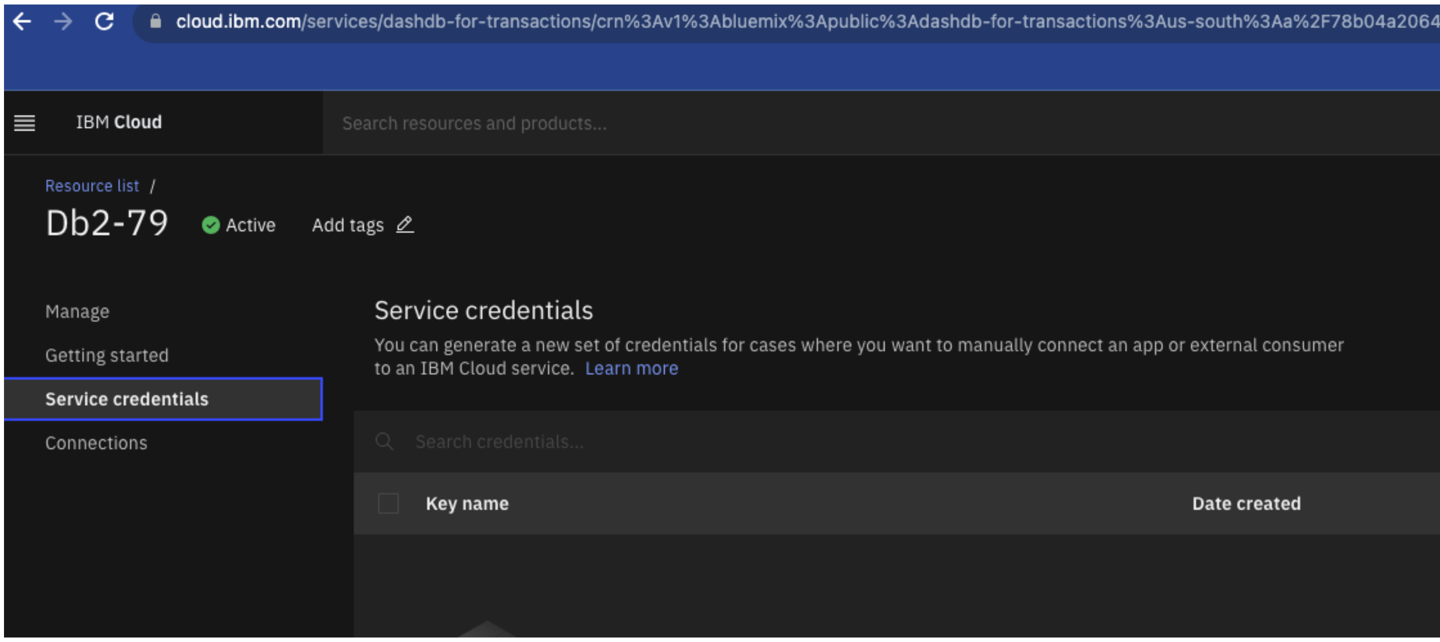
Populating IBM - DB2 Through CLI

Gather Service / Db2 Credentials


cloud.ibm.com/services/dashdb-for-transactions/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F78b04a2064
IBM Cloud
Search resources and products...
Resource list /
Db2-79
Active
Add tags
Manage
Getting started
Service credentials
Connections
Search credentials...
Key name
Date created

Service credentials

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. Learn more

- Accessing IBM DB2 Instance and hitting up the Service Credentials
- Created New Credentials and now the service instance has a length JSON object for a created service credential details to use later down the line

Exercise 3 - Create a db2cli dsn

You can access the **IBM DB2 cloud instance** using the web browser user interface.

Using the **db2cli** you can access your cloud IBM DB2 instance from the command line.

- db2cli can be very helpful in automating data load tasks.

In this exercise we will be creating a dsn (data source name). A **dsn** in short is a simple name that refers to a data source.

Creating a dsn is two step process.

Step 1: We add the database, host, port and the security mode details. A sample commmand is given for your reference below:

```
db2cli writecfg add -database dbname -host hostname -port 50001 -parameter
"SecurityTransportMode=SSL"
```

Step 2: We give a name to the data source we just created. This dsn name helps us to easily point to the IBM DB2 instance. A sample commmand is given for your reference below.

```
db2cli writecfg add -dsn dsn_name -database dbname -host hostname -port
50001
```

- Updating the host address in this document just to … (probably unnecessarily to limit any bad actors .. lol)

```
 "database": "bludb",
      "host_ros": [

"125f9f61-9715-46f9-9399-c8177b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cl
oud:31048"
      ],
      "hosts": [
        {
          "hostname":
"125f9f61-9715-46f9-99-c8177b21803b.c1ogj3sd0lqde00.databases.appdomain.clo
ud",
          "port": 30426
        }
      ],
      "jdbc_url": [

"jdbc:db2://125f9f61-9715-46f9-9399-c81773b.c1ogj3sd0tgtu0lqde00.databases.
appdomain.cloud:30426/bludb:user=<userid>;password=<your_password>;sslConne
ction=true;"
      ],
      "path": "/bludb",
      "query_options": {
        "authSource": "admin",
        "replicaSet": "replset"
      },
      "replica_set": "replset",
      "scheme": "db2",
      "type": "uri"
    }
```

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli writecfg add -database
bludb -host
129f61-9715-46f9-9399-c81703b.c1ogj3sd0tgtu0lqde00.databases.appdomain.clou
d -port 304 -parameter "SecurityTransportMode=SSL"

===========================================================================
====
db2cli writecfg completed successfully.
===========================================================================
====

theia@theiadocker-craigtrupp8:/home/project$ db2cli writecfg add -dsn
production -database bludb -host
19f61-9715-46f9-9399-c8121803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.clo
ud -port 3042
===========================================================================
====
db2cli writecfg completed successfully.
===========================================================================
====
```

Exercise 4 - Verify a db2cli dsn

Now that the dsn is created, we need to verify if it is working, before we go ahead and start using it.

The generic syntax for the command to verify the dsn is given below:

```
db2cli validate -dsn alias -connect -user userid -passwd password
```

- Similarly truncating some of the values in the command and output from the shell session

- The “cli” object in the connection … object for the db2 service service credentials is what was passed here to get the deets and output below

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli validate -dsn
production -connect -user cds032 -passwd jH9BvoJUg

============================================================================
====
Client information for the current copy:
============================================================================
====

Client Package Type       : IBM Data Server Driver Package
Client Version (level/bit): DB2 v11.5.4.0 (s2006161200/64-bit)
Client Platform           : Linux/X8664
Install/Instance Path     : /home/theia/dsdriver
DB2DSDRIVER_CFG_PATH value: <not-set>
db2dsdriver.cfg Path      : /home/theia/dsdriver/cfg/db2dsdriver.cfg
DB2CLIINIPATH value       : <not-set>
db2cli.ini Path           : /home/theia/dsdriver/cfg/db2cli.ini
db2diag.log Path          : /home/theia/dsdriver/db2dump/db2diag.log

============================================================================
====
db2dsdriver.cfg schema validation for the entire file:
============================================================================
====

Success: The system db2dsdriver.cfg schema validation completed
successfully without any errors.

============================================================================
====
db2cli.ini validation for data source name "production":
============================================================================
====

Note: The validation utility could not find the configuration file
db2cli.ini.
The file is searched at "/home/theia/dsdriver/cfg/db2cli.ini".
```

```
===========================================================================
====
db2dsdriver.cfg validation for data source name "production":
===========================================================================
====

[ Parameters used for the connection ]

Keywords                  Valid For     Value
---------------------------------------------------------------------------
DATABASE                  CLI,.NET,ESQL bludb
HOSTNAME                  CLI,.NET,ESQL
125f61-9715-46f9-9399-c817803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.clo
ud
PORT                      CLI,.NET,ESQL 304
SECURITYTRANSPORTMODE     CLI,.NET      SSL

===========================================================================
====
Connection attempt for data source name "production":
===========================================================================
====

[SUCCESS]

===========================================================================
====
The validation is completed.
===========================================================================
====

theia@theiadocker-craigtrupp8:/home/project$
```

- Your dsn is validated. You can now use it to access the IBM DB2 cloud instance.

Exercise 5 - Create the schema on production data warehouse

Step 1: Download the schema file.

```
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/star-schema.sql
--2023-10-06 12:08:49--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/star-schema.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 845 [application/x-sql]
Saving to: 'star-schema.sql'

star-schema.sql
100%[===============================================================>]
845  --.-KB/s    in 0s

2023-10-06 12:08:49 (87.2 MB/s) - 'star-schema.sql' saved [845/845]
```

Step 2: Create the schema.

- Ensure to use your username and password from service credentials
- The command below tells db2cli to run the commands in the file star-schema.sql on the production data warehouse.

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli execsql -dsn production
-user cds02 -passwd jH9BvoJUDGTMg -inputsql star-schema.sql
IBM DATABASE 2 Interactive CLI Sample Program
(C) COPYRIGHT International Business Machines Corp. 1993,1996
All Rights Reserved
Licensed Materials - Property of IBM
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
> CREATE TABLE FactBilling(billid integer not null primary key,customerid
integer NOT NULL,monthid integer NOT NULL,billedamount integer NOT NULL);
The SQL command completed successfully.
> CREATE TABLE DimMonth(monthid integer NOT NULL PRIMARY KEY,year integer
NOT NULL,month integer NOT NULL,monthname varchar(10) NOT NULL,quarter
integer NOT NULL,quartername varchar(2) NOT NULL);
The SQL command completed successfully.
> CREATE TABLE DimCustomer(customerid integer NOT NULL PRIMARY KEY,category
varchar(10) NOT NULL,country varchar(40) NOT NULL,industry varchar(40) NOT
NULL);
The SQL command completed successfully.
> ALTER TABLE FactBilling ADD FOREIGN KEY (customerid) REFERENCES
DimCustomer (customerid);
The SQL command completed successfully.
> ALTER TABLE FactBilling ADD FOREIGN KEY (monthid) REFERENCES DimMonth
(monthid);
The SQL command completed successfully.
>
theia@theiadocker-craigtrupp8:/home/project$
```

Exercise 6 - Populate the production data warehouse

Step 1: Download the data files.

```
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/DimCustomer.sql
--2023-10-06 12:11:27--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/DimCustomer.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 42724 (42K) [application/x-sql]
Saving to: 'DimCustomer.sql'

DimCustomer.sql
100%[===============================================================>]
41.72K  --.-KB/s    in 0s
```

```
2023-10-06 12:11:27 (107 MB/s) - 'DimCustomer.sql' saved [42724/42724]

theia@theiadocker-craigtrupp8:/home/project$
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/DimMonth.sql
--2023-10-06 12:11:27--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/DimMonth.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15279 (15K) [application/x-sql]
Saving to: 'DimMonth.sql'

DimMonth.sql
100%[===================================================================>]
14.92K  --.-KB/s    in 0s

2023-10-06 12:11:27 (253 MB/s) - 'DimMonth.sql' saved [15279/15279]

theia@theiadocker-craigtrupp8:/home/project$
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/FactBilling.sql
--2023-10-06 12:11:27--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/FactBilling.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3150636 (3.0M) [application/x-sql]
Saving to: 'FactBilling.sql'

FactBilling.sql
```

```
100%[==============================================================>]
3.00M  --.-KB/s    in 0.03s

2023-10-06 12:11:27 (108 MB/s) - 'FactBilling.sql' saved [3150636/3150636]

theia@theiadocker-craigtrupp8:/home/project$
theia@theiadocker-craigtrupp8:/home/project$ ls *.sql
DimCustomer.sql  DimMonth.sql  FactBilling.sql  star-schema.sql
theia@theiadocker-craigtrupp8:/home/project$
```

Step 2: Load the data in the data warehouse.

- Taking snippets from the command line output. Most important is the SQL command completion success/failure message after the attempted **execsql** command in the **db2cli**

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli execsql -dsn production
-user cds03902 -passwd jH9BvoJUDy8OGTMg -inputsql DimCustomer.sql
IBM DATABASE 2 Interactive CLI Sample Program
(C) COPYRIGHT International Business Machines Corp. 1993,1996
All Rights Reserved
Licensed Materials - Property of IBM
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
> INSERT INTO DimCustomer(customerid,category,country,industry) VALUES
(1,'Individual','Indonesia','Engineering'),(614,'Individual','United
States','Product
Management'),(615,'Individual','China','Services'),(616,'Individual','Russi
a','Accounting'),(617,'Individual','Chile','Business
Development'),(618,'Individual','Nicaragua','Human
Resources'),(41,'Company','Brazil','Marketing'),(619,'Individual','Russia',
'Business Development'),(620,'Individual','China','Business
Development'),(956,'Individual','Peru','Research and
Development'),(621,'Individual','Angola','Services'),(622,'Individual','Pol
and','Legal'),(623,'Individual','Italy','Training'),(624,'Individual','Indo
nesia','Sales'),(625,'Individual','Portugal','Services'),(626,'Individual',
'Mexico','Engineering'),(40,'Individual','Portugal','Human
Resources'),(627,'Company','Philippines','Services'),(894,'Company','Cuba',
'Business
Development'),(889,'Individual','France','Training'),(8,'Individual','Brazi
l','Engineering'),(919,'Company','Indonesia','Business
Development'),(890,'Company','South Africa','Training');
The SQL command completed successfully.
```

```
>
theia@theiadocker-craigtrupp8:/home/project$
```

- Just using the one output but ran the other two sql files and exec commands the same and just updated the sql file for the cli to execute.

Referential Integrity Note for SQL Population

- **Note** : Submitting and executing the sql scripts above was done in order of the "**Dimension**" tables being executed first followed by the "**Fact**" table. The "**Fact**" table holds the foreign keys for the dimensions so prior to uploading the fact you need the referential relationship to exist or would likely result in an error

Exercise 7 - Verify the data on the production data warehouse

Step 1: Download the verification sql file.

```
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/verify.sql
--2023-10-06 12:18:19--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Populating%20a%20Data%20Warehouse/verify.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 243 [application/x-sql]
Saving to: 'verify.sql'

verify.sql
100%[=============================================================>]
243  --.-KB/s    in 0s

2023-10-06 12:18:19 (27.3 MB/s) - 'verify.sql' saved [243/243]
```

Step 2: Verify the data in the data warehouse.

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli execsql -dsn production
-user cds03902 -passwd jH9BvoJUDy8OGTMg -inputsql verify.sql
IBM DATABASE 2 Interactive CLI Sample Program
(C) COPYRIGHT International Business Machines Corp. 1993,1996
All Rights Reserved
Licensed Materials - Property of IBM
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
> select count(*) as rowcount from DimMonth;
FetchAll:  Columns: 1
  ROWCOUNT
  132
FetchAll: 1 rows fetched.
> select count(*) as rowcount  from DimCustomer;
FetchAll:  Columns: 1
  ROWCOUNT
  1000
FetchAll: 1 rows fetched.
> select count(*) as rowcount from FactBilling;
FetchAll:  Columns: 1
  ROWCOUNT
  132000
FetchAll: 1 rows fetched.
>
theia@theiadocker-craigtrupp8:/home/project$
```

- Verify sql file

```sql
--Checking row in DimMonth Table
select count(*) as rowcount from DimMonth;
-- Checking row in DimCustomer Table
select count(*) as rowcount  from DimCustomer;
-- Checking row in FactBilling Table
select count(*) as rowcount from FactBilling;
```

Exercise 8 - Work with db2cli interactive command line

db2cli can also be used interactively.

Run the command below to open an interactive sql command shell to your production data warehouse. Make sure you use your username and password that you noted in Exercise 2.

- Get SQL CLI interactive access (truncated -user and -passwd args from actual session

```
theia@theiadocker-craigtrupp8:/home/project$ db2cli execsql -dsn production
-user cds02 -passwd jH9BvoJUDy8g
IBM DATABASE 2 Interactive CLI Sample Program
(C) COPYRIGHT International Business Machines Corp. 1993,1996
All Rights Reserved
Licensed Materials - Property of IBM
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
> select count(*) from DimMonth;
select count(*) from DimMonth;
FetchAll:  Columns: 1
  1
  132
FetchAll: 1 rows fetched.
> quit

theia@theiadocker-craigtrupp8:/home/project$
```