## Hands-on Lab: Working with Multiple Tables in MySQL using phpMyAdmin

Estimated time needed: 20 minutes

In this lab, you will learn how to create tables and load data in the MySQL database service using the phpMyAdmin graphical user interface (GUI) tool.

### Software Used in this Lab

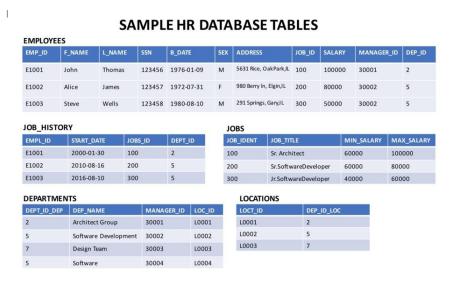
In this lab, you will use MySQL MySQL is a Relational Database Management System (RDBMS) designed to efficiently store, manipulate, and retrieve data.



To complete this lab you will utilize MySQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

### **Database Used in this Lab**

The database used in this lab is an internal database. You will be working on a sample HR database. This HR database schema consists of 5 tables called **EMPLOYEES**, **JOB\_HISTORY**, **JOBS**, **DEPARTMENTS** and **LOCATIONS**. Each table has a few rows of sample data. The following diagram shows the tables for the HR database:



## **Objectives**

After completing this lab you will be able to:

- Write SQL queries that access more than one table
- Compose queries that access multiple tables using a nested statement in the WHERE clause
- · Build queries with multiple tables in the FROM clause
- Write Implicit Join queries with join criteria specified in the WHERE clause
- Specify aliases for table names and qualify column names with table aliases

In this lab, you will through some SQL practice problems that will provide hands-on experience with SQL queries that access multiple tables. You will be:

- · Accessing Multiple Tables with Sub-Queries
- Accessing Multiple Tables with Implicit Joins

How does an Implicit version of CROSS JOIN (also known as Cartesian Join) statement syntax look?

- 1. 1
- SELECT column\_name(s)
- FROM table1, table2;

Copied!

How does an Implicit version of INNER JOIN statement syntax look?

- l. 1
- 2. 2
- 2. 2

about:blank 1/7

- SELECT column\_name(s)
- FROM table1, table2
   WHERE table1.column\_name = table2.column\_name;

Copied!

# **Exercise 1: Accessing Multiple Tables with Sub-Queries**

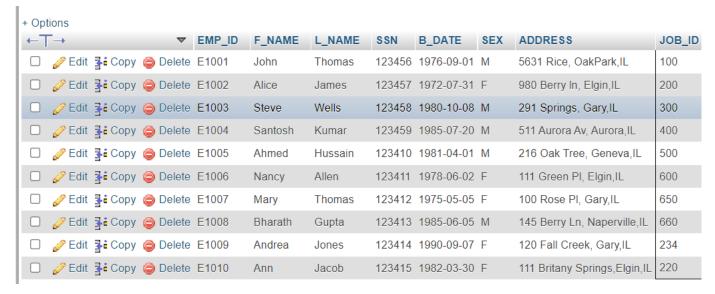
1. Problem:

Retrieve only the EMPLOYEES records that correspond to jobs in the JOBS table.

**▼** Solution

 select \* from EMPLOYEES where JOB\_ID IN (select JOB\_IDENT from JOBS); Copied!

**▼** Output



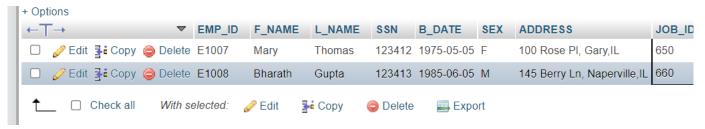
### 2 Problem:

Retrieve only the list of employees whose JOB TITLE is Jr. Designer.

▼ Solution

1. select \* from EMPLOYEES where JOB\_ID IN (select JOB\_IDENT from JOBS where JOB\_TITLE= 'Jr. Designer'); Copied!

▼ Output



3. Problem:

Retrieve JOB information and who earn more than \$70,000.

**▼** Solution

1. select JOB\_TITLE, MIN\_SALARY,MAX\_SALARY,JOB\_IDENT from JOBS where JOB\_IDENT IN (select JOB\_ID from EMPLOYEES where SALARY > 70000 ); Copied!

▼ Output

about:blank 2/7



4. Problem:

Retrieve JOB information and list of employees whose birth year is after 1976.

▼ Solution

1. 1
1. select JOB\_TITLE, MIN\_SALARY,MAX\_SALARY,JOB\_IDENT from JOBS where JOB\_IDENT IN (select JOB\_ID from EMPLOYEES where YEAR(B\_DATE)>1976 );

Copied!

▼ Output



5. Problem:

Retrieve JOB information and list of female employees whose birth year is after 1976.

▼ Solution

1. select JOB\_TITLE, MIN\_SALARY,MAX\_SALARY,JOB\_IDENT from JOBS where JOB\_IDENT IN (select JOB\_ID from EMPLOYEES where YEAR(B\_DATE)>1976 and Copied!

▼ Output



# **Exercise 2: Accessing Multiple Tables with Implicit Joins**

1. Problem:

Perform an implicit cartesian/cross join between EMPLOYEES and JOBS tables.

▼ Solution

```
1. 1
1. select * from EMPLOYEES, JOBS;
Copied!
```

▼ Output

about:blank 3/7

+ Options <b>EMP ID</b>	F NAME	L NAME	SSN	B DATE	SEX	ADDRESS	JOB_ID	SALARY	MANAGER ID	DEP_I
E1010	Ann	Jacob		1982-03-30	0200	111 Britany Springs,Elgin,IL	220	70000.00	30004	5
E1009	Andrea	Jones	123414	1990-09-07	F	120 Fall Creek, Gary,IL	234	70000.00	30003	7
E1008	Bharath	Gupta	123413	1985-06-05	M	145 Berry Ln, Naperville,IL	660	65000.00	30003	7
E1007	Mary	Thomas	123412	1975-05-05	F	100 Rose Pl, Gary,IL	650	65000.00	30003	7
E1006	Nancy	Allen	123411	1978-06-02	F	111 Green PI, Elgin,IL	600	90000.00	30001	2
E1005	Ahmed	Hussain	123410	1981-04-01	M	216 Oak Tree, Geneva,IL	500	70000.00	30001	2
E1004	Santosh	Kumar	123459	1985-07-20	M	511 Aurora Av, Aurora,IL	400	60000.00	30004	5
E1003	Steve	Wells	123458	1980-10-08	M	291 Springs, Gary,IL	300	50000.00	30002	5
E1002	Alice	James	123457	1972-07-31	F	980 Berry In, Elgin,IL	200	80000.00	30002	5
E1001	John	Thomas	123456	1976-09-01	M	5631 Rice, OakPark,IL	100	100000.00	30001	2
E1010	Ann	Jacob	123415	1982-03-30	F	111 Britany Springs,Elgin,IL	220	70000.00	30004	5
E1009	Andrea	Jones	123414	1990-09-07	F	120 Fall Creek, Gary,IL	234	70000.00	30003	7
E1008	Bharath	Gupta	123413	1985-06-05	M	145 Berry Ln, Naperville,IL	660	65000.00	30003	7
E1007	Mary	Thomas	123412	1975-05-05	F	100 Rose PI, Gary,IL	650	65000.00	30003	7
Consol	Nancy le	Allen	123411	1978-06-02	F	111 Green PI, Elgin,IL	600	90000.00	30001	2

### 2. Problem:

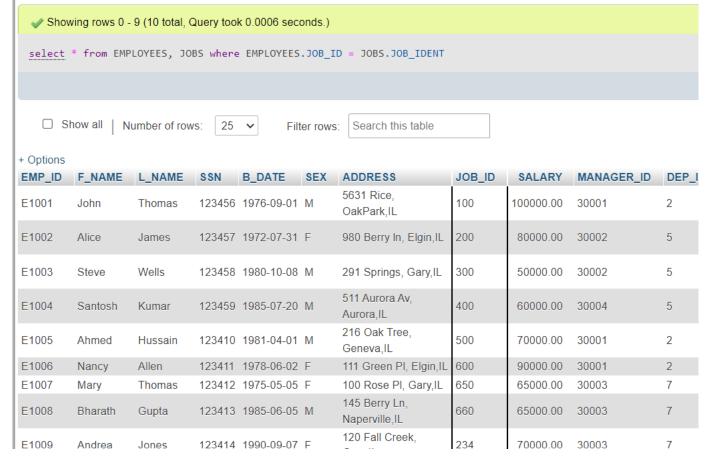
Retrieve only the EMPLOYEES records that correspond to jobs in the JOBS table.

### **▼** Solution

1. 1
 1. select \* from EMPLOYEES, JOBS where EMPLOYEES.JOB\_ID = JOBS.JOB\_IDENT;
 Copied!

 $\blacktriangledown$  Output

about:blank 4/7



### 3. Problem:

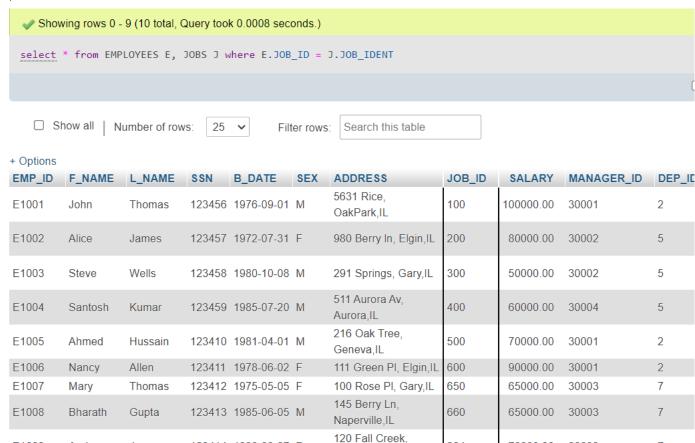
Redo the previous query, using shorter aliases for table names.

**▼** Solution

1. 1
1. select \* from EMPLOYEES E, JOBS J where E.JOB\_ID = J.JOB\_IDENT;
Copied!

**▼** Output

about:blank 5/7



Gary,IL

234

70000.00

30003

7

#### 4. Problem:

E1009

■ Console

Andrea

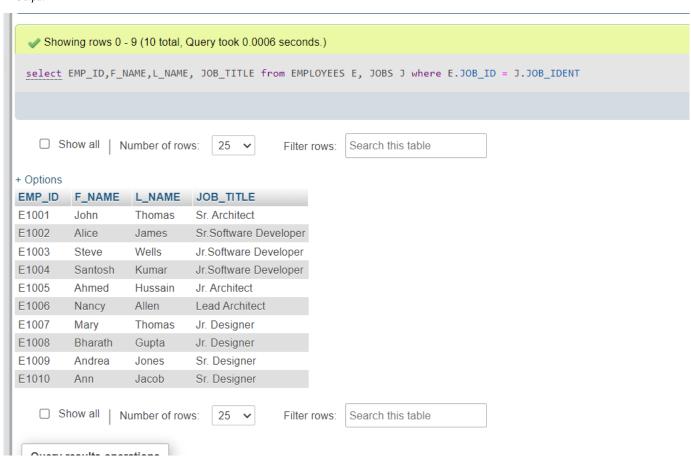
Jones

Redo the previous query, but retrieve only the Employee ID, Employee Name and Job Title.

- **▼** Solution
- 1. 1
  1. select EMP\_ID,F\_NAME,L\_NAME, JOB\_TITLE from EMPLOYEES E, JOBS J where E.JOB\_ID = J.JOB\_IDENT;
  Copied!

123414 1990-09-07 F

**▼** Output



5. Problem:

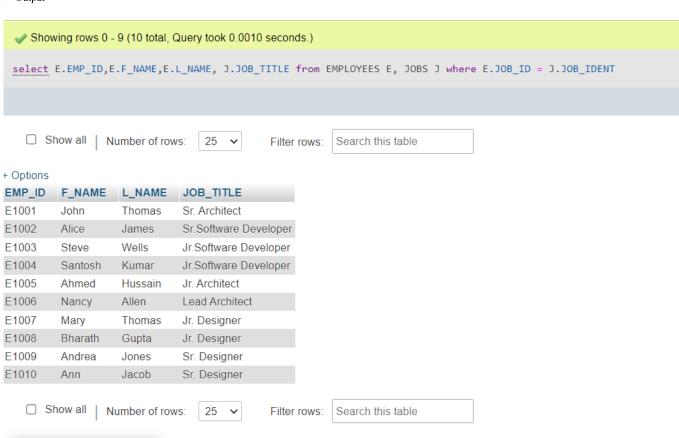
Redo the previous query, but specify the fully qualified column names with aliases in the SELECT clause.

**▼** Solution

```
1. 1
1. select E.EMP_ID,E.F_NAME,E.L_NAME, J.JOB_TITLE from EMPLOYEES E, JOBS J where E.JOB_ID = J.JOB_IDENT;

Copied!
```

**▼** Output



# **Solution Script**

If you would like to run all the solution queries of the SQL problems of this lab with a script, download the script below. Import the script to mysql phpadmin interface and run. Follow Hands-on Lab: Create tables using SQL scripts and Load data into tables on how to import a script to MYsql phpadmin interface and run it

• MultipleTables Solution Script.sql

Congratulations! You have completed this lab, and you are ready for the next topic.

# Author(s)

Lakshmi Holla

Malika Singla

# Changelog

Date	Version	Changed by	Change Description
2023-05-10	0.3	Eric Hao & Vladislav Boyko	Updated Page Frames
2023-05-04	0.2	Rahul Jaideep	Updated Markdown file
2021-11-01	0.1	Lakshmi Holla, Malika Singla	Initial Version

© IBM Corporation 2023. All rights reserved.

about:blank 7/7