1. Statement to create the Contact table.

* CREATE TABLE Contact (

    ContactID INT PRIMARY KEY,

    CompanyID INT,

FirstName VARCHAR(45),

LastName VARCHAR(45),

Street VARCHAR(45),

City VARCHAR(45),

State VARCHAR(2),

Zip VARCHAR(10),

isMain Boolean,

Email VARCHAR(45),

    Phone VARCHAR(12)

);

1. Statement to create the Employee table .

* CREATE TABLE Employee (

    EmployeeID INT PRIMARY KEY,

    FirstName VARCHAR(45),

    LastName VARCHAR(45),

Salary DECIMAL(10,2),

HireDate DATE,

    JobTitle VARCHAR(25),

Email VARCHAR(45),

Phone VARCHAR(12),

**);**

3) Statement to create the ContactEmployee table

HINT: Use DATE as the datatype for ContactDate. It allows you to store the date in this format: YYYY-MM-DD (i.e., ‘2014-03-12’ for March 12, 2014).

* CREATE TABLE ContactEmployee (

    ContactEmployeeID INT,

ContactID INT,

    EmployeeID INT,

    ContactDate DATE,

Description VARCHAR(45),

);

4)In the Employee table, the statement that changes Lesley Bland’s phone number to 215-555-8800

* UPDATE Employee

SET PhoneNumber = '215-555-8800'

WHERE FirstName = 'Lesley' AND LastName = 'Bland';

5) In the Company table, the statement that changes the name of “Urban Outfitters, Inc.” to “Urban Outfitters” .

* UPDATE Company

SET CompanyName = 'Urban Outfitters'

WHERE CompanyName = 'Urban Outfitters, Inc.';

1. In ContactEmployee table, the statement that removes Dianne Connor’s contact event with Jack Lee (one statement). HINT: Use the primary key of the ContactEmployee table to specify the correct record to remove.

* irst, find the correct ContactID and EmployeeID, then:

DELETE FROM ContactEmployee

WHERE ContactID = <contact\_id\_of\_event>

AND EmployeeID = <employee\_id\_of\_Jack\_Lee>;

(Replace <contact\_id\_of\_event> and <employee\_id\_of\_Jack\_Lee> with actual IDs)

1. Write the SQL SELECT query that displays the names of the employees that have contacted Toll Brothers (one statement). Run the SQL SELECT query in MySQL Workbench. Copy the results below as well.

* SELECT DISTINCT e.FirstName, e.LastName

FROM Employee e

JOIN ContactEmployee ce ON e.EmployeeID = ce.EmployeeID

JOIN Contact c ON ce.ContactID = c.ContactID

JOIN Company co ON c.CompanyID = co.CompanyID

WHERE co.CompanyName = 'Toll Brothers';

1. What is the significance of “%” and “\_” operators in the LIKE statement?

* %: Matches any number of characters (including zero).

\_: Matches exactly one character.

Examples:

WHERE name LIKE 'A%'     -- Starts with A

WHERE name LIKE '\_a%'    -- Second letter is 'a'

1. Explain normalization in the context of databases.

* Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity by splitting data into related tables. The main forms include:
* 1NF: Eliminate repeating groups
* 2NF: Remove partial dependencies

• 3NF: Remove transitive dependencies

1. What does a join in MySQL mean?

* JOIN is used to combine rows from two or more tables based on a related column.

Types:

* INNER JOIN: Matches rows in both tables
* LEFT JOIN: All rows from left table, matched rows from right
* RIGHT JOIN: All from right, matched from left
* FULL OUTER JOIN (not native to MySQL but can be simulated)

1. 19.What do you understand about DDL, DCL, and DML in MySQL?

|  |  |  |
| --- | --- | --- |
| Type | Meaning | Examples |
| DDL | Data Definition Language | CREATE, ALTER, DROP |
| DML | Data Manipulation Language | SELECT, INSERT, UPDATE, DELETE |
| DCL | Data Control Language | GRANT, REVOKE |

12) What is the role of the MySQL JOIN clause in a query, and what are some common types of joins?

* The JOIN clause merges rows from two or more tables based on a common field. It allows combining data across tables for richer insights.