

KGiSL INSTITUTE OF TECHNOLOGY

INSTITUTION CODE : 7117

NAAN MUDHALVAN

PROJECT TITLE:

Data Warehousing with IBM Cloud Db2 Warehouse.

MENTOR:

Mrs. R. Indu Poornima

TEAM MEMBERS:

- 1.** *Keerthana. J*
- 2.** *Saruthi. T*
- 3.** *Kishore. K*
- 4.** *Jeevanandham. S*

IMPLEMENTING ADVANCED ANALYTICS AND MACHINE LEARNING MODELS FOR PREDICTIVE ANALYSIS IN THE DATA WAREHOUSE

PROBLEM STATEMENT

The current state of the data warehouse lacks the capability to provide predictive insights from historical data. As a result, we are unable to anticipate trends, make proactive decisions, or optimize our operations effectively. This presents a significant challenge in an increasingly competitive business landscape. The main objective of the project focused on data warehousing using IBM Cloud Db2 Warehouse is to enable efficient data management and analysis for decision-making processes within an organization.

In this phase, we have described the continuation of our project.

We here continued building the data warehouse by implementing ETL processes and enabling data exploration.

ETL PROCESSES:

MySQL Database Connection:

The code establishes a connection to a MySQL database, assuming it exists, with the necessary connection parameters provided.

Data Extraction from MySQL:

The code executes a SELECT query to fetch data from the MySQL database, specifically from the 'CUSTOMER' table. It fetches all the rows from the 'CUSTOMER' table.

Data Transformation:

The code assumes that the data formats between the MySQL and Db2 Warehouse are compatible, hence no explicit transformation is conducted. If data transformation were necessary, this is where you'd perform operations such as data type conversions or data cleansing.

Data Loading into Db2 Warehouse:

The code iterates through the fetched rows from the MySQL database and inserts each row into the 'CUSTOMER' table in IBM Cloud Db2 Warehouse. It utilizes the `ibm_db` library to execute SQL queries on the Db2 Warehouse connection.

DATA EXPLORATION AND ANALYSIS:

SQL Query for Data Exploration:

The code defines a sample SQL query that retrieves data from multiple tables (CUSTOMER, ORDERS, ORDER_ITEMS, and PRODUCT) by performing inner joins between them.

Executing the SQL Query:

The code uses the `ibm_db` library to execute the SQL query on the Db2 Warehouse connection.

Fetching and Printing Results:

The code fetches the results of the executed SQL query and prints each result row to the console.

CLOSING CONNECTIONS:

The code appropriately closes the connections to both the MySQL database and the Db2 Warehouse after the data extraction, transformation, loading, and exploration processes have been completed.

CODING:

Extract, Transform, and Load (ETL) Processes

Extract data from an external database (e.g., MySQL) and load it into Db2 Warehouse

Assuming you have a MySQL database

import mysql.connector

Connect to MySQL database

```
mysql_conn = mysql.connector.connect(
    host="MYSQL_HOST",
    user="MYSQL_USER",
    passwd="MYSQL_PASSWORD",
    database="MYSQL_DATABASE"
)
```

Extract data from MySQL

```
mysql_cursor = mysql_conn.cursor()
mysql_cursor.execute("SELECT * FROM CUSTOMER")
mysql_result = mysql_cursor.fetchall()
```

Transform data if necessary

Here we assume the data formats are compatible

Load data into Db2 Warehouse

for row in mysql_result:

```
ibm_db.exec_immediate(conn, f'''INSERT INTO CUSTOMER
                        VALUES ({row[0]}, '{row[1]}', '{row[2]}', '{row[3]}',
                        '{row[4]}')''')
```

Close the MySQL connection

```
mysql_conn.close()
```

Data Exploration and Analysis

Enable data architects to explore and analyze data within Db2 Warehouse using SQL queries

Example SQL query for data exploration

```
sql_query = '''
```

```
SELECT c.name, p.product_name, o.order_date, oi.quantity, oi.unit_price
FROM CUSTOMER c
```

```
JOIN ORDERS o ON c.customer_id = o.customer_id
```

```
JOIN ORDER_ITEMS oi ON o.order_id = oi.order_id
```

```
JOIN PRODUCT p ON oi.product_id = p.product_id
```

```
'''
```

Execute the SQL query

```
stmt = ibm_db.exec_immediate(conn, sql_query)
```

Fetch and print results

```
result = ibm_db.fetch_tuple(stmt)
```

while result:

```
    print(result)
```

```
result = ibm_db.fetch_tuple(stmt)
```

```
# Close the connection
```

```
ibm_db.close(conn)
```

The above code snippet demonstrates how to extend the existing data warehouse setup by implementing ETL processes to extract data from an external database (MySQL), transform it if necessary, and load it into IBM Cloud Db2 Warehouse. It also showcases how to enable data exploration and analysis within the Db2 Warehouse environment using SQL queries.