

Connect to Db2 database on Cloud using Python

The objective of the report was to use python programming language to communicate with databases on Cloud by creating ,inserting , updating and retriving the data frame table from the IBM Db2 cloud database.

```
In [26]: import ibm_db

In [27]: dsn_hostname = "764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"
dsn_uid = "yj193087"
dsn_pwd = "YPTftIMnGNW1JDrc"
dsn_driver = "{IBM DB2 ODBC DRIVER }"
dsn_database = "BLUDB"
dsn_port = "32536"
dsn_protocol = "TCP/IP"
dsn_security = "SSL"

In [28]: # creating the dsn connection string

dsn = (
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};").format(dsn_driver , dsn_database, dsn_hostname, dsn_port, dsn_protocol, dsn_uid , dsn_pwd, dsn_security)

print(dsn)

DRIVER={IBM DB2 ODBC DRIVER };DATABASE=BLUDB;HOSTNAME=764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=32536;PROTOCOL=TCP/IP;UID=yj193087;PWD=YPTftIMnGNW1JDrc;SECURITY=SSL;

In [29]: try:
    conn = ibm_db.connect(dsn, "", "")
    print("Connected to database: ",dsn_database, "as user: ", dsn_uid, "on host: ", dsn_hostname)

except:
    print("Unable to connect: ",ibm_db.conn_errormsg())

Connected to database:  BLUDB as user:  yj193087 on host:  764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud

In [30]: #Retrieve Metadata for the Database Server
server= ibm_db.server_info(conn)

print("DBMS_NAME: ", server.DBMS_NAME)
print("DBMS_VER:  ",server.DBMS_VER)
print("DB_NAME:    ",server.DB_NAME)

DBMS_NAME:  DB2/LINUX8664
DBMS_VER:   11.05.0800
DB_NAME:    BLUDB
```

Creating a table in the database

```
In [37]: # I will drop the table INSTRUCTOR in case it exists from a previous attempt
dropQuery = "drop table INSTRUCTOR"

dropStmt = ibm_db.exec_immediate(conn,dropQuery)

In [38]: # The error will occur .It just implies that the INSTRUCTOR table already exist.
createQuery =" create table INSTRUCTOR(ID INTEGER PRIMARY KEY NOT NULL ,FNAME VARCHAR (20) , LNAME VARCHAR(20), CITY VARCHAR(20),CCCODE CHAR(2))"

createStmt = ibm_db.exec_immediate(conn,createQuery)

In [39]: # DDL staement to insert values on the table

insertQuery = "Insert into INSTRUCTOR values(1 , 'Rav' , 'Ahuja','TORONTO','CA')"
```

```
insertStmt= ibm_db.exec_immediate(conn,insertQuery)

In [42]: # Adding moreraws to the table
insertQuery2 = "Insert into INSTRUCTOR values(2 , 'Raul' , 'Chong', 'Markham' , 'CA'), (3,'Hima','Vasudevan','Chicago','US')"
```

```
insertSt2 = ibm_db.exec_immediate(conn,insertQuery2)
```

Querying data in the table

```
In [44]: # Retrieving all rows from the INSTRUCTOR
selectQuery = "select * from INSTRUCTOR "
```

```
selectStmt = ibm_db.exec_immediate(conn,selectQuery)

In [45]: while ibm_db.fetch_row(selectStmt) !=False:
    print("ID:", ibm_db.result(selectStmt,0), "FNAME:", ibm_db.result(selectStmt, "FNAME"))

ID: 1 FNAME: Rav
ID: 2 FNAME: Raul
ID: 3 FNAME: Hima

In [49]: updateQuery = "update INSTRUCTOR set CITY = 'MOOSETOWN' where ID = '1'"

updateStmt = ibm_db.exec_immediate(conn,updateQuery)
```

Retrieving data into Pandas

```
In [52]: # Libraries
import pandas
import ibm_db_dbi

In [53]: # connecting to pandas
pconn = ibm_db_dbi.Connection(conn)

In [56]: # Retrieving all rows in the INSTRUCTOR table

selectQuery = "select*From INSTRUCTOR"

pdf =pandas.read_sql(selectQuery,pconn)

pdf

C:\Users\VictorM\AppData\Local\Temp\ipykernel_22120\116001121.py:3: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not te
sted. Please consider using SQLAlchemy.
pdf =pandas.read_sql(selectQuery,pconn)

Out[56]:
```

| | ID | FNAME | LNAME | CITY | CCCODE |
|---|----|-------|-----------|-----------|--------|
| 0 | 1 | Rav | Ahuja | MOOSETOWN | CA |
| 1 | 2 | Raul | Chong | Markham | CA |
| 2 | 3 | Hima | Vasudevan | Chicago | US |

```
In [57]: # The shape method to see how many rows and columns are in the data frame
pdf.shape

Out[57]: (3, 5)

In [59]: #Closing the connections so that we can avoid unused connections taking up resources
ibm_db.close(conn)

Out[59]: True

In [ ]:
```