

Data Warehousing with IBM Cloud Db2 Warehouse

Phase 3: Development Part 1

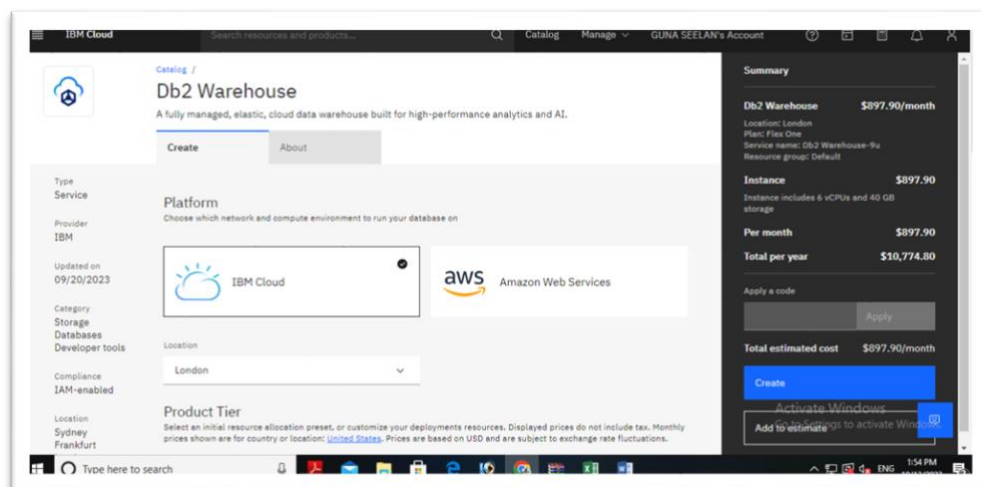
To do:

Start building the data warehouse using IBM Cloud Db2 Warehouse.

Define the schema and structure of the data warehouse tables. Identify data sources (e.g., CSV files, databases) and design a strategy to integrate them into the data warehouse.

Steps to be followed:

- **Set Up IBM Cloud Db2 Warehouse:** First of all we need to create an IBM Cloud account and provision Db2 Warehouse on IBM Cloud. Follow the documentation and guides provided by IBM to set up Db2 Warehouse in your IBM Cloud account.



- **Define Schema and Structure:** The first step is to define the schema and structure of your data warehouse tables. This involves designing the tables that will store your data. Consider the type of data you'll be storing, the relationships between data, and how you'll use this data. Create an initial schema with tables, columns, and data types.

Eg:

Creating a sample "Sales" table

```
CREATE TABLE Sales (  
    SaleID INT,  
    ProductID INT,  
    SaleDate DATE,  
    Amount DECIMAL(10, 2)  
);
```

- **Identify Data Sources:** Identify the data sources you want to integrate into the data warehouse. These sources can include:
- CSV Files: If you have data in CSV files, plan to upload them to Db2 Warehouse.
- Databases: If your data is stored in other databases, you'll need to plan for data extraction and transformation.

Example for load data:

Load data from a CSV file into the "Sales" table

IMPORT FROM 'your_file.csv'

OF DEL

INSERT INTO Sales;

- **Design Data Integration Strategy:** Your data integration strategy should involve the following steps:
 - a. **Data Extraction:** Extract data from your identified sources. For CSV files, you can use data loading tools or scripts to import data. For databases, consider using ETL (Extract, Transform, Load) tools like IBM DataStage or writing custom scripts to extract data.
 - b. **Data Transformation:** Once data is extracted, you may need to transform it to fit the structure of your data warehouse. This might include data cleansing, data type conversion, and other transformations.
 - c. **Data Loading:** Load the transformed data into your Db2 Warehouse tables. IBM provides various methods for data loading, including the LOAD utility and SQL-based inserts.
 - d. **Scheduling and Automation:** Consider how often you need to refresh your data warehouse. You may want to set up a schedule or automation process for regular data updates.
- **Data Warehouse Maintenance:** Regularly maintain and optimize your data warehouse. This includes monitoring performance, managing data growth, and ensuring data quality.
- **Access and Query Data:** Once your data warehouse is populated, you can use SQL queries and tools to access and analyze the data. Ensure you have the necessary user accounts and permissions set up for data access.
- **Security and Compliance:** Implement security measures to protect your data warehouse. Ensure that your data warehouse complies with any regulatory requirements applicable to your industry.
- **Backup and Recovery:** Set up backup and recovery procedures to safeguard your data in case of unexpected data loss.
- **Documentation:** Keep detailed documentation of your data warehouse setup, schema, integration processes, and data sources for future reference.

Sample code for the above process:(python).

```
Import ibm_db
Import pandas as pd
# Replace these variables with your IBM Cloud Db2 Warehouse Lite service credentials
Dsn_hostname = "your-db2-hostname"
Dsn_uid = "your-db2-username"
Dsn_pwd = "your-db2-password"
Dsn_port = "your-db2-port"
Database_name = "your-db2-database-name"
# Connect to the Db2 Warehouse Lite instance
Dsn = (
    F"DRIVER={{IBM DB2 ODBC DRIVER}};"
    F"DATABASE={database_name};"
    F"HOSTNAME={dsn_hostname};"
    F"PORT={dsn_port};"
    F"PROTOCOL=TCPIP;"
    F"UID={dsn_uid};"
    F"PWD={dsn_pwd};"
)
Conn = ibm_db.connect(dsn, "", "")
# Define the schema and create a table
Create_table_sql = """
CREATE TABLE Sales (
    SaleID INT,
    ProductID INT,
    SaleDate DATE,
    Amount DECIMAL(10, 2)
)
"""

Stmt = ibm_db.exec_immediate(conn, create_table_sql)

# Load data from a CSV file into the table using pandas
Csv_file_path = "path/to/your/data.csv"
Data = pd.read_csv(csv_file_path)
For _, row in data.iterrows():
    Insert_sql = f"""
    INSERT INTO Sales (SaleID, ProductID, SaleDate, Amount)
    VALUES ({row['SaleID']}, {row['ProductID']}, {row['SaleDate']}, {row['Amount']})
    """

    Stmt = ibm_db.exec_immediate(conn, insert_sql)

# Commit the changes and close the connection
Ibm_db.commit(conn)
Ibm_db.close(conn)
```
