Installation Apache PIG

I. Install JAVA

1. First, we have to Install JDK in Linux. For that purpose, the following command will be executed.

   $ sudo apt install default-jdk

2. At last, the JRE File of Java will be installed using the following command.

   $ sudo apt install default-jre

3. To verify the installation, the following command you can use. It will prompt the Java Version used there.

   $ java -version

II. Install Hadoop

1. Update your system. Below are the 2 commands to update your system.

   $ sudo apt-get update

   $ sudo apt-get install update

2. Now download the package that you will going to install.

   https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz

3. Once you have download hadoop-3.4.0.tar.gz, extract this file with below command.

   $ sudo tar xvzf hadoop-3.4.0.tar.gz

4. Now navigate inside the folder using the below command.

   $ cd hadoop-3.4.0/

5. Create and open a new ***bash.sh*** file inside the directory.

   $ gedit bash.sh

6. We configure file, copy the below command inside this file and save it.

   export JAVA_HOME=$(readlink -f $(which javac) | awk 'BEGIN {FS="/bin"} {print $1}')

   export PATH=$(echo $PATH):$(pwd)/bin

   export CLASSPATH=$(hadoop classpath)

7. Execute the bash.sh File using following command

   $ source bash.sh

8. Verify ***JAVA_HOME*** variable to be set to Java Path and ***PATH*** variable has your Hadoop Folder.

9. Verify Hadoop is Installed or not by executing hadoop command. If command gives Information about Hadoop command, then Hadoop is Successfully Installed.

III. Install PIG

1. Download the new release of Apache Pig from the below link. In my case I have downloaded the pig-0.17.0.tar.gz version of Pig which is latest and about 220MB in size.

   https://downloads.apache.org/pig/pig-0.17.0/

2. Now we extract this tar file with the help of below command (make sure to check your tar filename).

   $ tar -xvf pig-0.17.0.tar.gz

3. Create and open a new bash.sh file inside the directory.

   $ gedit bash.sh

4. We configure file, copy the below command inside this file and save it.

   export PIG_INSTALL=$(pwd)

   export PATH=$PATH:$(pwd)/bin

5. Execute the bash.sh File using following command

   $ source bash.sh

6. You can check your pig version with the below command.

   $ pig -version

7. Once you get it correct that's it we have successfully install pig to our Hadoop single node setup, now we start pig with below pig command.

   $ pig

PIG Grunt Queries

I. Crop Production Dataset

1. Load the dataset

- crop_prod = LOAD 'crop_production.csv' USING PigStorage(',') AS (State_Name:chararray, District_Name:chararray, Crop_Year:int, Season:chararray, Crop:chararray, Area:float, Production:float);
- DESCRIBE crop_prod;

2. Calculate the total production of each crop

- total_production = GROUP crop_prod BY Crop;
- sum_production = FOREACH total_production GENERATE group AS Crop, SUM(crop_prod.Production) AS Total_Production;
- DUMP sum_production;

3. Find the average production per year for each crop

- grouped_by_crop_year = GROUP crop_prod BY (Crop, Crop_Year);
- average_production = FOREACH grouped_by_crop_year GENERATE group.Crop AS Crop, group.Crop_Year AS Crop_Year, AVG(crop_prod.Production) AS Avg_Production;
- DUMP average_production;

4. List all the crops grown in a specific state (e.g., 'Andaman and Nicobar Islands')

- specific_state = FILTER crop_prod BY State_Name == 'Andaman and Nicobar Islands';
- unique_crops = GROUP specific_state BY Crop;
- DUMP unique_crops;

5. Calculate the total area used for each crop in a specific year (e.g., 2000)

- specific_year = FILTER crop_prod BY Crop_Year == 2000;
- total_area = GROUP specific_year BY Crop;
- sum_area = FOREACH total_area GENERATE group AS Crop, SUM(specific_year.Area) AS Total_Area;
- DUMP sum_area;

II. Exams Dataset

1. Load the dataset

- exams = LOAD 'exams.csv' USING PigStorage(',') AS (gender:chararray, race_ethnicity:chararray, parental_level_of_education:chararray, lunch:chararray, test_preparation_course:chararray, math_score:int, reading_score:int, writing_score:int);
- DESCRIBE exams;

2. Count the Number of Students in Each Race/Ethnicity Group

- grouped_by_race = GROUP exams BY race_ethnicity;
- count_students_by_race = FOREACH grouped_by_race GENERATE group AS race_ethnicity, COUNT(exams) AS student_count;
- DUMP count_students_by_race;

3. Concatenate Gender and Parental Level of Education for Each Record

- concatenated_fields = FOREACH exams GENERATE CONCAT(gender, ' - ', parental_level_of_education) AS gender_education;
- DUMP concatenated_fields;

4. List all the unique parental levels of education

- unique_education_levels = GROUP exams BY parental_level_of_education;
- DUMP unique_education_levels;

III. Iris dataset

1. Load the dataset

- iris = LOAD 'iris.csv' USING PigStorage(',') AS (sepal_length:float, sepal_width:float, petal_length:float, petal_width:float, species:chararray);
- DESCRIBE iris;

2. Calculate the average sepal length for each species

- grouped_by_species = GROUP iris BY species;
- average_sepal_length = FOREACH grouped_by_species GENERATE group AS species, AVG(iris.sepal_length) AS avg_sepal_length;
- DUMP average_sepal_length;

3. Find the maximum petal width for each species

- grouped_by_species = GROUP iris BY species;
- max_petal_width = FOREACH grouped_by_species GENERATE group AS species, MAX(iris.petal_width) AS max_petal_width;
- DUMP max_petal_width;

4. List all the unique species in the dataset

- unique_species = GROUP iris BY species;
- DUMP unique_species;

IV. Olympic Athletes Dataset

1. Load the dataset

- athletes = LOAD 'olympic_athletes.csv' USING PigStorage(',') AS (athlete_url: chararray, athlete_full_name: chararray, games_participations: int, first_game: chararray, athlete_year_birth: float, athlete_medals: chararray, bio: chararray);
- DESCRIBE athletes;

2. Filter athletes who participated in the "Beijing 2022" games

- beijing_2022_athletes = FILTER athletes BY first_game == 'Beijing 2022';
- DUMP beijing_2022_athletes;

3. Group athletes by the number of game participations and count them

- grouped_by_participations = GROUP athletes BY games_participations;
- counted_participations = FOREACH grouped_by_participations GENERATE group AS games_participations, COUNT(athletes) AS num_athletes;
- DUMP counted_participations;

4. Filter athletes who have won medals

- medalists = FILTER athletes BY athlete_medals IS NOT NULL;
- DUMP medalists;

V. Olympic Hosts Dataset

1. Load the dataset

- hosts = LOAD 'olympic_hosts.csv' USING PigStorage(',') AS (game_slug: chararray, game_end_date: chararray, game_start_date: chararray, game_location: chararray, game_name: chararray, game_season: chararray, game_year: int);
- DESCRIBE hosts;

2. Filter the games held in "China"

- games_in_china = FILTER hosts BY game_location == 'China';
- DUMP games_in_china;

3. Group games by season and count the number of games in each season

- grouped_by_season = GROUP hosts BY game_season;
- counted_by_season = FOREACH grouped_by_season GENERATE group AS game_season, COUNT(hosts) AS num_games;
- DUMP counted_by_season;

4. Filter games that occurred after the year 2000

- games_after_2000 = FILTER hosts BY game_year > 2000;
- DUMP games_after_2000;