Joins

We will run through some SQL practice problems that will provide hands-on experience with the different kinds of join operations.

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

```
SELECT column_name(s)
FROM table1
CROSS JOIN table2;
```

How does an INNER JOIN statement syntax look?

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
WHERE condition;
```

How does a LEFT OUTER JOIN statement syntax look?

```
SELECT column_name(s)
FROM table1
LEFT OUTER JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;
```

How does a RIGHT OUTER JOIN statement syntax look?

```
SELECT column_name(s)
FROM table1
RIGHT OUTER JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;
```

How does a FULL OUTER JOIN statement syntax look?

```
SELECT column_name(s)
FROM table1
FULL OUTER JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;
```

How does a SELF JOIN statement syntax look?

```
SELECT column_name(s)
FROM table1 T1, table1 T2
WHERE condition;
```

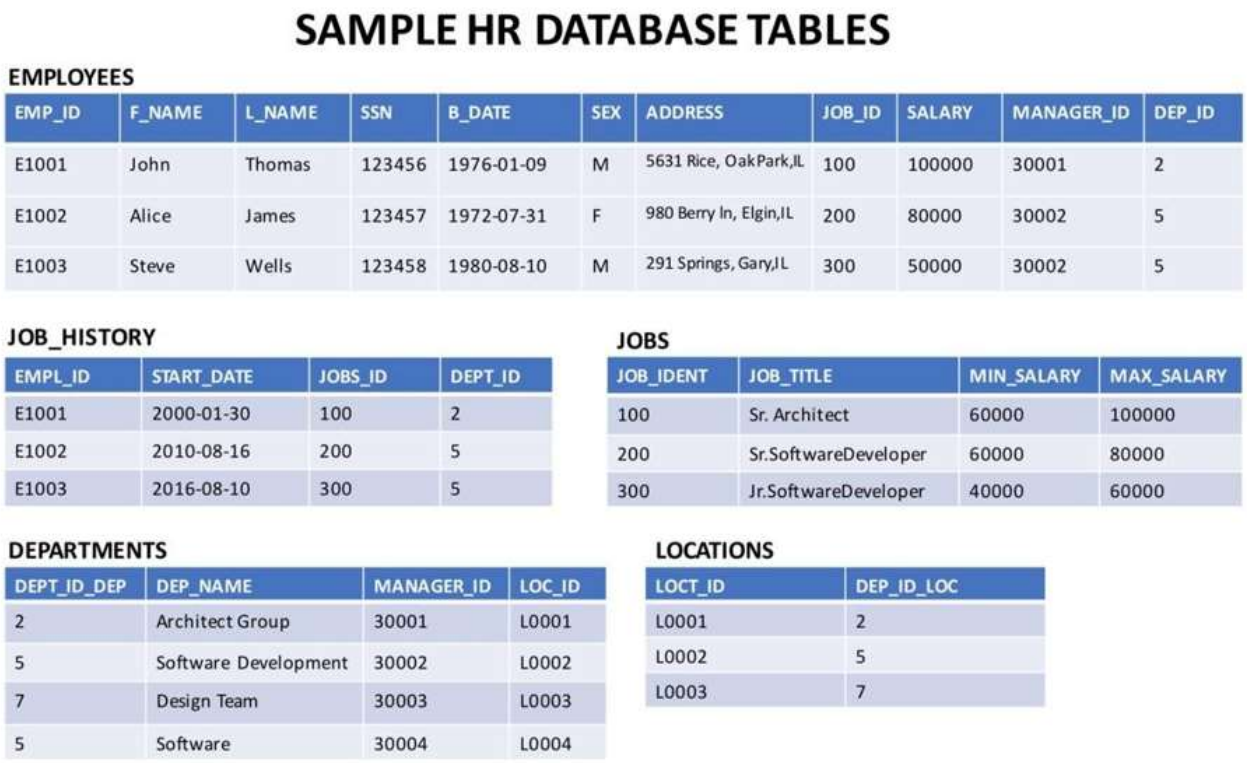
Software Used in this Lab

In this lab, you will use an IBM Db2 Database. Db2 is a Relational Database Management System (RDBMS) from IBM, designed to store, analyze and retrieve data efficiently.

To complete this lab you will utilize a Db2 database service on IBM Cloud.

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

- Performing different kinds of join operations

Instructions

When you approach the exercises in this lab, follow the instructions to run the queries on Db2:

- Go to the Resource List of IBM Cloud by logging in where you can find the Db2 service instance that you created in a previous lab under **Services** section. Click on the **Db2-xx service**. Next, open the Db2 Console by clicking on **Open Console** button. Click on the 3-bar menu icon in the top left corner and go to the **Run SQL** page. The Run SQL tool enables you to run SQL statements.

Exercise

1. Problem:

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

   Solution:

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

   Output:

2. Problem:

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

Solution:

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

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

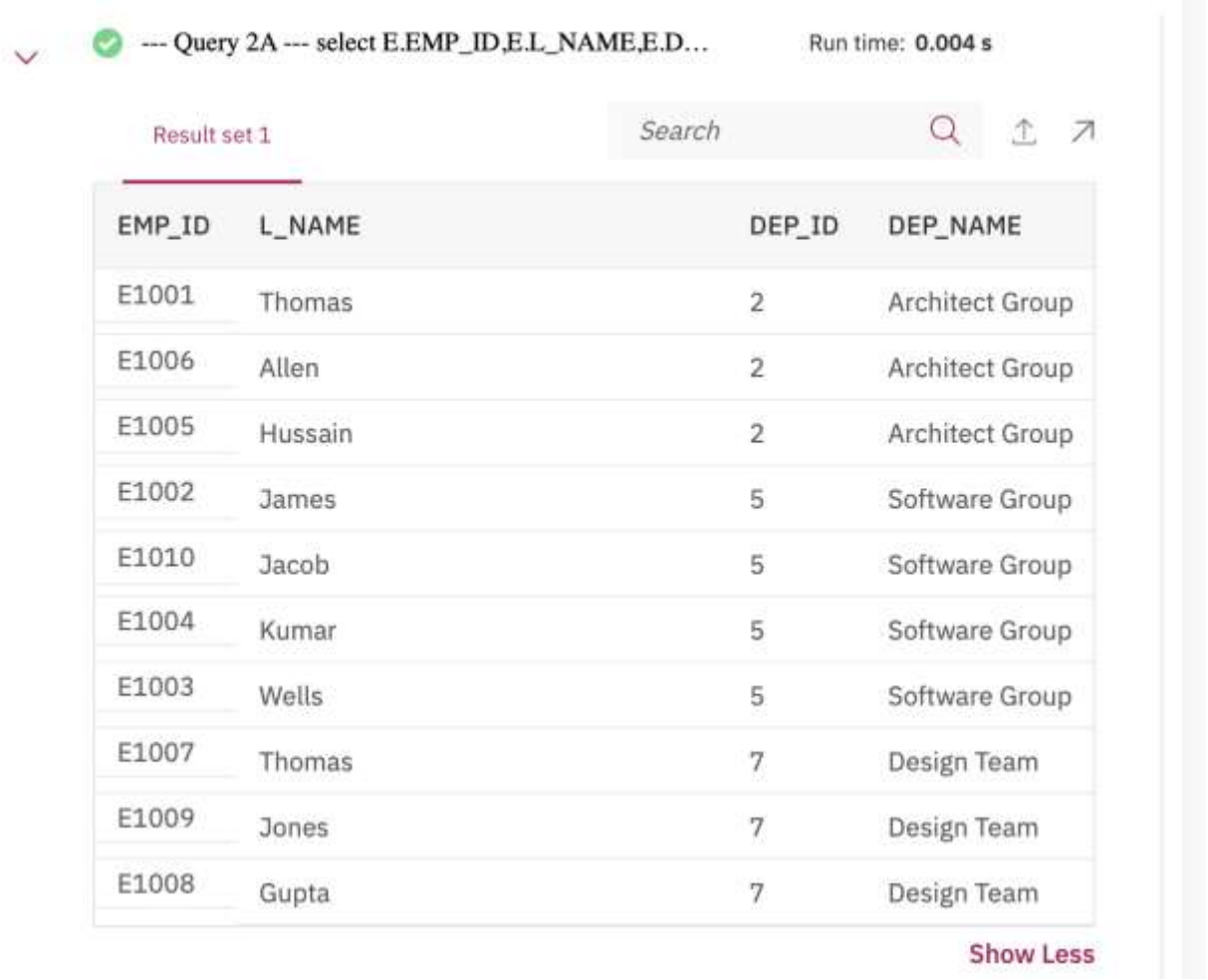
3. Problem:

   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.

   Solution:

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

   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

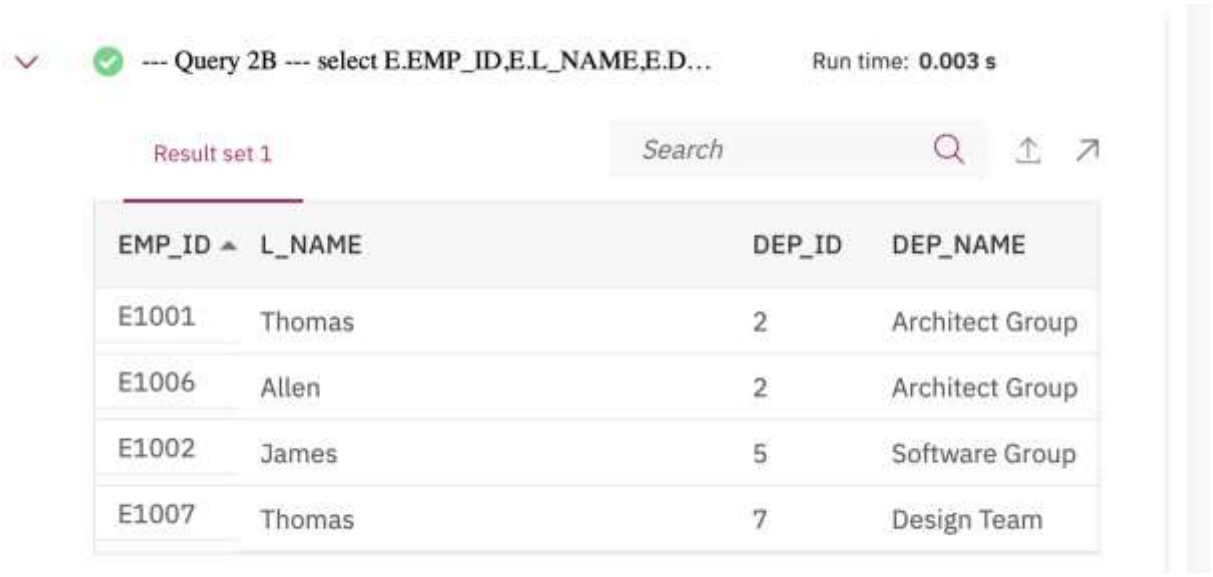
4. Problem:

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

   Solution:

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

   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

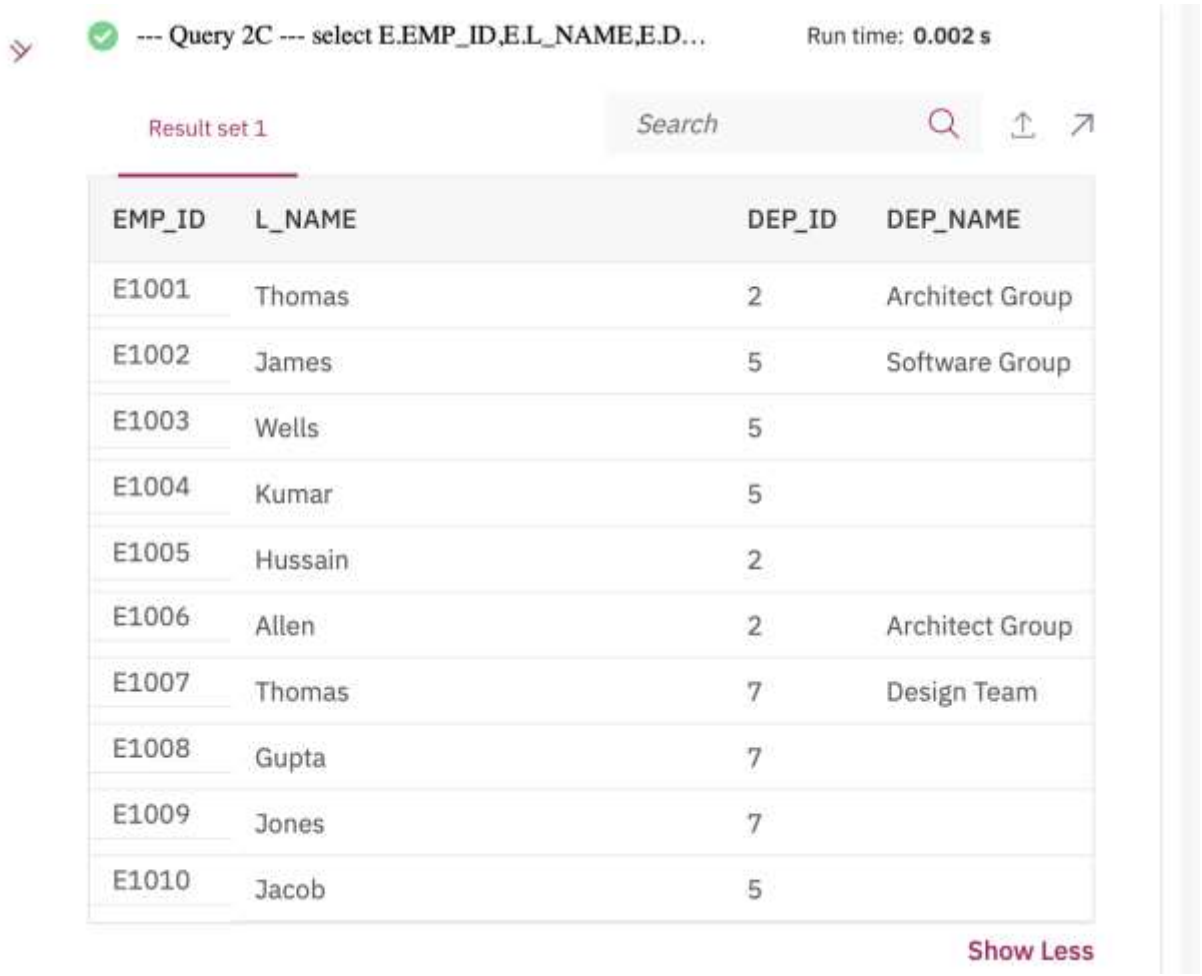
5. Problem:

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.

Solution:

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

Output:



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

Show Less

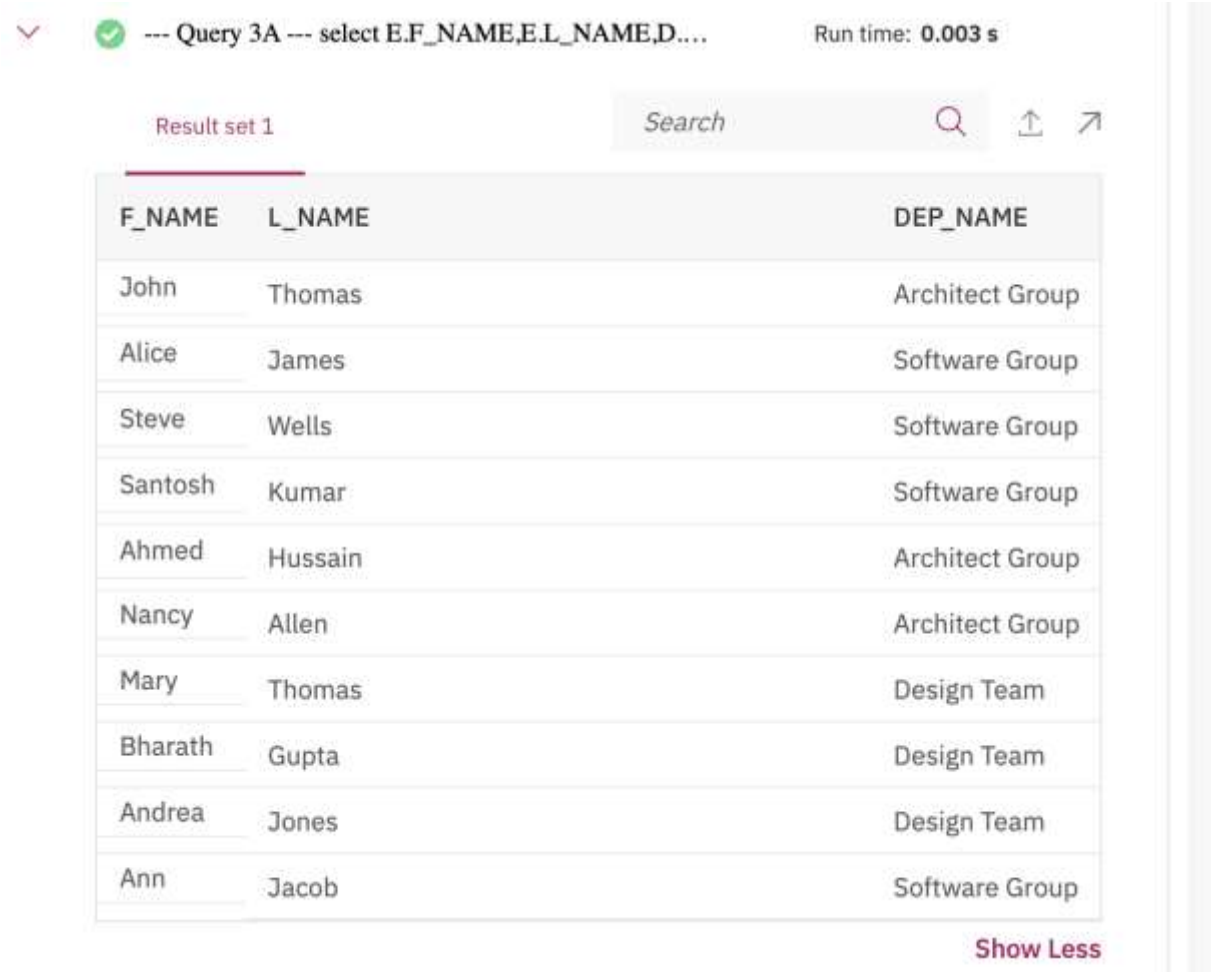
6. Problem:

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

   Solution:

```
select E.F_NAME,E.L_NAME,D.DEP_NAME
from EMPLOYEES AS E
FULL OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP;
```

   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.

   Solution:

```
select E.F_NAME,E.L_NAME,D.DEPT_ID_DEP, D.DEP_NAME
from EMPLOYEES AS E
FULL OUTER JOIN DEPARTMENTS AS D ON E.DEP_ID=D.DEPT_ID_DEP AND E.SEX = 'M';
```

   Output: