|  |
| --- |
| Big Data Analytics – Advanced (Industry Assignment – 1) |
| Big Data Analytics – Advanced (Industry Assignment – 1)  Title: *Job Position and Trends Analysis*  Duration : 15 hours (Approx).  Technology to be used:   1. Apache Hadoop with HDFS commands and MapReduce. 2. HIVE, HIVE JDBC Driver. 3. Apache Kafka 4. Java/Python programming for connectivity and interface.   Dataset to be used:  CSV file: listings2019\_2022  Background of Work:  A Talent Provider company is constantly working on the current job market trends, technologies, potential recruiters, and other details as required for complete analysis. The idea is to recruit relevant profiles and/or develop workforce to meet current job market requirement and supply manpower to recruiting companies. Assume yourself to be a Big Data Analyst of the company.  Statement of Work:   1. **Create a directory on HDFS with the name “Analysis”.**   step 1: To configure docker  /home/user34/Desktop/BigData$ sudo docker -compose up    step 2: Now we need to enter into hadoop cluster using command  /home/user34/Desktop/BigData$ sudo docker exec -it namenode bash  step 3: Now we can create directory 'Analysis' in HDFS  root@573d8339fa78:/# hdfs dfs –mkdir /Analysis  step 4: Display list of directories and check whether 'Analysis' created or not     1. **Load the dataset to the directory as above using HDFS commands.**   step 1: we use hive to load dataset for that we must start hive  /home/user34/Desktop/BigData$ sudo docker exec -it hive-server bash    step 2: Now go to the path where dataset is and perform command to display it in local system  /home/user34/Desktop/Assignment1$ docker container ls  /home/user34/Desktop/Assignment1$ sudo docker cp./listings2019\_2022.csv a0a9d611810b:/listings2019\_2022.csv    step 3: Now go to root node and check whether dataset is present or not  **root@a0a9d611810b:/# ls**  step 4: Copy dataset to 'Analysis' directory  root@a0a9d611810b:/# hdfs dfs -copyFromLocal /listings2019\_2022.csv /Analysis    step 5: To check whether dataset copied to directory  root@a0a9d611810b:/# hdfs dfs -ls /Analysis     1. **jAnalyze the fields/columns in the .csv file and create tables in HIVE appropriately as required for analysis.**   step 1: Firstly enter into hive command  /home/user34/Desktop/BigData$ sudo docker exec -it hive-server bash  root@a0a9d611810b:/opt# hive  step 2: Create table (job) with necessary fields for analysis  hive> CREATE TABLE IF NOT EXISTS JOB(jobId STRING,jobClassification STRING,jobTitle STRING,listingDate TIMESTAMP,expiryDate TIMESTAMP,R INT,Python INT,Matlab INT,SQL INT,Stata INT,Minitab INT,SPSS INT,Ruby INT,C INT,Scala INT,Tableau INT,Java INT,Hadoop INT,SAS INT,Julia INT,Knime INT,D3 INT,Clojure INT,Haskell INT,Lisp INT,Golang INT,Spark INT,Javascript INT,F  INT,Fortran STRING,recruiter INT, year INT, CompanyName STRING)COMMENT 'JOB INFO' ROW FORMAT DELIMITED FIELDS TERMINATED BY "," STORED AS TEXTFILE;  step 3: To show the tables whether table created or not  hive> show tables;  step 4: To describe the table and its datatypes  hive> describe job;     1. **Load the dataset from HDFS to HIVE tables created.**   step 1: Load file from hdfs to hive use command hive> load data inpath '/Analysis/listings2019\_2022.csv' overwrite into table job;step 2: To display number of records in datasethive> select count(\*) from job;    1. **Use Select query to test the data in Hive tables and perform JOIN operations to extract relevant information related to:** 2. Year wise count of technologies being recruited.  Query 1: hive> select year,count(jobClassification),recruiter from job group by year,recruiter;     in above query, output 0 stands for not recruited and 1 stands for recruited   1. Companies recruiting the most year-wise.   Query 2: hive> select companyName,year, count(recruiter) as cnt from job where recruiter=1 group by CompanyName,year sort by cnt desc limit 10;  1. Top 5 JobTitles being recruited year-wise.   Query 3:  **hive> select year,jobTitle,count(recruiter) as cnt from job where recruiter=1 and year=2022 group by year,jobTitle,recruiter order by year desc limit 5 ;**       1. Develop a HIVE JDBC program using HiveServer2 to connect to HIVE tables created and perform the following operations: 2. Develop a CLI based interface for any ad-hoc query resolution, where users can type HIVE SQL queries and it would display the output. 3. Make provision for insertion of data from user console. 4. Update and correct data in HIVE table(s) based on user interaction.   step 1: Enter into beeline  root@c24098fba847:/# beeline  step 2: developing hive jdbc program into hiveserver2  beeline> !connect jdbc:hive2://localhost:10000  step 3: Now we are into local host 10000, perform command to display whether table present or not  0: jdbc:hive2://localhost:10000> show tables;    here we can see job table is present  a) Develop a CLI based interface for any ad-hoc query resolution, where users can type HIVE SQL queries and it would display the output.  From step 3 we got  0: jdbc:hive2://localhost:10000>  so we can perform hive queries on beeline,  Query 1:  jdbc:hive2://localhost:10000> select year,jobTitle,count(recruiter) as cnt from job where recruiter=1 and year=2022 group by year,jobTitle,recruiter order by year desc limit 5 ;    In above query we performed same query which we performed in hive to display top 5 job titles for year 2022  b) Make provision for insertion of data from user console. Query 2:  0: jdbc:hive2://localhost:10000> INSERT INTO job  VALUES(55701106,"Information & Communication Technology","data scientist",16-01-2019,26-08-  2019,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2020,"TCS");  c)Update and correct data in HIVE table(s) based on user interaction.  Query 3:  update job set jobtitle='AI Resercher' where jobClassification=Science & Technology;    The above code will run properly if changes are made to hive-site.xml, in order to avoid any consequences, I am not trying that here |
|  |

Vaishnavi Bhambure

vaishnavisbhambure@gmail.com