Final Assignment - Database Administration - Part 3

Estimated time needed: **45** minutes.

About This SN Labs Cloud IDE

This Skills Network Labs Cloud IDE provides a hands-on environment for course and project related labs. It utilizes Theia, an open-source IDE (Integrated Development Environment) platform, that can be run on desktop or on the cloud. To complete this lab, we will be using the Cloud IDE based on Theia running in a Docker container. You will also need an instance of DB2 running in IBM Cloud.

Important Notice about this lab environment

Please be aware that sessions for this lab environment are not persisted. Every time you connect to this lab, a new environment is created for you. Any data you may have saved in the earlier session would get lost. Plan to complete these labs in a single session, to avoid losing your data.

Scenario

You have been assigned the work to provison a cloud instance of IBM DB2 server and perform the tasks like restoration of data, index creation to improve the query performance. You will create views to make queries easier to write. Optionally You will also connect to the cloud instance of IBM DB2 server and from command line.

Objectives

In part 3 of this assignment you will be working on the following aspects of Database Administration.

- Restore data
- Indexing
- View creation
- Connecting from command line

Note - Screenshots

Throughout this lab you will be prompted to take screenshots and save them on your own device. These screenshots will be needed to be uploaded for peer review in the next section of the course. You can use various free screengrabbing tools to do this or use your operating system's shortcut keys to do this (for example *Alt+PrintScreen* in Windows).

Exercise 3.1 - Prepare the lab environment

Before you proceed with the assignment, you need to have access to a cloud instance of IBM DB2 database. If you do not have access, use the instructions in this lab [Hands-on Lab: Sign up for IBM Cloud and Create a Db2 service instance](#) to create a instance for yourself.

Download the file [billing.csv](#)

Exercise 3.2 - Restore data

Task 3.1 - Restore the table billing

Use the **billing.csv** you have downloaded earlier, restore the csv file into a table named billing.

Note: You will see that each column has data type and column width auto generated based on the content. Edit column attributes by clicking on the pencil icon next to the respective attributes to change the width of country column to varchar of 30 and month column to varchar of 7.

Take a screenshot of the import status clearly showing the number of rows imported.

Name the screenshot as **restore-table.jpg**. (images can be saved with either .jpg or .png extension)

Exercise 3.3 - Create a view

Task 3.2 - Create a view named `basicbilldetails` with the columns customerid, month, billedamount

Take a screenshot of the sql statement used to create the view.

Name the screenshot as **create-view.jpg**. (images can be saved with either .jpg or .png extension)

Exercise 3.4 - Indexing

Task 3.3 - Baseline query performance

Write a query to find out all the rows with a billing amount of 19929.

Take a screenshot of the command you used along with the query run time.

Name the screenshot as **query-base-line-db2.jpg**. (images can be saved with either .jpg or .png extension)

Task 3.4 - Create an index

Create an index that can make the query in the previous task faster. Name the index as `billingamount`.

Take a screenshot of the sql statement you used and the output.

Name the screenshot as **index-creation-db2.jpg**. (images can be saved with either .jpg or .png extension)

Task 3.5 - Document the improvement in query performance

Find out if the index has any impact on query performance.

Re-run the query to find out all the rows with a billing amount of 19929.

Take a screenshot of the command you used and the output along with the query time.

Note: Sometimes, the query time after index creation may increase. This can happen due to various factors like

- bandwidth at the time of firing the query
- the load on free cloud tier that your IBM DB2 instance uses is dynamic and other load may impact your query time

You will NOT be evaluated on the query run time. However, you are encouraged to run the query multiple times and pick the result with lowest query run time.

Name the screenshot as **query-after-index.jpg**. (images can be saved with either .jpg or .png extension)

OPTIONAL Exercise (Non-graded) - Connecting to IBM DB2 from command line

Bonus Task 3.6 - Connect to the clould instance of IBM DB2 using the db2cli command line tool

From the theia lab environment, you can use the db2cli to connect the IBM DB2. The generic syntax to connect is as follows.

db2cli execsql -connString "DATABASE=<databasename>;UID=<username>;PWD=<password>;HOSTNAME=<hostname>;port=<port>;Security=SSL"

Collect the databasename, username, password, hostname and port details using the service credentials of the IBM DB2 cloud console.

Run a sql query on the db2cli command line to find out the number of rows in the table billing.

Note the row count.

End of assignment - Part 3.

Authors

Ramesh Sannareddy

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2021-10-01	0.1	Ramesh Sannareddy	Created initial version
2021-10-19	0.2	Alison Woolford	Updated after review
2021-10-22	0.3	Steve Hord	QA pass
2023-05-04	0.4	Rahul Jaideep	Updated Markdown file

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2021-10-01	0.1	Ramesh Sannareddy	Created initial version