



Hands-on Lab: Using Views

Estimated time needed: 10 minutes

In this lab, you will learn about using views. In SQL, a view is an alternative way of representing data that exists in one or more tables. Just like a real table, it contains rows and columns. The fields in a view are fields from one or more real tables in the database. Though views can be queried like a table, views are dynamic; only the definition of the view is stored, not the data.

How does the syntax of a CREATE VIEW statement look?

```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```



How does the syntax of a REPLACE VIEW statement look?

```
CREATE OR REPLACE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```



How does the syntax of a DROP VIEW statement look?

```
DROP VIEW view_name;
```



Software Used in this Lab

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

To complete this lab you will utilize a Db2 database service on IBM Cloud. If you did not already complete this lab task earlier in this module, you will not yet have access to Db2 on IBM Cloud, and you will need to follow the lab below first:

- [Hands-on Lab : Sign up for IBM Cloud, Create Db2 service instance and Get started with the Db2 console](#)

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, Oak Park, 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_ID | JOB_TITLE | MIN_SALARY | MAX_SALARY |
|--------|------------------------|------------|------------|
| 100 | Sr. Architect | 60000 | 100000 |
| 200 | Sr. Software Developer | 60000 | 80000 |
| 300 | Jr. Software Developer | 40000 | 60000 |

DEPARTMENTS

| DEPT_ID | DEPT_NAME | MANAGER_ID | LOC_ID |
|---------|----------------------|------------|--------|
| 2 | Architect Group | 30001 | L0001 |
| 5 | Software Development | 30002 | L0002 |
| 7 | Design Team | 30003 | L0003 |
| 5 | Software | 30004 | L0004 |

LOCATIONS

| LOC_ID | DEPT_ID | LOC |
|--------|---------|-----|
| L0001 | 2 | |
| L0002 | 5 | |
| L0003 | 7 | |

NOTE: This lab requires you to have all 5 of these tables of the HR database populated with sample data on Db2. If you don't have the tables above populated with sample data on Db2, please go through the lab below first:

- [Hands-on Lab : Create tables using SQL scripts and Load data into tables](#)

Objectives

After completing this lab, you will be able to:

- Create a View and show a selection of data for a given table
- Update a View to combine two or more tables in meaningful ways
- Drop a created View

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.
 - If needed, follow [Hands-on Lab : Sign up for IBM Cloud, Create Db2 service instance and Get started with the Db2 console](#)

Exercise 1: Create a View

In this exercise, you will create a View and show a selection of data for a given table.

1. Let's create a view called **EMPSALARY** to display salary along with some basic sensitive data of employees from the HR database. To create the **EMPSALARY** view from the **EMPLOYEES** table, copy the code below and paste it to the textbox of the **Run SQL** page. Click **Run all**.

```
CREATE VIEW EMPSALARY AS
SELECT EMP_ID, F_NAME, L_NAME, B_DATE, SEX, SALARY
FROM EMPLOYEES;
```



```
CREATE OR REPLACE VIEW EMP_SALARY AS
SELECT EMP_ID, F_NAME, L_NAME, B_DATE, SEX, JOB_TITLE, MIN_SALARY, MAX_SALARY
FROM EMPLOYEES, JOBS
WHERE EMPLOYEES.JOB_ID = JOBS.JOB_IDENT;
```

NOTE: Don't worry if you don't understand how to combine two tables using implicit inner join. You will learn more about joins later on. For now, just think you are combining the data of two different tables, **EMPLOYEES** and **JOBS** by connecting their respective columns **JOB_ID** and **JOB_IDENT** since both the columns contain common unique data. You can have a look at the diagram (at the beginning of the lab) showing the tables for the HR database to observe how the **JOB_ID** and **JOB_IDENT** columns from the **EMPLOYEES** and **JOBS** tables respectively contain common unique data.

```
1 CREATE OR REPLACE VIEW EMP_SALARY AS
2 SELECT EMP_ID, F_NAME, L_NAME, B_DATE, SEX, JOB_TITLE, MIN_SALARY, MAX_SALARY
3 FROM EMPLOYEES, JOBS
4 WHERE EMPLOYEES.JOB_ID = JOBS.JOB_ID;
```

Result - Dec 14, 2020 11...
✓ CREATE OR REPLACE VIEW EMP_SALARY AS ... Run time: 0.022 s
Status: Success | Affected Rows: 0

SELECT * FROM EMP\$SALARY;

Result set 1

Search

| EMP_ID | F_NAME | L_NAME | B_DATE | SEX | JOB_TITLE | MIN_SALARY | MAX_SALARY |
|--------|---------|---------|------------|-----|-----------------|------------|------------|
| E1001 | John | Thomas | 1976-01-09 | M | Sr. Architect | 60000.00 | 100000.00 |
| E1002 | Alice | James | 1972-07-31 | F | Sr.Software Dev | 60000.00 | 80000.00 |
| E1003 | Steve | Wells | 1980-08-10 | M | Jr.Software Dev | 40000.00 | 60000.00 |
| E1004 | Santosh | Kumar | 1985-07-20 | M | Jr.Software Dev | 40000.00 | 60000.00 |
| E1005 | Ahmed | Hussain | 1981-01-04 | M | Jr. Architect | 50000.00 | 70000.00 |
| E1006 | Nancy | Allen | 1978-02-06 | F | Lead Architect | 70000.00 | 100000.00 |
| E1007 | Mary | Thomas | 1975-05-05 | F | Jr. Designer | 60000.00 | 70000.00 |
| E1008 | Bharath | Gupta | 1985-05-06 | M | Jr. Designer | 60000.00 | 70000.00 |
| E1009 | Andrea | Jones | 1990-07-09 | F | Sr. Designer | 70000.00 | 90000.00 |
| E1010 | Ann | Jacob | 1982-03-30 | F | Sr. Designer | 70000.00 | 90000.00 |

In this exercise, you will drop a created View.

The screenshot shows the SQL editor interface after executing the command. The top bar displays the executed query: "DROP VIEW EMPLOYEES;". Below the editor, the results pane shows a single row with a green checkmark icon, indicating success. The message "Result - Dec 14, 2020 11..." is visible, along with the run time "Run time: 0.010 s". A status bar at the bottom indicates "Status: Success | Affected Rows: 0".

SELECT * FROM EMP_SALARY;

1

2


SELECT * FROM EMP_SALARY;

Result - Dec 14, 2020 11...

▼

⋮

✕

✓  SELECT * FROM EMP_SALARY

Run time: 0.008 s

Status: **Failed**

Error message

"ZJH17769.EMP_SALARY" is an undefined name.. SQLCODE=-204, SQLSTATE=42704, DRIVER=4.26.14

[Learn more about this error](#)

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

Author(s)

- [Sandip Saha Joy](#)

Other Contributor(s)

-

Changelog

| Date | Version | Changed by | Change Description |
|------------|---------|-----------------|-------------------------|
| 2020-12-24 | 1.1 | Steve Ryan | ID reviewed |
| 2020-12-14 | 1.0 | Sandip Saha Joy | Created initial version |

© IBM Corporation 2020. All rights reserved.