Hands-on Lab: Working with Joins 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:

SAMPLE HR DATABASE TABLES

EMPLOYEES

EMP_ID	F_NAME	L_NAME	SSN	B_DATE	SEX	ADDRESS	JOB_ID	SALARY	MANAGER_ID	DEP_ID
E1001	John	Thomas	123456	1976-01-09	M	5631 Rice, OakPark,IL	100	100000	30001	2
E1002	Alice	James	123457	1972-07-31	F	980 Berry ln, Elgin,IL	200	80000	30002	5
E1003	Steve	Wells	123458	1980-08-10	M	291 Springs, Gary,IL	300	50000	30002	5

JOB_HISTORY

EMPL_ID	START_DATE	JOBS_ID	DEPT_ID
E1001	2000-01-30	100	2
E1002	2010-08-16	200	5
E1003	2016-08-10	300	5

JOBS

JOB_IDENT	JOB_TITLE	MIN_SALARY	MAX_SALARY
100	Sr. Architect	60000	100000
200	Sr.SoftwareDeveloper	60000	80000
300	Jr.SoftwareDeveloper	40000	60000

DEPARTMENTS

DEPT_ID_DEP	DEP_NAME	MANAGER_ID	LOC_ID
2	Architect Group	30001	L0001
5	Software Development	30002	L0002
7	Design Team	30003	L0003

LOCATIONS

LOCT_ID	DEP_ID_LOC
L0001	2
L0002	5
L0003	7

In this lab, you will run through some SQL practice problems that will provide hands-on experience with the different kinds of join operations.

NOTE: This lab requires you to have all 5 of these tables of the HR database populated with sample data on MySQL. If you don't have the tables above populated with sample data on MySQL, please go through the lab below first:

Hands-on Lab: Create and Load Tables using SQL Scripts

How does a CROSS JOIN (also known as Cartesian Join) statement syntax look?

```
1. 1
2. 2
3. 3
```

```
1. SELECT column_name(s)
2. FROM table1
3. CROSS JOIN table2;
```

Copied!

How does an INNER JOIN statement syntax look?

```
1. 1
2. 2
3. 3
4. 4
5. 5
```

```
1. SELECT column_name(s)
2. FROM table1
3. INNER JOIN table2
4. ON table1.column_name = table2.column_name;
5. WHERE condition;
```

Copied!

How does a LEFT OUTER JOIN statement syntax look?

```
1. 1
2. 2
3. 3
4. 4
5. 5
```

```
1. SELECT column_name(s)
2. FROM table1
3. LEFT OUTER JOIN table2
```

```
4. ON table1.column_name = table2.column_name
5. WHERE condition;
```

Copied!

How does a RIGHT OUTER JOIN statement syntax look?

```
1. 1
2. 2
3. 3
4. 4
5. 5
```

```
1. SELECT column_name(s)
2. FROM table1
3. RIGHT OUTER JOIN table2
4. ON table1.column_name = table2.column_name
5. WHERE condition;
```

Copied!

How does a FULL OUTER JOIN statement syntax look?

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
10. 10
11. 11
12. 12
13. 13
```

```
1. SELECT column_name(s)
2. FROM table1
3. LEFT  OUTER JOIN table2
4. ON table1.column_name = table2.column_name
5. WHERE condition
6.
7. UNION
8.
9. SELECT column_name(s)
10. FROM table1
11. RIGHT  OUTER JOIN table2
12. ON table1.column_name = table2.column_name
13. WHERE condition
```

Copied!

Union operator

The UNION operator is used to combine the result-set of two or more SELECT statements.

Every SELECT statement within UNION must have the same number of columns
The columns must also have similar data types
The columns in every SELECT statement must also be in the same order

```
1. 1
2. 2
3. 3
```

```
1. SELECT column_name(s) FROM table1
2. UNION
3. SELECT column_name(s) FROM table2;
```

Copied!

How does a SELF JOIN statement syntax look?

```
1. 1
2. 2
3. 3
```

```
1. SELECT column_name(s)
2. FROM table1 T1, table1 T2
3. WHERE condition;
```

Copied!

Exercise

1. Problem:

   Select the names and job start dates of all employees who work for the department number 5.

   ▾ Hint

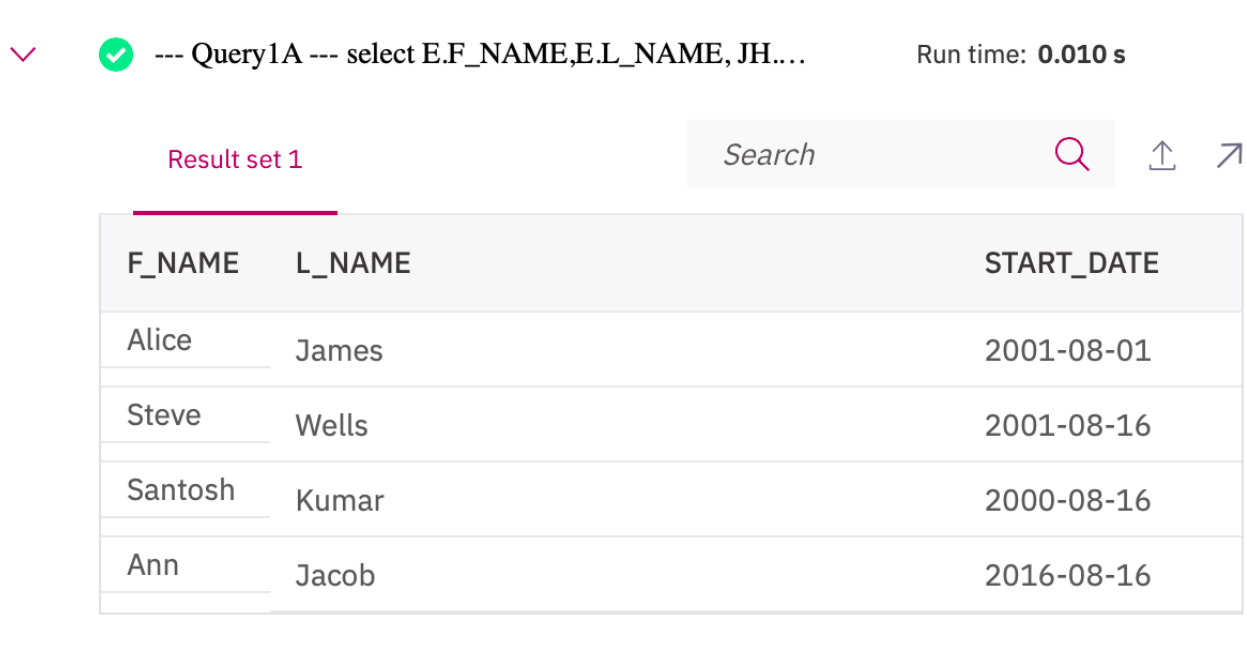
   Use the Inner join operation with the EMPLOYEES table as the left table and the JOB_HISTORY table as the right table.

   ▾ Solution

   ```
   1. 1
   2. 2
   3. 3
   4. 4
   1. select E.F_NAME,E.L_NAME, JH.START_DATE
   2. from EMPLOYEES as E
   3. INNER JOIN JOB_HISTORY as JH on E.EMP_ID=JH.EMPL_ID
   4. where E.DEP_ID ='5';
   ```

   Copied!

▾ Output



F_NAME	L_NAME	START_DATE
Alice	James	2001-08-01
Steve	Wells	2001-08-16
Santosh	Kumar	2000-08-16
Ann	Jacob	2016-08-16

2. Problem:

Select the names, job start dates, and job titles of all employees who work for the department number 5.

▾ Hint

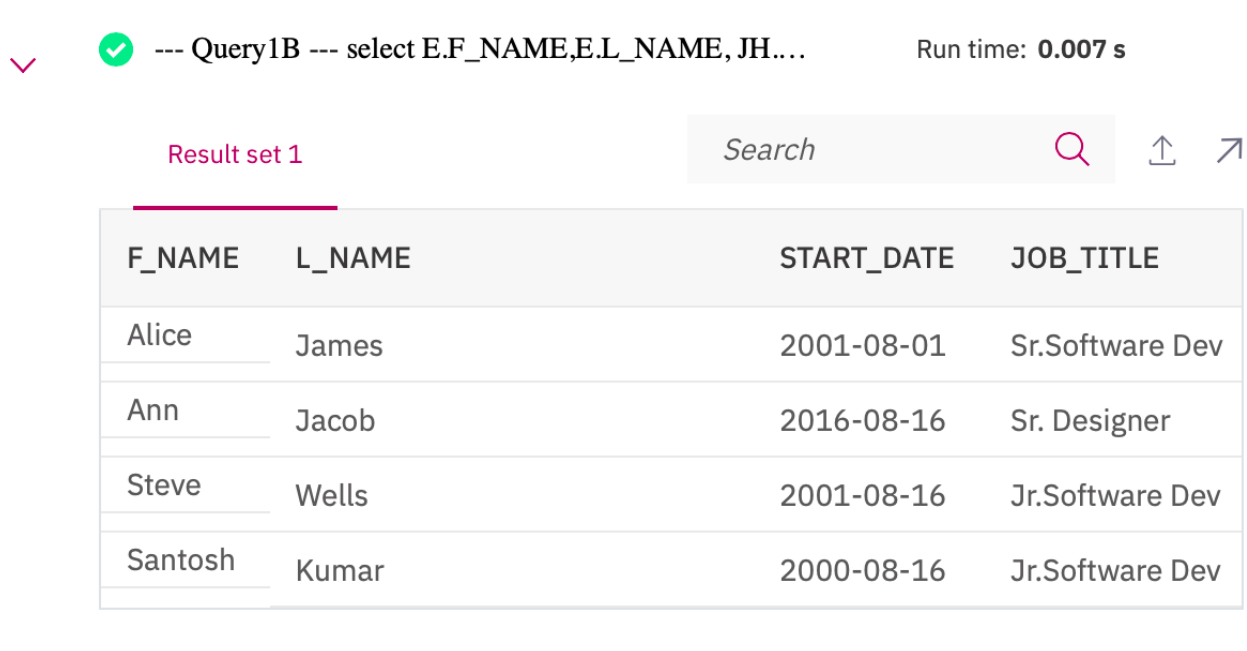
Perform an INNER JOIN with 3 tables EMPLOYEES, JOB_HISTORY, JOBS.

▾ Solution

```
1. 1
2. 2
3. 3
4. 4
5. 5
1. select E.F_NAME,E.L_NAME, JH.START_DATE, J.JOB_TITLE
2. from EMPLOYEES as E
3. INNER JOIN JOB_HISTORY as JH on E.EMP_ID=JH.EMPL_ID
4. INNER JOIN JOBS as J on E.JOB_ID=J.JOB_IDENT
5. where E.DEP_ID ='5';
```

Copied!

▾ Output



F_NAME	L_NAME	START_DATE	JOB_TITLE
Alice	James	2001-08-01	Sr.Software Dev
Ann	Jacob	2016-08-16	Sr. Designer
Steve	Wells	2001-08-16	Jr.Software Dev
Santosh	Kumar	2000-08-16	Jr.Software Dev

3. Problem:

Perform a Left Outer Join on the EMPLOYEES and DEPARTMENT tables and select employee id, last name, department id and department name for all employees.

▾ Hint

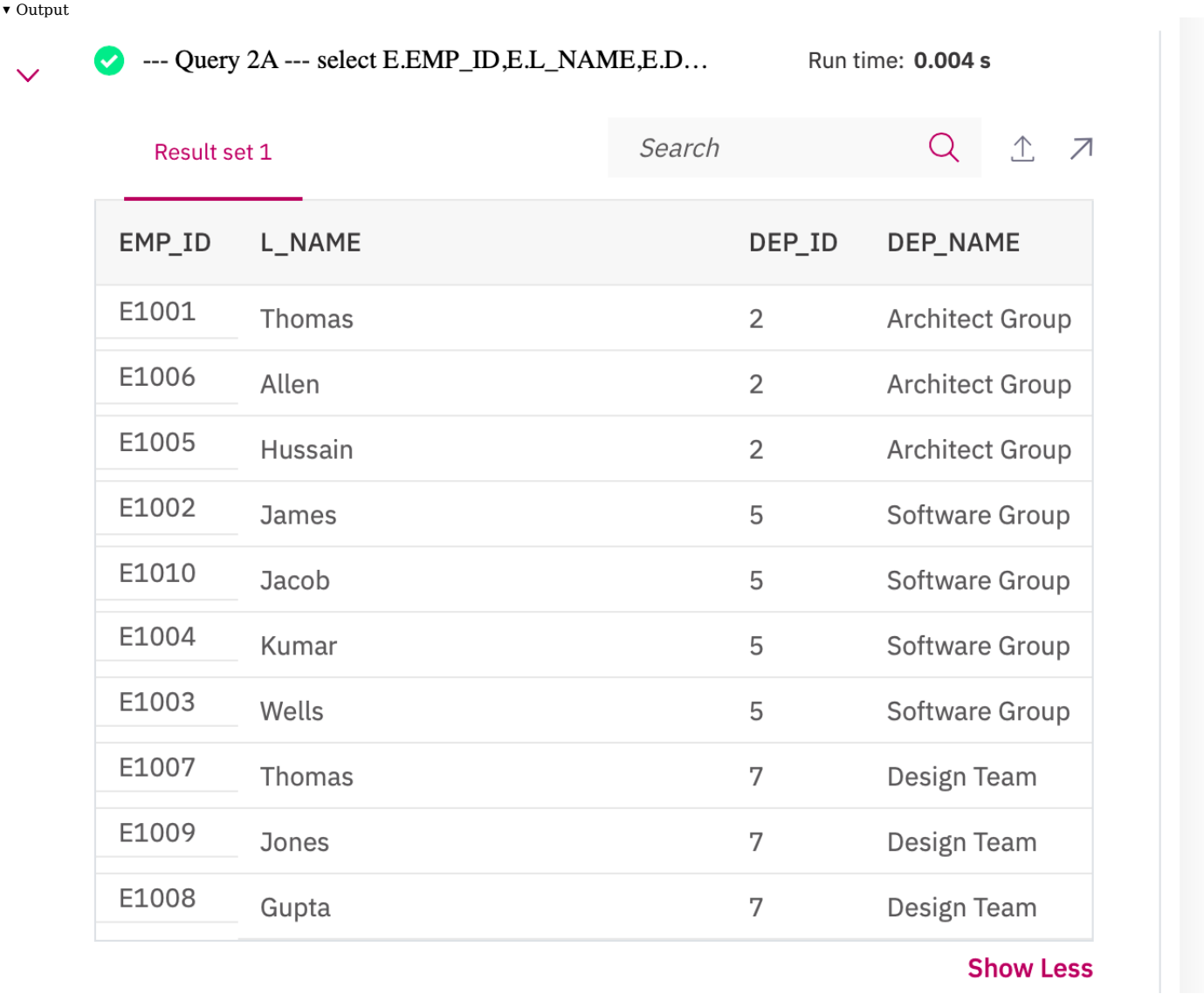
Use the Left Outer Join operation with the EMPLOYEES table as the left table and the DEPARTMENTS table as the right table.

▾ Solution

```
1. 1
2. 2
3. 3
1. select E.EMP_ID,E.L_NAME,E.DEP_ID,D.DEP_NAME
2. from EMPLOYEES AS E
3. LEFT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP;
```

Copied!

▾ Output



EMP_ID	L_NAME	DEP_ID	DEP_NAME
E1001	Thomas	2	Architect Group
E1006	Allen	2	Architect Group
E1005	Hussain	2	Architect Group
E1002	James	5	Software Group
E1010	Jacob	5	Software Group
E1004	Kumar	5	Software Group
E1003	Wells	5	Software Group
E1007	Thomas	7	Design Team
E1009	Jones	7	Design Team
E1008	Gupta	7	Design Team

Show Less

4. Problem:

Re-write the previous query but limit the result set to include only the rows for employees born before 1980.

▾ Hint

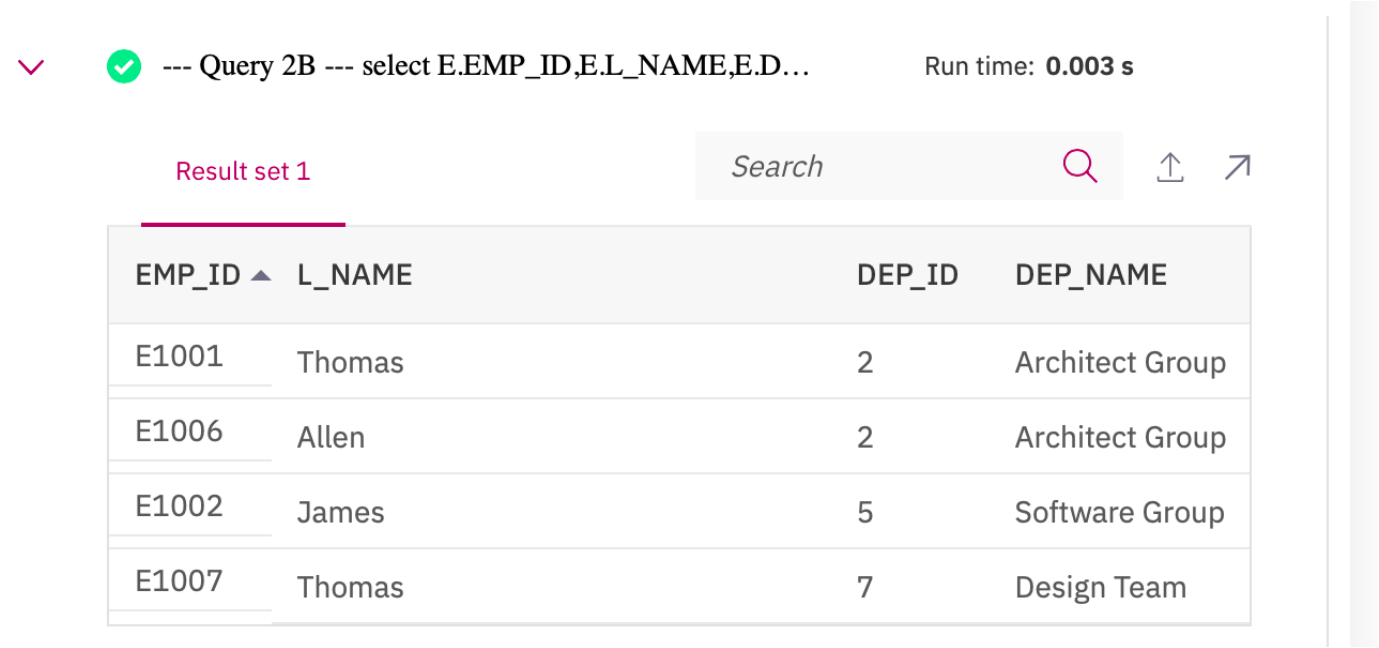
Use a WHERE clause and Left Outer Join operation. Alternatively, you could also use an INNER JOIN.

▾ Solution

```
1. 1
2. 2
3. 3
4. 4
1. select E.EMP_ID,E.L_NAME,E.DEP_ID,D.DEP_NAME
2. from EMPLOYEES AS E
3. LEFT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP
4. where YEAR(E.B_DATE) < 1980;
```

Copied!

▾ Output



EMP_ID	L_NAME	DEP_ID	DEP_NAME
E1001	Thomas	2	Architect Group
E1006	Allen	2	Architect Group
E1002	James	5	Software Group
E1007	Thomas	7	Design Team

5. Problem:

   Re-write the previous query but have the result set include all the employees but department names for only the employees who were born before 1980.

   ▾ Hint

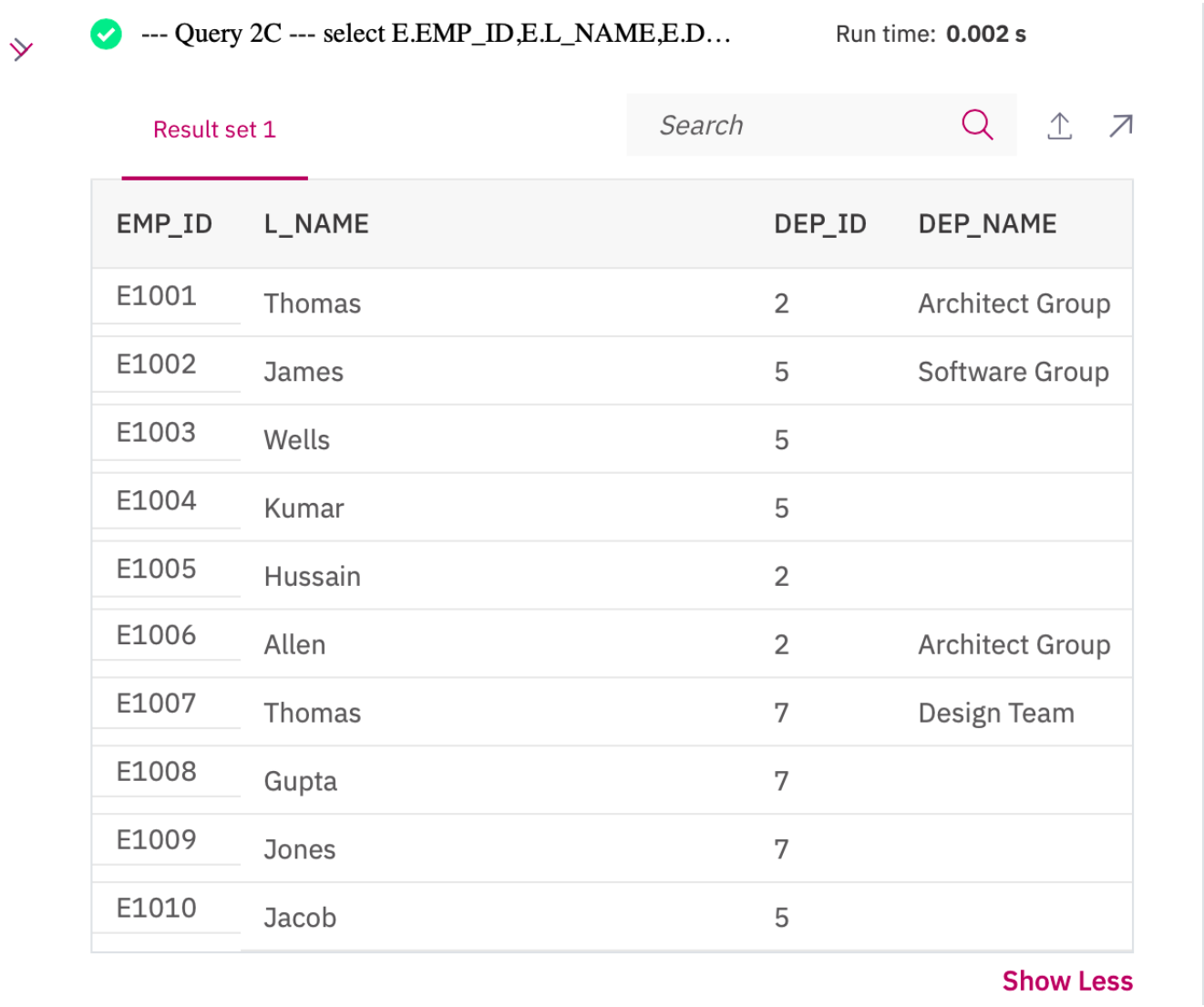
   Use an AND in the LEFT OUTER JOIN clause.

   ▾ Solution

```
1. 1
2. 2
3. 3
4. 4
1. select E.EMP_ID,E.L_NAME,E.DEP_ID,D.DEP_NAME
2. from EMPLOYEES AS E
3. LEFT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP
4. AND YEAR(E.B_DATE) < 1980;
```

Copied!

   ▾ Output



6. Problem:

Perform a Full Join on the EMPLOYEES and DEPARTMENT tables and select the First name, Last name and Department name of all employees.

▾ Hint

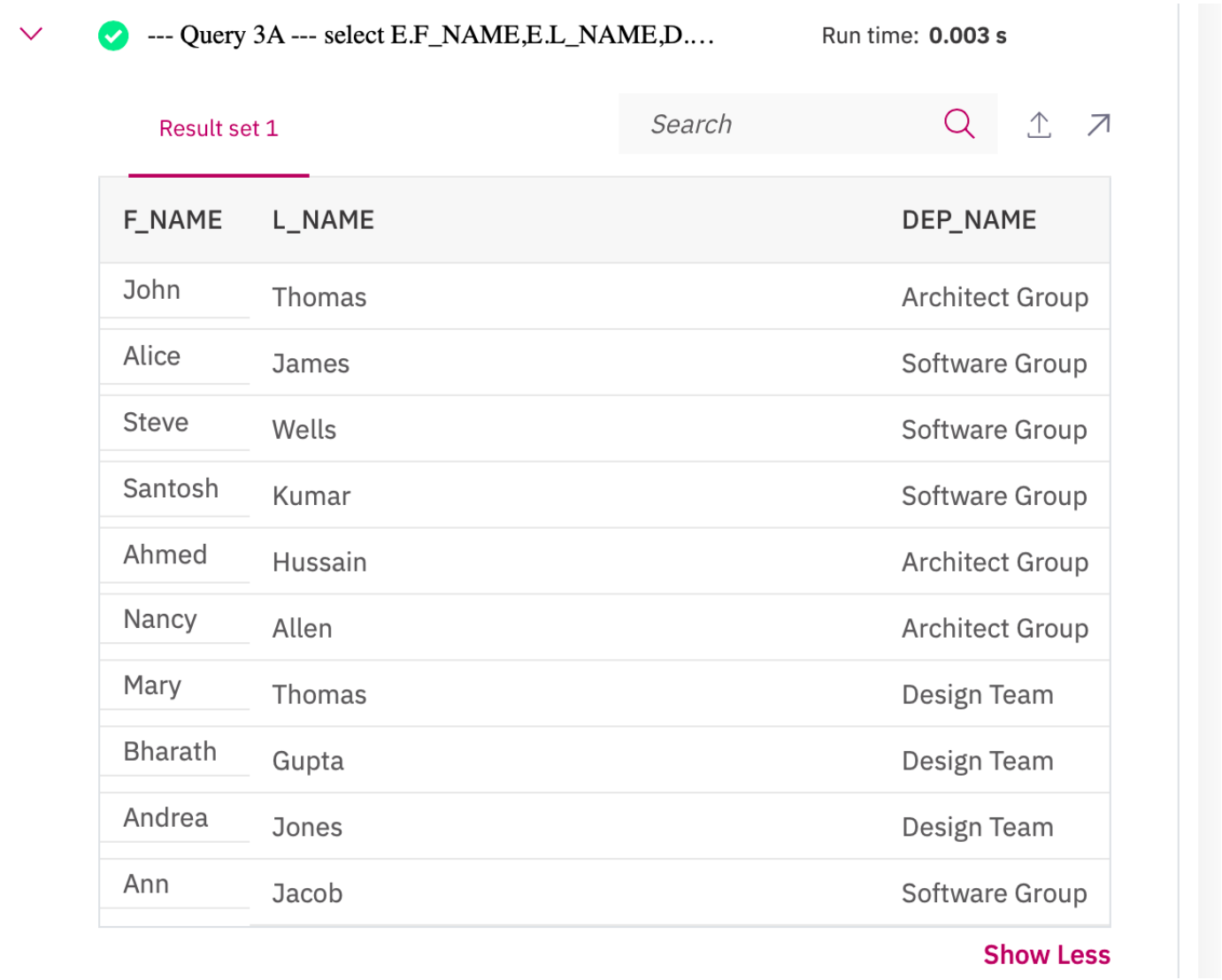
Use the Full Outer Join operation with the EMPLOYEES table as the left table and the DEPARTMENTS table as the right table.

▾ Solution

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
1. select E.F_NAME,E.L_NAME,D.DEP_NAME
2. from EMPLOYEES AS E
3. LEFT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP
4.
5. UNION
6.
7. select E.F_NAME,E.L_NAME,D.DEP_NAME
8. from EMPLOYEES AS E
9. RIGHT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP
```

Copied!

▾ Output



7. Problem:

   Re-write the previous query but have the result set include all employee names but department id and department names only for male employees.

   ▾ Hint

   Add an AND in Query 3A to filter on male employees in the ON clause. Alternatively, you can also use Left Outer Join.

   ▾ Solution

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
1. select E.F_NAME,E.L_NAME,D.DEPT_ID_DEP, D.DEP_NAME
2. from EMPLOYEES AS E
3. LEFT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP AND E.SEX = 'M'
4.
5. UNION
6.
7. select E.F_NAME,E.L_NAME,D.DEPT_ID_DEP, D.DEP_NAME
8. from EMPLOYEES AS E
9. RIGHT OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP AND E.SEX = 'M';
```

   Copied!

   ▾ Output

--- Query 3B --- select E.F_NAME,E.L_NAME,D.... Run time: **0.003 s**

Result set 1 Search

F_NAME	L_NAME	DEPT_ID_DEP	DEP_NAME
John	Thomas	2	Architect Group
Steve	Wells	5	Software Group
Santosh	Kumar	5	Software Group
Ahmed	Hussain	2	Architect Group
Bharath	Gupta	7	Design Team
Alice	James		
Nancy	Allen		
Mary	Thomas		
Andrea	Jones		
Ann	Jacob		

Show Less

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. Follow Hands-on Lab : Create tables using SQL scripts and Load data into tables on how to import a script to mysql phpadmin interface.

- JOIN_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
2023-05-05	0.4	Rahul Jaideep	Updated Markdown file
2022-10-28	0.3	Appalabhaktula Hema	Updated image links
2021-08-09	0.2	Sathya Priya	Updated SQL link
2021-11-01	0.1	Lakshmi Holla, Malika Singla	Initial Version