Data Definition Language(DDL)

A DDL is a language used to define data structures and modify data. For example, DDL commands can be used to add, remove, or modify tables within in a database.

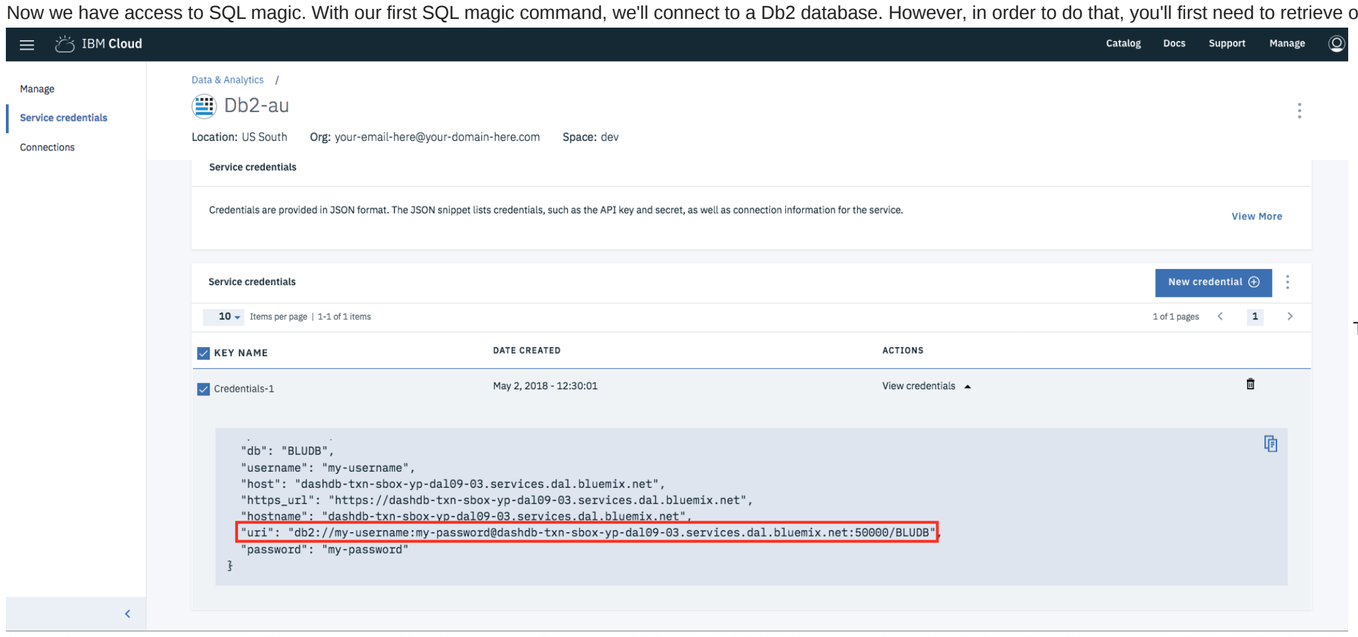
Connecting with the database

To communicate with SQL Databases from within a JupyterLab notebook, we can use the SQL "magic" provided by the ipython-sql extension. "Magic" is JupyterLab's term for special commands that start with "%". Below, we'll use the *load_ext* magic to load the ipython-sql extension. In the lab environemnt provided in the course the ipython-sql extension is already installed and so is the ibm_db_sa driver.

In [ ]:
```
# you will need to run the following command if you don't have them installed already
# ! conda install -c conda-forge ipython-sql
# ! conda install -c auto ibm_db_sa
```

In [ ]:
```
%load_ext sql
```

Now we have access to SQL magic. With our first SQL magic command, we'll connect to a Db2 database. However, in order to do that, you'll first need to retrieve or create your credentials to access your Db2 database.



This image shows the location of your connection string if you're using Db2 on IBM Cloud. If you're using another host the format is: username:password@hostname:port/database-name Enter your Db2 credentials in the connection string below. Recall you created Service Credentials in Part III of the first lab of the course in Week 1 i.e. from the `uri` field in the Service Credentials copy everything after `db2://` (but remove the double quote at the end) for example, if your credentials are as in the screenshot above, you would write: `%sql ibm_db_sa://my-username:my-password@dashdb-txn-sbox-yp-dal09-03.services.dal.bluemix.net:50000/BLUDB` Note the `ibm_db_sa://` prefix instead of `db2://` This is because JupyterLab's `ipython-sql` extension uses sqlalchemy (a python SQL toolkit) which in turn uses IBM's sqlalchemy dialect: `ibm_db_sa`

In [ ]:
```
# replace with your own credentials
%sql ibm_db_sa://my-username:my-password@dashdb-txn-sbox-yp-dal09-03.services.dal.bluemix.net:50000/BLUDB
```

1. Create Command

As the name suggest, it is used to create objects like databases and tables as shown below:

In [3]:
```
%%sql

create table STUDENTS (
        country VARCHAR(50),
        first_name VARCHAR(50),
        last_name VARCHAR(50)
);
```

```
 * ibm_db_sa://lzv55210:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[3]:
```
[]
```

2. Alter Command

Alter command is used to add or delete attributes from existing schema.

In [5]:
```
# Adding another column
%sql alter table STUDENTS add test_score INT
```

```
 * ibm_db_sa://lzv55210:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[5]:
```
[]
```

3. Drop Command

It is used to delete objects from the databases like tables.

In [2]:
```
%sql drop table STUDENTS
```

```
 * ibm_db_sa://lzv55210:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[2]:
```
[]
```