Name	**:**	G.Jothivel
College Code	**:**	9530
College Name	**:**	St.Mother Theresa Engineering College
Team id	**:**	
Nan Mudhalvan Id	**:**	au953021104021
Project Name	**:**	Big Data Analysis with IBM cloud databases

Problem Solution:

Big Data Analysis with IBM Cloud Databases

Objectives

- Provision a SQL database
- Create the database schema (table) and load data
- Deploy a pre-built containerized app to Code Engine
- Connect the app and database service (share credentials)
- Monitor, Secure, Backup & Recovery of cloud databases

Step 1: Provision the SQL Database

Start by creating an instance of the **Db2 Warehouse on Cloud** service.

1. Visit the IBM Cloud® console. Click on **Catalog** in the top navigation bar.
2. Click on **Databases** on the left pane and select **Db2 Warehouse**.
3. Pick the **Flex One** plan and change the suggested service name to **sqldatabase** (you will use that name later on). Pick a resource group and a location for the deployment of the database.
4. Click on **Create**. The provisioning starts.
5. In the **Resource List**, locate the new instance under **Databases** and wait for it to be available (sometimes you may need to refresh the page). Click on the entry for our Db2 Warehouse on Cloud service.
6. Click on **Open Console** to launch the database console.

Step 2: Create a table

1. In the console for Db2 Warehouse on Cloud click on the upper left menu icon, then **Run SQL** in the navigation bar.
2. Click on the + symbol (**Add a new script**) next to the **Untitled - 1** tab.
3. Click on **From file** and select the file `cityschema.txt` from the GitHub repository that was previously cloned to our local directory and open it.
4. Click on **Run all** to execute the statement. It should show a success message.

Step 3: Load data

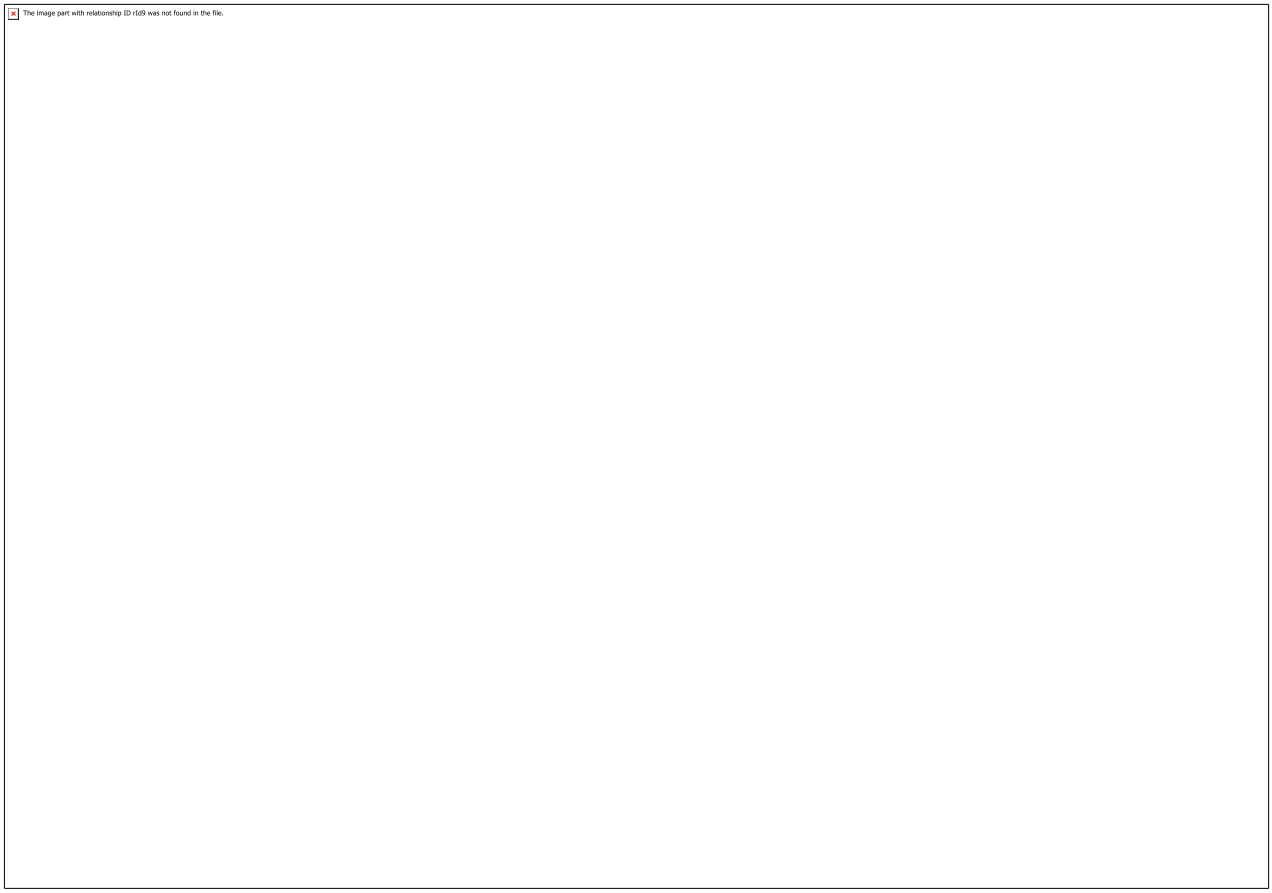
Now that the table "cities" has been created, you are going to load data into it. This can be done in different ways, for example from our local machine or from cloud object storage (COS) or Amazon S3 interface. We are going to upload data from our machine. During that process, you adapt the table structure and data format to fully match the file content.

1. In the console for Db2 Warehouse on Cloud click on the upper left menu icon, then **Data** in the navigation bar.
2. As **Source** keep the selection on **My Computer**.
3. Under **File selection**, click on **Drag a file here or browse files** to locate and pick the file "cities1000.txt" you downloaded in the first section of this guide.
4. Click **Next** to get to the **Target** overview with a **Schema** selection. Choose the schema **BLUADMIN**, then the table **CITIES**. Click on **Next** again.

   Because the table is empty it does not make a difference to either append to or overwrite existing data.

5. Now customize how the data from the file "cities1000.txt" is interpreted during the load process. First, disable **Header in first row** because the file contains data only.
6. Next, type in **0x09** as separator. It means that values within the file are delimited by tab(ulator).
7. Last, pick "YYYY-MM-DD" as date format. Now, everything should look similar to what is shown in this screen capture.

   Zoom



Screen capture showing the sampled data

8. Click **Next** and you are offered to review the load settings. Agree and click **Begin Load** to start loading the data into the **CITIES** table. The progress is displayed. Once the data is uploaded it should only take few seconds until the load is finished and some statistics are presented.
9. Click on **View Table** to browse the data. You may scroll down or click on column names to change the sort order.

Step 4: Verify Loaded Data Using SQL

The data has been loaded into the relational database. There were no errors, but you should run some quick tests anyway. Use the built-in SQL editor to type in and execute some SQL statements.

1. In the left navigation click on **Run SQL** to get back to the SQL editor. Click on the + symbol (**Add new script**) and **Create new** to create a new editor tab.

   Instead of the built-in SQL editor you can use cloud-based and traditional SQL tools on our desktop or server machine with Db2 Warehouse on Cloud. The connection information can be found in the **Administration** menu in the left navigation.

2. In the editor type or copy in the following query:

```
3.select count(*) from cities;
```

   Select the text of the query, then, in dropdown next to **Run All**, choose **Run selected**. In the section with results, the same number of rows as reported by the load process should be shown.

4. In the "SQL Editor" enter the following statement on a new line:

```
5.select countrycode, count(name) from cities
6.group by countrycode
7.order by 2 desc;
```

   Mark the text of the above statement and click the **Run selected** button. Only this statement is executed, returning some by country statistics in the results section.

8. Finally, run the following statement similarly to retrieve details about San Francisco in California:

9. select*from cities
10. where name='San Francisco'
11. andcountrycode='US';

Step 5: Deploy the application code

Change back to the terminal. Now you are going to deploy the application code, using a pre-built container image. We can modify the application code and build the container image on our own. If you are not logged in, use `ibmcloud login` or `ibmcloud login --sso` to log in interactively. Setthe region and resource group to where the database has been provisioned. Replace **RESOURCE_GROUP** and **REGION** accordingly.

1. ibmcloud target -g RESOURCE_GROUP -r REGION
2. Create a new Code Engine project named **sqldatabase**:

3. ibmcloudce project create --name sqldatabase

   Select the new project as the active one:
   ibmcloudce project select --name sqldatabase

4. Then, deploy the app naming it **worldcities**.
5. ibmcloudce app create
6. Last, create a service binding between the existing Db2 Warehouse on Cloud database and the app:
7. ibmcloudce application bind --name worldcities --service-instance sqldatabaseOnce the binding is created, a new app revision is started.
8. ibmcloudce app get --name worldcities

Step 6: Security, Backup & Recovery, Monitoring

The Db2 Warehouse on Cloud is a managed service. IBM takes care of securing the environment, daily backups and system monitoring. When you are using one of the enterprise plans there are several options to manage access and to configure enhanced data encryption.

In addition to the traditional administration options the Db2 Warehouse on Cloud service also offers a REST API for monitoring, user management, utilities, load, storage access and more.

Step 7: Test the App

The app to display city information based on the loaded data set is reduced to a minimum .It offers a search form to specify a city name - names are case sensitive - and few preconfigured cities. They are translated to either `/search?name=cityname` (search form) or `/city/cityname` (directly specified cities). Both requests are served from the same lines of code in the background. The `cityname` is passed as value to a prepared SQL statement using a parameter marker for security reasons. The rows are fetched from the database and passed to an HTML template for rendering.

Step 8: Cleanup

To clean up resources follow these steps:

1. Visit the IBM Cloud® Resource List.
2. In the Code Engine section locate the project **sqldatabase**. Click on the three dots and select **Delete** to delete the project and its app.
3. Locate the database `sqldatabase` under **Databases**. Again, click on the three dots and select **Delete** to delete the database.
4. Depending on the resource it might not be deleted immediately, but retained (by default for 7 days). You can reclaim the resource by deleting it permanently or restore it within the retention period.

Github Link: