NAAN MUDHALVAN : IBM

TECHNOLOGY : CLOUD COMPUTING

PROJECT TITLE :-

BIG DATA ANALYSIS WITH IBM CLOUD DATABASE

PHASE : 3

THE DEVELOPEMENT PART :-

"Development part" typically refers to the phase of creating, designing, and implementing the software applications, scripts, or data processing pipelines that interact with the IBM Cloud database for the purpose of analyzing and extracting insights from large volumes of data.

BIG DATA :-

"Big Data" refers to the large and complex sets of data that can be effectively managed, processed, and analyzed using cloud-based infrastructure and services.

PROGRAM :

Import ibm_db

```
# Define the connection parameters
Db_credentials = {
    "hostname": "your-db-hostname",
    "port": 50000,
    "user": "your-username",
    "password": "your-password",
```

```
    "database": "your-database-name"

}


# Create the database connection

Conn = ibm_db.connect(

    F"DATABASE={db_credentials['database']};HOSTNAME={db_credentials['hostname']};"

    F"PORT={db_credentials['port']};PROTOCOL=TCPIP;UID={db_credentials['user']};"

    F"PWD={db_credentials['password']};",

    "",

    ""

)


# Execute a SQL query to retrieve data

Sql_query = "SELECT * FROM your_table"

Stmt = ibm_db.exec_immediate(conn, sql_query)


# Fetch and print the results

Result = ibm_db.fetch_assoc(stmt)

While result:

    Print(result)

    Result = ibm_db.fetch_assoc(stmt)


# Close the database connection

Ibm_db.close(conn)
```

BIG DATA - DATA EXTRACTION :-

"Data extraction" refers to the process of retrieving and transferring data from a source location or system to another destination, typically within a cloud-based environment.

PROGRAM :

```
Import ibm_db
```

```python
# Define the connection parameters
Db_credentials = {
    "hostname": "your-db-hostname",
    "port": 50000,
    "user": "your-username",
    "password": "your-password",
    "database": "your-database-name"
}

# Create the database connection
Conn = ibm_db.connect(
    F"DATABASE={db_credentials['database']};HOSTNAME={db_credentials['hostname']};"
    F"PORT={db_credentials['port']};PROTOCOL=TCPIP;UID={db_credentials['user']};"
    F"PWD={db_credentials['password']};",
    "",
    ""
)

# Execute a SQL query to extract data
Sql_query = "SELECT * FROM your_table"
Stmt = ibm_db.exec_immediate(conn, sql_query)

# Fetch and process the results
While ibm_db.fetch_row(stmt):
    # Extract data and process as needed
    Data = ibm_db.result(stmt, "COLUMN_NAME_OR_INDEX")
    Print(data)

# Close the database connection
```

Ibm_db.close(conn)

BIG DATA – DATA ANALYSIS :-

Data analysis in cloud computing refers to the process of examining and deriving insights from large volumes of data using cloud-based resources and services.

PROGRAM :

```
From pyspark.sql import SparkSession


# Create a Spark session
Spark = SparkSession.builder.appName("BigDataAnalysis").getOrCreate()


# Read data from cloud storage (e.g., Amazon S3)
Data = spark.read.csv("s3://your-bucket/your-data.csv", header=True, inferSchema=True)


# Perform data analysis
Result = data.groupBy("column_name").agg({"numeric_column": "mean"})


# Show the analysis results
Result.show()


# Stop the Spark session
Spark.stop()
```

CAMPUS ECONOMIC DATA ANALYSIS :-

Campus economic data analysis for Big Data analysis with an IBM Cloud database refers to the process of examining and extracting insights from large datasets related to the financial and economic aspects of a college or university campus.

PROGRAM :

```
Import ibm_db


# Define the connection parameters
```

```python
Db_credentials = {
    "hostname": "your-db-hostname",
    "port": 50000,
    "user": "your-username",
    "password": "your-password",
    "database": "your-database-name"
}

# Create the database connection
Conn = ibm_db.connect(
    F"DATABASE={db_credentials['database']};HOSTNAME={db_credentials['hostname']};"
    F"PORT={db_credentials['port']};PROTOCOL=TCPIP;UID={db_credentials['user']};"
    f"PWD={db_credentials['password']};",
    "",
    ""
)

# Execute SQL queries for economic data analysis
sql_query = """
SELECT * FROM economic_data
WHERE year >= 2020
"""

stmt = ibm_db.exec_immediate(conn, sql_query)

# Fetch and process the results
while ibm_db.fetch_row(stmt):
    # Extract and analyze economic data
    # Perform your data analysis operations here
```

```
    # For example: calculate averages, generate reports, or create visualizations


# Close the database connection

ibm_db.close(conn)
```

FINAL PROGRAM :-

```
Import ibm_db

Import pandas as pd

Import matplotlib.pyplot as plt


# Define your database credentials

Db_credentials = {

    "hostname": "your-db-hostname",

    "port": 50000,

    "user": "your-username",

    "password": "your-password",

    "database": "your-database-name"

}


# Create a database connection

Conn = ibm_db.connect(

    F"DATABASE={db_credentials['database']};HOSTNAME={db_credentials['hostname']};"

    F"PORT={db_credentials['port']};PROTOCOL=TCPIP;UID={db_credentials['user']};"

    F"PWD={db_credentials['password']};",

    "",

    ""

)


# Define an SQL query to extract campus economic data
```

```python
Sql_query = "SELECT * FROM campus_economics"

Stmt = ibm_db.exec_immediate(conn, sql_query)


# Fetch data and create a Pandas DataFrame

Data = []

While ibm_db.fetch_row(stmt):

    Data.append([ibm_db.result(stmt, "YEAR"), ibm_db.result(stmt, "REVENUE"),
ibm_db.result(stmt, "EXPENSES")])


Columns = ["Year", "Revenue", "Expenses"]

Df = pd.DataFrame(data, columns=columns)


# Calculate profit for each year and add it to the DataFrame

Df["Profit"] = df["Revenue"] – df["Expenses"]


# Display the DataFrame

Print("Campus Economic Data:")

Print(df)


# Perform some basic data analysis, e.g., visualize profits

Plt.plot(df["Year"], df["Profit"])

Plt.title("Campus Economic Profits Over the Years")

Plt.xlabel("Year")

Plt.ylabel("Profit")

Plt.show()


# Close the database connection

Ibm_db.close(conn)
```

OUTPUT :

```
YEAR,REVENUE,EXPENSES
2020,1000000,800000
2021,1100000,820000
2022,1200000,840000
```

Campus Economic Data:

	Year	Revenue	Expenses	Profit
0	2020	1000000	800000	200000
1	2021	1100000	820000	280000
2	2022	1200000	840000	360000