I am a Data Engineer with **two years** of experience specializing in Data Migration and Analytics using **SAP BODS and HANA.**

Recently, I **transitioned** into **Azure Data** Engineering, where I’ve worked **on projects** involving scalable data **pipelines**, data **transformation**, and analytics using Azure Data Factory, Databricks, and Synapse Analytics.

At Accenture, I earned a **fast-track promotion** after passing the Tech Leap exam in Azure, which highlights my dedication to growing my cloud-based data engineering skills.

I’m now looking for opportunities to apply my expertise in delivering end-to-end cloud-based data solutions.  
Since I wanted to move to azure so I created three self driven projects in azure based on my own interest  
  
 **Azure ETL**

**Olympics Pipeline**

**Bing Analytics**

Thank you I happy to explain for you

--------------------------------------------------------------------------------------------------------------  
  
I have individually handled the end-to-end process of three key objects, including gathering requirements from the functional team and creating mapping documents that detail the data sources, transformations, and target destinations, ensuring full alignment with business needs.

For the data migration process, we start by extracting data from CSV files, transforming it according to business requirements, and generating a pre-load file for business validation.

Once the business validates the data, we proceed with loading it into the target system. After the data load is completed, we generate a post-load report for the business to verify and confirm.

Additionally, I have experience working with and optimizing existing ETL workflows for modules such as **OTC (Order to Cash)**, **CTD (Customer to Data)**, and **BP (Business Partner)**.

**Project Overview:**

* **Stages**: Ingestion, Transformation, Loading, and Reporting
* **Key Focus**: Azure Data Factory, Azure Databricks, Synapse Analytics, Data Lake, Delta Lake, and Azure Logic Apps

**1. Environment Setup & Data Ingestion:**

* **Azure Self-Hosted Integration Runtime (IR)**:
  + Used for **securely extracting data** from on-premises **MySQL database** to **Azure Data Lake Gen2**.
  + **Security**: **Azure Key Vault** for storing credentials (username, password) ensuring they are **masked** and **secure**.
* **Data Ingestion Pipeline**:
  + **Azure Data Factory (ADF)**: Built a **pipeline** with a **Lookup Activity** to fetch table names and schema from the database.
  + Used **Self-Hosted IR** for **secure data transfer** from on-premises systems to cloud.
  + **For Each Loop**: To iterate over multiple tables and load data using **Copy Data Activity** into the **Bronze layer** (raw data).
* **Data Storage**:
  + Data saved in **CSV format** in **Azure Data Lake Gen2** (Bronze layer).
* **Mail Trigger Integration** (for monitoring pipeline status):
  + **Success & Failure Notifications**: Configured with **Azure Logic Apps**.
  + **Web Activity**: Used to send email notifications on success or failure, detailing pipeline status and timestamps.

**Data Transformation:**

1. **Bronze, Silver, and Gold** – Data layers for raw, cleaned, and business-ready data.
2. **Mounting** – Linking Azure Data Lake storage to Databricks file system.

**Bronze to Silver Transformation:**

1. **from\_utc\_timestamp** – Convert timestamp to a specific time zone.
2. **date\_format** – Standardize date formats.
3. **filter()** – Filter rows based on conditions (e.g., removing invalid data).
4. **fillna()** – Handle missing or null values by filling them with default values.
5. **Delta format** – Store data in Delta format for efficient querying and updates.

**Silver to Gold Transformation:**

1. **join()** – Enrich data by joining multiple data sources (e.g., left join).
2. **Derived Columns** – Create new columns like Profit Margin: (sales\_amount - cost) / sales\_amount.
3. **dropDuplicates()** – Remove duplicate rows based on key columns (e.g., order\_id).

**Column Transformation:**

1. **column renaming (snake\_case)** – Standardize column names (e.g., ProductName → product\_name).
2. **substring()** – Trim column values (e.g., reducing URL length).
3. **Saving Data** – Store final data in **Delta format for efficient processing** and schema management.

**3. Creating Token for Secure Communication:**

* **Token Authentication**: Set up **token-based authentication** between **Databricks** and **ADF** pipeline for secure interaction.
* **Azure Key Vault**: Stored the **token securely** in Azure Key Vault for masking and secure retrieval.

**4. Synapse Analytics Integration:**

* **SQL Database Creation** in **Azure Synapse Analytics**.
  + **Serverless SQL Pool**: Opted for cost-effective on-demand querying.
  + **Dedicated SQL Pool**: Not used here, as it is more suitable for heavy, high-performance workloads.
* **Creating Views**:
  + **View** created on data in **Azure Data Lake** using **Synapse SQL**.
  + **Advantages**: Views automatically reflect changes in the underlying data, ensuring **real-time access** without reloading data.

**Challenges faced**:  
I couldn’t able to extract data from on-premises because fire wall is restricting the action

* **Tokiyo Olympic**   
    
  Ingestion  
  lookup – look for all tables   
  For each loop – inside Copy data activity -to copy all files one by one to store in the azure data lake storage as a CSV file in the bronze folder  
  following that created notebook for bronze to silver , silver to gold   
    
  transformation
* **Fabric Project**
* This Project consist of **five stages** – Data ingestion,trf,sentiment analysis,data visualization, Alert.  
    
  **Project Summary:**

This project end-to-end data engineering solution built on **Microsoft Fabric**, designed to power a **Bing News Data Analytics platform**.

providing insights for data-driven decision-making.

**Key Steps & Technologies:**

1. **Data Ingestion:**
   * Raw JSON data is ingested from the **Bing API** into **OneLake** using **Azure Data Factory**.
   * This step pulls in **articles, headlines, and publication details, date, detailed description of the news**
   * In this pipeline, build the notebook (data trf), and sentiment analysis.
2. **Data Transformation:**
   * **Synapse Data Engineering (Spark Notebooks)** is used to clean and transform the raw data into structured **Delta Tables**.
   * The data is stored in a **Medallion Architecture**:
     + **Bronze Layer**: Raw data (ingested JSON).
     + **Silver Layer**: Cleaned and transformed data.
     + **Gold Layer**: Aggregated and analyzed data ready for visualization.
3. **Sentiment Analysis:**
   * The transformed data undergoes **Sentiment Analysis** using **Synapse Data Science** and machine learning models.
   * The sentiment of news articles is classified as **positive**, **neutral**, or **negative**.
4. **Data Visualization:**
   * An interactive **Power BI** dashboard is created to visualize trends, sentiment analysis results, and key insights for stakeholders.
5. **Alerts:**
   * **Data Activator** is configured to send alerts when specific conditions, like **negative sentiment** or trends, are detected.

END>>>>>………………………………………………………………………………….

# Reading data from a CSV file

df = spark.read.format("csv").option("header", "true").option("inferSchema", "true").load("path/to/input/file.csv")

# Writing data to a Parquet file

df.write.format("parquet").save("path/to/output/file.parquet")

**Features of Databricks:**

1. **Unified Platform**: Combines tools for data engineers, scientists, and analysts to work together on one platform.
2. **Spark Integration**: Databricks manages Apache Spark for fast, scalable data processing—great for big data.
3. **Delta Lake**: A feature that ensures data consistency, reliability, and allows you to track changes over time.
4. **Collaborative Notebooks**: Interactive notebooks that support multiple programming languages, helping teams collaborate and share insights.
5. **Machine Learning Tools**: Databricks makes it easy to create, train, and deploy machine learning models with tools like **MLflow**.
6. **Scalability**: Automatically adjusts resources to handle both small and large amounts of data, making it cost-effective.
7. **SQL Analytics**: Easy querying of data using SQL, with support for BI tools like **Power BI** and **Tableau**.
8. **Real-time Processing**: Lets you process data in real-time, so you can monitor and react to data instantly.
9. **Security**: Databricks includes features like role-based access control (RBAC) to secure data and meet enterprise security standards.
10. **Cloud Integration**: Works well with cloud services like Azure, AWS, and Google Cloud, and integrates with their storage systems.

**Key Points for Your Interview:**

1. **Collaboration**: Explain how Databricks allows teams to work together more effectively using shared notebooks and a unified platform.
2. **Apache Spark**: Share how Databricks simplifies using **Apache Spark** to process large datasets quickly.
3. **Data Consistency**: Mention how **Delta Lake** helps ensure your data is always accurate and consistent.
4. **Real-Time Data**: Talk about how you can process data in real-time to build applications like live dashboards.
5. **Machine Learning**: If relevant, explain how Databricks helps build and deploy machine learning models, tracking them with **MLflow**.
6. **Scalability**: Discuss how Databricks automatically scales resources up or down depending on your data, which saves costs.
7. **Security**: Explain how **role-based access control (RBAC)** ensures only the right people can access sensitive data.

These are simplified points that should be easier to understand and communicate in an interview. Let me know if you need further explanation!