BIGDATA ANALYSIS WITH IBM CLOUD DATABASES

712221205046 - ZIBRIL A

PHASE-3 SUBMISSION DOCUMENT

INTRODUCTION :

Building a big data analysis solution using IBM Cloud Databases is a multi-step process. It involves setting up the necessary infrastructure, ingesting and storing data, and then performing analysis. Here's a high-level overview of the steps you can follow. If you don't already have an IBM Cloud account, you'll need to create one. Visit the IBM Cloud website (https://www.ibm.com/cloud) and sign up for an account. Once you have an account, log in to the IBM Cloud console using your credentials. IBM Cloud offers various database services, including IBM Db2, IBM Cloud Databases for PostgreSQL, IBM Cloud Databases for MySQL, and more. Select the database service that best fits your requirements for your big data analysis.

DATABASE SETUP :

If you don't have an IBM Cloud account, follow these steps to create one:

- Visit the IBM Cloud website (https://www.ibm.com/cloud) in your web browser.
- Click on the "Sign Up" or "Get Started for Free" button.
- Follow the registration process, which typically includes providing your email, creating a password, and verifying your identity.

To create a database instance, follow these steps:

- In the IBM Cloud console, navigate to the "Catalog" section.
- Use the search bar to find the database service you want to use (e.g., "IBM Cloud Databases for PostgreSQL").
- Click on the service to go to its details page.
- Click the "Create" button.
- **Log in to IBM Cloud:**
- Open a command prompt or terminal and log in to your IBM Cloud account using the following command. You'll be prompted to enter your IBM Cloud credentials.
- **Target the appropriate resource group:**
- You need to target a specific resource group where you want to create your database instance. Replace **`your-resource-group`** with the name of your resource group.
- Use the following command to create an IBM Cloud Databases for PostgreSQL instance. Replace the placeholders with your preferred values.

bashCopy code
ibmcloud databases-for-postgresql create <instance-name> --name <instance-name> --plan <plan-name> --location <regioibmcloud databases-for-postgresql create my-postgresql-instance --name my-postgresql-instance --plan standard --location us-south

- `<instance-name>`: Replace with a unique name for your database instance.
- `<plan-name>`: Choose the plan you want to use (e.g., "standard").
- `<region>`: Choose the region where you want to deploy the instance (e.g., "us-south").

- **Wait for Deployment:**

  The database instance will take some time to be provisioned. You can check its status using the following command:

  bashCopy code
  `ibmcloud databases-for-postgresql list`

  Wait until the status of your instance changes to "Active."

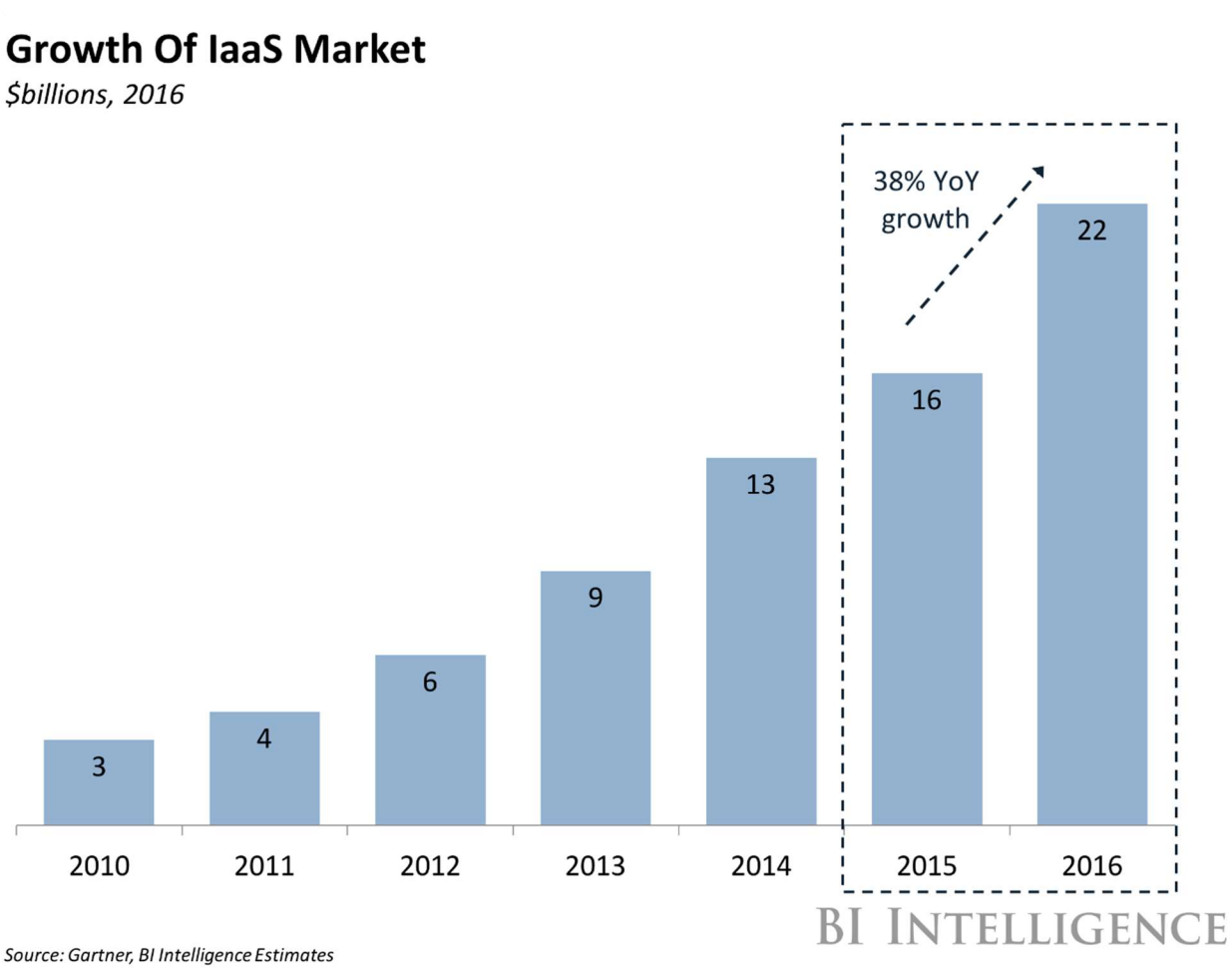
- **Access Your Database:**

  After the instance is active, you can access it using the provided connection details (hostname, port, and credentials).
  You can retrieve the connection details using the following command:

  bashCopy code
  `ibmcloud databases-for-postgresql get-connection <instance-name>`

  Replace `<instance-name>` with the name of your database instance.

Now you have successfully created an IBM Cloud Databases for PostgreSQL instance using the IBM Cloud CLI. You can use these connection details to connect to and work with your PostgreSQL database.



DATA EXPLORATION AND ANALYSIS:

To explore and analyze a dataset, and to perform basic data cleaning and transformation, you will typically use SQL queries if you are working with a relational database like IBM Db2 or IBM Cloud Databases for PostgreSQL. Here are some example SQL queries and data manipulation scripts you can use as a starting point:

1. **Explore the Dataset:**

To get an overview of the data, you can run basic SQL queries to inspect the structure and content of your dataset:

```
-- Show the first few rows of a table

SELECT * FROM your_table LIMIT 10;



-- List all tables in the database

\dt
```

2. **Filter and Select Data:**

To filter and select specific data from the dataset, you can use SQL's **SELECT** statement with conditions

```
-- Select specific columns and filter rows based on a condition

SELECT column1, column2

FROM your_table

WHERE column3 > 100;
```

;

3. **Aggregation and Grouping:**

You can perform aggregate functions and group data by one or more columns to get summary statistics

```
-- Calculate average and total for a numeric column, grouped by a categorical column

SELECT category_column, AVG(numeric_column) AS avg_value, SUM(numeric_column) AS total_value

FROM your_table

GROUP BY category_column;
```

-- Calculate average and total for a numeric column, grouped by a categorical

Data Cleaning and Transformation:

Data cleaning and transformation often involve handling missing values and correcting data inconsistencies:

- **Handle Missing Values:**

  You can identify and handle missing values in your dataset. For example, to replace missing values in a numeric column with the mean of that column:

- **Data Type Conversion:**

  Convert data types as needed, such as changing date strings to date objects

  ```
  ALTER TABLE your_table

  ALTER COLUMN date_column TYPE DATE USING to_date(date_column, 'YYYY-MM-DD');V
  ```

- **Data Standardization:**

  Standardize data values, such as capitalizing text

  ```
  UPDATE your_table

  SET text_column = INITCAP(text_column);
  ```

4.Joining Tables:

If you have multiple tables in your database, you can join them to combine data for analysis

UPDATE your_table

SET numeric_column = (SELECT AVG(numeric_column) FROM your_table WHERE numeric_column IS NOT NULL)

WHERE numeric_column IS NULL;

Data Aggregation and Analysis:

Perform more complex data analysis using SQL functions and expressions. For instance, calculating the correlation between two numeric columns.

SELECT CORR(column1, column2) AS correlation

FROM your_table;