College code -7125

Name: N.Muthulingaselvi

Register number :712521104025

Data warehousing with IBM cloud Db2 warehouse

Connect to Db2 database on cloud using python:

- Import the ibm_db Python library
- Enter the database connection credentials
- Create the database connection
- Close the database connection

Import the `ibm_db` Python library

The `ibm_db` API provides a variety of useful Python functions for accessing and manipulating data in an IBM® data server database, including functions for connecting to a database, preparing and issuing SQL statements, fetching rows from result sets, calling stored procedures, committing and rolling back transactions, handling errors, and retrieving metadata.

Identify the database connection credentials

Connecting to dashDB or DB2 database requires the following information:

- Driver Name
- Database name
- Host DNS name or IP address
- Host port
- Connection protocol
- User ID (or username)
- User Password

Create the DB2 database connection

Ibm_db API uses the IBM Data Server Driver for ODBC and CLI APIs to connect to IBM DB2 and Informix.

Close the Connection

We free all resources by closing the connection. Remember that it is always important to close connections so that we can avoid unused connections taking up resources.

Python program that connects to an IBM Db2 Warehouse on IBM Cloud:

```
python
import ibm_db

# Replace the placeholders with
your own IBM Cloud credentials
dsn_driver = "{IBM DB2 ODBC
DRIVER}"
dsn_database = "<your-db-name>"
dsn_hostname =
"<your-hostname>"
dsn_port = "<your-port>"
dsn_protocol = "TCPIP"
dsn_uid = "<your-username>"
dsn_pwd = "<your-password>"
```

```
# Establish a connection to the
Db2 Warehouse
conn_string = f"DRIVER={dsn_dri
ver};DATABASE={dsn_database};HO
STNAME={dsn_hostname};PORT={dsn
_port};PROTOCOL={dsn_protocol};
UID={dsn_uid};PWD={dsn_pwd};"
conn =
ibm_db.connect(conn_string, "",
"")

# Execute a sample SQL query
query = "SELECT * FROM
your_table"
stmt =
ibm_db.exec_immediate(conn,
query)

# Fetch and print the results
while ibm_db.fetch_row(stmt):
    result =
ibm_db.result_tuple(stmt)
    print(result)

# Close the connection
ibm_db.close(conn)
```