**Installation Apache PIG**

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

**$ sudo apt install default-jdk**

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

**$ sudo apt install default-jre**

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

**$ java -version**

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

**$ sudo apt-get update**

**$ sudo apt-get install update**

1. 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>

1. 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**

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

**$ cd hadoop-3.4.0/**

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

**$ gedit bash.sh**

1. 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)**

1. Execute the bash.sh File using following command

**$ source bash.sh**

1. Verify ***JAVA\_HOME*** variable to be set to Java Path and ***PATH*** variable has your Hadoop Folder.
2. Verify Hadoop is Installed or not by executing hadoop command. If command gives Information about Hadoop command, then Hadoop is Successfully Installed.
3. **Install PIG**
4. 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/>

1. 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**

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

**$ gedit bash.sh**

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

**export PIG\_INSTALL=$(pwd)**

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

1. Execute the bash.sh File using following command

**$ source bash.sh**

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

**$ pig -version**

1. 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**

1. **Crop Production Dataset**
2. 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;

1. 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;

1. 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;

1. 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;

1. 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;

1. **Exams Dataset**
2. 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;

1. 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;

1. 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;

1. List all the unique parental levels of education

* unique\_education\_levels = GROUP exams BY parental\_level\_of\_education;
* DUMP unique\_education\_levels;

1. **Iris dataset**
2. 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;

1. 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;

1. 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;

1. List all the unique species in the dataset

* unique\_species = GROUP iris BY species;
* DUMP unique\_species;

1. **Olympic Athletes Dataset**
2. 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;

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

* beijing\_2022\_athletes = FILTER athletes BY first\_game == 'Beijing 2022';
* DUMP beijing\_2022\_athletes;

1. 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;

1. Filter athletes who have won medals

* medalists = FILTER athletes BY athlete\_medals IS NOT NULL;
* DUMP medalists;

1. **Olympic Hosts Dataset**
2. 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;

1. Filter the games held in "China"

* games\_in\_china = FILTER hosts BY game\_location == 'China';
* DUMP games\_in\_china;

1. 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;

1. Filter games that occurred after the year 2000

* games\_after\_2000 = FILTER hosts BY game\_year > 2000;
* DUMP games\_after\_2000;