----------------------------------------------------------

Processing the Data to convert from XML to CSV and JSON

----------------------------------------------------------

1) We created a Huge Machine (24 CPU, 500 GB RAM) on GCP.

2) We ran a script to convert XML data to CSV.(download\_file.sh and xml\_2\_csv.py )

3) The Script and python file is Provided in the submission bundle.

4) THe CSV data was creating problems as it did not carry the Column Name.

5) So we ran a different script on similar machine to convert data to JSON.(download\_file.sh and xml\_2\_json.py )

6) JSON provides both column name and data so we can work with that.

7) The script and python file is provided in the submission bundle.

---------------------------------------

Loading Data into the Bucket:-

---------------------------------------

1) Select the Storage Option in Main Menu

2) Create a new Bucket.

3) Create a new folder named 'Dataset'

4) Create sub folders with name same as that of data file/s.

5) Upload the file/s in their respected folder

---------------------------------------------------------

Steps to start a Cluster in Google Cloud Platform:-

---------------------------------------------------------

1) Make a new Google CLoud Project.

2) Select Dataproc option (present under Big Data submenu).

3) Create a new CLuster with 1 Master and 5 worker Nodes ( 4 CPU,80 GB memory and 500 GB disk size ).

4) After creating new Cluster click on it and check its VM Instances option.

5) Click on the SSH button present alongside the Master Node.

6) It will open up a console where we will write further commands.

---------------------------------------------------------------------------

Setting up a Common user for team members to make tables and run Queries:-

---------------------------------------------------------------------------

1) Go to root user by typing 'su -i' in console.

2) Write the command 'adduser username' to create a new user.

3) It will ask to create a new password.

4) It will ask for more data just press Enter and press 'Y' when it asks.

5) Now the user is created.

6) To login as that user type 'su - username' and type in the password.

---------------------------------------------

Setting up Environment for Spark sql

---------------------------------------------

1) Now that we are logged in as common user we start the spark sql.

2) We need to set up a configuration file from running spark sql.

3) Write command 'nano log4j.properties'. This will launch a text editor.

4) Copy and paste in the following lines into the Nano text editor window:

# Set everything to be logged to the console

log4j.rootCategory=WARN, console

log4j.appender.console=org.apache.log4j.ConsoleAppender

log4j.appender.console.target=System.err

log4j.appender.console.layout=org.apache.log4j.PatternLayout

log4j.appender.console.layout.ConversionPattern=%d{yy/MM/dd HH:mm:ss} %p %c{1}: %m%n

# Settings to quiet third party logs that are too verbose

log4j.logger.org.eclipse.jetty=WARN

log4j.logger.org.eclipse.jetty.util.component.AbstractLifeCycle=ERROR

log4j.logger.org.apache.spark.repl.SparkIMain$exprTyper=WARN

log4j.logger.org.apache.spark.repl.SparkILoop$SparkILoopInterpreter=WARN

5) Press CTRL+X then press Y and then hit Enter to save the file.

6) Write command 'spark-sql --driver-java-options "-Dlog4j.configuration=file:///home/myusername/log4j.properties"'

7) Now you are inside the spark sql shell.

-------------------------------------------

Loading Data into Spark sql from Bucket

-------------------------------------------

Google Cloud Storage link -> https://console.cloud.google.com/storage/browser/cloud-cs-643

gs://cloud-cs-643

1) Now we have to write queries for creating local tables and load data into them

2) The code for loading data is provided in 'Source Code' file.

-------------------------------------------

Running Queries for generating Insights

-------------------------------------------

1) After loading data we run some queries to gather relevant information.

2) The Code for the Queries is provided in 'Source Code' file.

-----------------------------------------------------

Generating Charts from results to visualize data

-----------------------------------------------------

1) We copy the results generated from the queries and store them into an excel sheet.

2) We used the chart option available in Google Sheets to generate the Graphs

-------------------------------------------

Checking Fault Tolerance and Scalability

-------------------------------------------

1) To check Scalability we tried to increase the number of worker nodes and ran the same query to note time difference.

2) To Check Fault Tolerance we manually started closing the worker nodes while running query simultaneously and noted

the time difference and completeness of the result

--------------------------------------------------------------

Individual Contribution

--------------------------------------------------------------

Dhwanil Desai

* Worked on Insight 2 and 3
* Did performance Analysis on Fault Tolerance
* Cluster Configuration
* Data Processing and Cleaning

Valmik Kalathia

* Worked on Insight 4
* Helped with Visualisation
* Cluster Configuration

Samarth Desai

* Worked on Insight 1
* Helped with Visualisation
* Cluster Configuration
* Data Processing and Cleaning

Himanshu Patel

* Worked on Insights 5
* Did performance Analysis on Scalability
* Cluster Configuration
* Data Processing and Cleaning