start the program

Step 2: Create a table "triggers1" with the required fields.

Step 3: Insert the relevant and scattered records using insert command.

Step 4: Create a MySQL Trigger code for triggers1 table to the changes occur for update, insert and delete command

Step 5: Use the update, insert and delete commands on the table triggers1 and display the results.

Step 6: Stop the process.

Result: The above MySQL trigger was successfully executed and obtained the result.

## EXCEPTION HANDLING.

Aim:-

To illustrate a Pysan program to set exception handling in customer table.

Create a table "customers" with the following fields : custid number(6), name Varchar(25), address Varchar(25).

Algorithm:-

Step 1: Start the program

Step 2: Create a table "customers" with the required fields and insert records.

Step 3: Display the customers table using Select command.

Step 4: Create a Pysan code to Set Exception handling to customers table.

Step 5: check the condition whenever an exception occurs and display the message accordingly. To do this, write the code in begin section to give the condition. If the C-ID  $C=0$ , then display the error message as "id must be greater than zero".

Step 6: Write the code in exception section to find if a customer id is used not found and display the message accordingly such as id must greater than zero or "no such customer".

Step 7: Handle the exception when a user enters the value and display the result.

Step 8: Stop the process.

Result: The above Python Exception handling program was successfully executed and obtained the result.