## Inferring Schema from a Google Drive File Using PySpark

Google Colab can be integrated with **Google Drive** to process datasets using **PySpark**. This allows seamless access to **CSV, Parquet, and other structured files** stored in Drive while automatically inferring their schema.

The process begins by **mounting Google Drive** in Colab using `drive.mount('/content/drive')`. This makes files available under `/content/drive/My Drive/`. Next, **PySpark is installed and initialized**, setting up a Spark session for data processing.

To **infer schema from a CSV file**, PySpark reads the file with `inferSchema=True`, which detects column names and data types automatically. For **Parquet files**, PySpark natively understands the schema. The inferred schema is printed using `df.printSchema()`, and sample records are displayed using `df.show(5)`.

This method helps avoid **manual schema definition**, ensuring flexible handling of different datasets. Additionally, the inferred schema can be extracted as JSON using `df.schema.json()`, making it useful for **logging and metadata storage**.

By leveraging **PySpark and Google Drive**, Colab users can efficiently analyze large datasets, perform **ETL operations**, and prepare data for **BigQuery or machine learning workflows**. This approach simplifies **data processing at scale**, reducing manual effort while ensuring accurate schema detection.

**Pyspark Code :**

```
from pyspark.sql import SparkSession

# Initialize Spark session
spark = SparkSession.builder.appName("InferSchemaGzip").getOrCreate()

# Define the path to your gzip file
file_path = [
   "/content/drive/My Drive/GSynergy Challenge/hier.clnd.dlm.gz"
]

# Read the gzip file (assuming it's pipe-delimited `|`)
df = spark.read.option("header", "true") \
              .option("inferSchema", "true") \
```

```
                .option("sep", "|") \
                .csv(file_path)

# Show inferred schema
df.printSchema()
```

**Result :**

```
root
 |-- fscldt_id: integer (nullable = true)
 |-- fscldt_label: string (nullable = true)
 |-- fsclwk_id: integer (nullable = true)
 |-- fsclwk_label: string (nullable = true)
 |-- fsclmth_id: integer (nullable = true)
 |-- fsclmth_label: string (nullable = true)
 |-- fsclqrtr_id: integer (nullable = true)
 |-- fsclqrtr_label: string (nullable = true)
 |-- fsclyr_id: integer (nullable = true)
 |-- fsclyr_label: integer (nullable = true)
 |-- ssn_id: string (nullable = true)
 |-- ssn_label: string (nullable = true)
 |-- ly_fscldt_id: integer (nullable = true)
 |-- lly_fscldt_id: integer (nullable = true)
 |-- fscldow: integer (nullable = true)
 |-- fscldom: integer (nullable = true)
 |-- fscldoq: integer (nullable = true)
 |-- fscldoy: integer (nullable = true)
 |-- fsclwoy: integer (nullable = true)
 |-- fsclmoy: integer (nullable = true)
 |-- fsclqoy: integer (nullable = true)
 |-- date: date (nullable = true)
```

Similarly schemas are inferred from all the given files and used to create the **ER diagram**.