PIG DOCUMENTATION

- By VIGNESH.R

Apache Pig

Apache Pig is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs. The salient property of Pig programs is that their structure is amenable to substantial parallelization, which in turns enables them to handle very large data sets.

Key properties:

- **Ease of programming.** It is trivial to achieve parallel execution of simple, "embarrassingly parallel" data analysis tasks. Complex tasks comprised of multiple interrelated data transformations are explicitly encoded as data flow sequences, making them easy to write, understand, and maintain.
- **Optimization opportunities.** The way in which tasks are encoded permits the system to optimize their execution automatically, allowing the user to focus on semantics rather than efficiency.
- **Extensibility.** Users can create their own functions to do special-purpose processing.

Apache Pig Vs MapReduce

Apache Pig	MapReduce
Data flow language.	Data processing paradigm.
High level language.	Low level and rigid.
Performing a Join is pretty simple.	It is quite difficult to perform a Join operation between datasets.
Uses multi-query approach, thereby reducing the length of the codes.	Require almost 20 times more the number of lines to perform the same task.
There is no need for compilation.	MapReduce jobs have a long compilation process.

Apache Pig Vs Hive

Apache Pig	Hive
Apache Pig uses a language called Pig Latin.	Hive uses a language called HiveQL.
Pig Latin is a data flow language.	HiveQL is a query processing language.
Pig Latin is a procedural language.	HiveQL is a declarative language.
Apache Pig can handle structured, unstructured, and semi-structured data.	Hive is mostly for structured data.

Apache Pig Mode

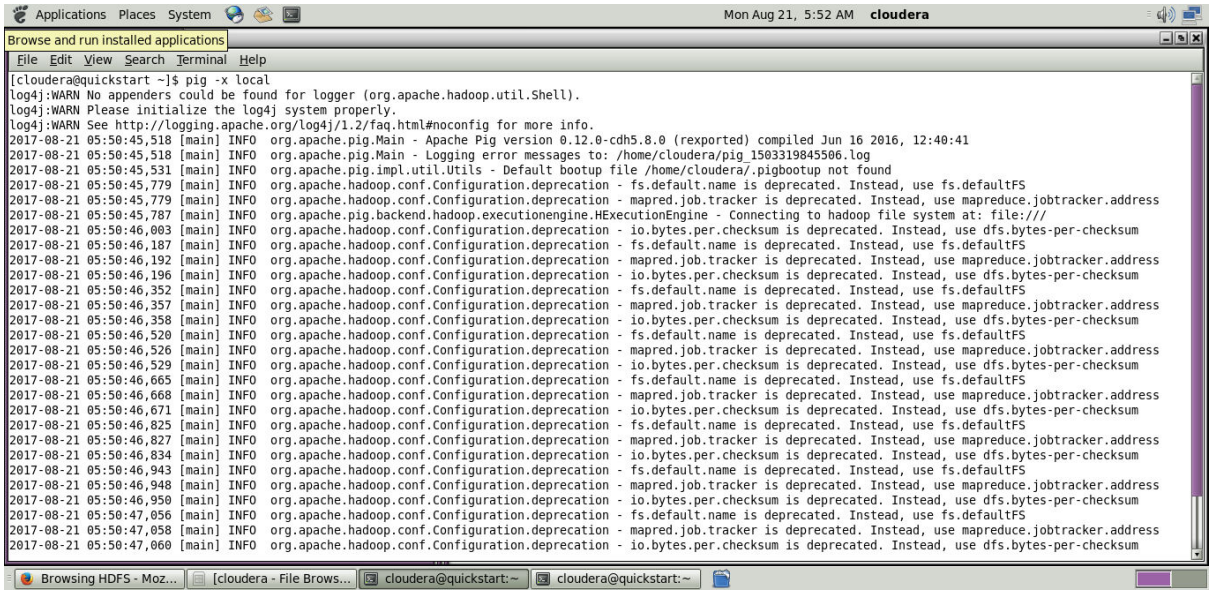
Apache Pig has two modes.

- **Local Mode** - All the files are installed and run from the local host and local file system.
- **HDFS mode** - MapReduce mode is where we load or process the data that exists in the Hadoop File System (HDFS) using Apache Pig.

Invoking the Grunt Shell:-

Command (Local mode) : pig -x local

Command (HDFS) : pig -x mapreduce



Apache Pig Execution Mechanisms

Apache Pig scripts can be executed in three ways, namely:-

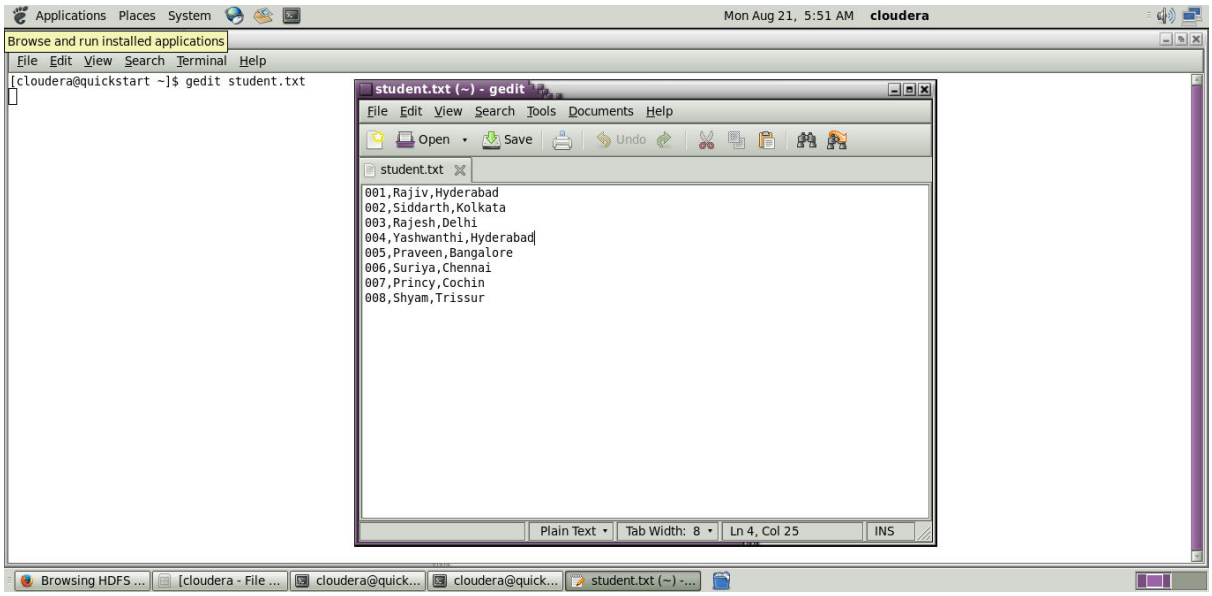
- **Interactive mode**
- **Batch mode**
- **Embedded mode.**

Interactive Mode (Grunt shell):-

After invoking the Grunt shell, you can execute a Pig script by directly entering the Pig Latin statements in it.

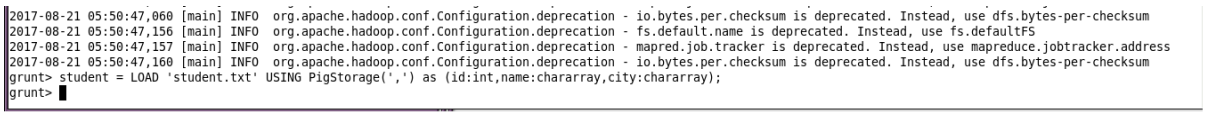
Step 1: Create a text file.

Command: gedit student.txt



Step 2: Load the data into grunt shell

Command: student = LOAD 'student.txt' USING PigStorage(',') as (id:int,name:chararray,city:chararray);



Step 3: Display the data.

Command: Dump student;

```
Success!

Job Stats (time in seconds):
JobId   Alias   Feature Outputs
job_local1613044507_0001        student MAP_ONLY        file:/tmp/temp-34040847/tmp1652691499,

Input(s):
Successfully read records from: "file:///home/cloudera/student.txt"

Output(s):
Successfully stored records in: "file:/tmp/temp-34040847/tmp1652691499"

Job DAG:
job_local1613044507_0001


2017-08-21 05:54:42,412 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2017-08-21 05:54:42,415 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
2017-08-21 05:54:42,415 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2017-08-21 05:54:42,415 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-08-21 05:54:42,416 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-08-21 05:54:42,434 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-08-21 05:54:42,434 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(1,Rajiv,Hyderabad)
(2,Siddarth,Kolkata)
(3,Rajesh,Delhi)
(4,Yashwanthi,Hyderabad)
(5,Praveen,Bangalore)
(6,Suriya,Chennai)
(7,Princy,Cochin)
(8,Shyam,Trissur)
grunt>
```

Batch Mode (Script):-

We can write an entire Pig Latin script in a file and execute it using the –x command.

Step 1: Create a file with .pig extension.

Command: gedit sample_pig.pig

```
[cloudera@quickstart ~]$ gedit sample_pig.pig
```

```
student = LOAD 'student.txt' USING PigStorage(',') as
(id:int,name:chararray,city:chararray);

Dump student;
```

Step 2: Paste the following command into that file.

Command: student = LOAD 'student.txt' USING PigStorage(',') as (id:int,name:chararray,city:chararray);

Dump student;

Step 3: Run from terminal.

Command: exec /home/cloudera/sample_pig.pig;

```
Success!

Job Stats (time in seconds):
JobId   Alias   Feature Outputs
job_local184040564_0001 student MAP_ONLY        file:/tmp/temp1571375856/tmp-573656058,

Input(s):
Successfully read records from: "file:///home/cloudera/student.txt"

Output(s):
Successfully stored records in: "file:/tmp/temp1571375856/tmp-573656058"

Job DAG:
job_local184040564_0001


2017-08-21 05:56:36,716 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2017-08-21 05:56:36,718 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
2017-08-21 05:56:36,718 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2017-08-21 05:56:36,718 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-08-21 05:56:36,719 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-08-21 05:56:36,736 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-08-21 05:56:36,736 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(1,Rajiv,Hyderabad)
(2,Siddarth,Kolkata)
(3,Rajesh,Delhi)
(4,Yashwanthi,Hyderabad)
(5,Praveen,Bangalore)
(6,Suriya,Chennai)
(7,Princy,Cochin)
(8,Shyam,Trissur)
grunt>
```

Diagnostic Operators

i. **DUMP** - To print the contents of a relation on the console.
   Command : DUMP student;

ii. **DESCRIBE** - To describe the schema of a relation.
   Command : Describe student;

iii. **EXPLAIN** - To view the logical, physical, or MapReduce execution plans to compute a relation.
   Command : Explain student;

iv. **ILLUSTRATE** - To view the step-by-step execution of a series of statements.
   Command : Illustrate student;

Pig Latin – Relational Operations

1. **LOAD** - To Load the data from the file system (local/HDFS) into a relation.
   Command : student = LOAD 'student.txt' USING PigStorage(',') as (id:int,name:chararray,city:chararray);
   Dump student;

```
2017-08-21 05:50:47,060 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-08-21 05:50:47,156 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
2017-08-21 05:50:47,157 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2017-08-21 05:50:47,160 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
grunt> student = LOAD 'student.txt' USING PigStorage(',') as (id:int,name:chararray,city:chararray);
grunt>
```

```
(1,Rajiv,Hyderabad)
(2,siddarth,Kolkata)
(3,Rajesh,Delhi)
(4,Yashwanthi,Hyderabad)
(5,Praveen,Bangalore)
(6,Suriya,Chennai)
(7,Princy,Cochin)
(8,Shyam,Trissur)
grunt>
```

2. **STORE** - To save a relation to the file system (local/HDFS).
   Command : STORE student INTO '/home/cloudera/pig_output' USING PigStorage (',');

```
Applications  Places  System                                   Mon Aug 21, 10:41 AM   cloudera
Browse and run installed applications        cloudera@quickstart:~
792122_0030_m_000000_0' to file:/home/cloudera/pig_output/_temporary/0/task_local1741792122_0030_m_000000
2017-08-21 10:41:16,844 [LocalJobRunner Map Task Executor #0] INFO  org.apache.hadoop.mapred.LocalJobRunner - map
2017-08-21 10:41:16,844 [LocalJobRunner Map Task Executor #0] INFO  org.apache.hadoop.mapred.Task - Task 'attempt_local1741792122_0030_m_000000_0' done.
2017-08-21 10:41:16,844 [LocalJobRunner Map Task Executor #0] INFO  org.apache.hadoop.mapred.LocalJobRunner - Finishing task: attempt_local1741792122_0030_m_000000_0
2017-08-21 10:41:16,844 [Thread-174] INFO  org.apache.hadoop.mapred.LocalJobRunner - map task executor complete.
2017-08-21 10:41:17,036 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - HadoopJobId: job_local1741792122_0030
2017-08-21 10:41:17,036 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Processing aliases student
2017-08-21 10:41:17,036 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - detailed locations: M: student[22,10],student[-1,-
1] C:  R:
2017-08-21 10:41:17,038 [main] WARN  org.apache.pig.tools.pigstats.PigStatsUtil - Failed to get RunningJob for job job_local1741792122_0030
2017-08-21 10:41:17,038 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 100% complete
2017-08-21 10:41:17,038 [main] INFO  org.apache.pig.tools.pigstats.SimplePigStats - Detected Local mode. Stats reported below may be incomplete
2017-08-21 10:41:17,038 [main] INFO  org.apache.pig.tools.pigstats.SimplePigStats - Script Statistics:

HadoopVersion   PigVersion      UserId  StartedAt       FinishedAt      Features
2.6.0-cdh5.8.0  0.12.0-cdh5.8.0 cloudera        2017-08-21 10:41:16     2017-08-21 10:41:17     UNKNOWN

Success!

Job Stats (time in seconds):
JobId   Alias   Feature Outputs
job_local1741792122_0030        student MAP_ONLY        /home/cloudera/pig_output,

Input(s):
Successfully read records from: "file:///home/cloudera/student.txt"

Output(s):
Successfully stored records in: "/home/cloudera/pig_output"

Job DAG:
job_local1741792122_0030


2017-08-21 10:41:17,038 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
grunt>
New Tab - Mozilla Firef...   cloudera@quickstart:~   cloudera@quickstart:~   [cloudera - File Browser]
```

3. **GROUP** - To group the data in a single relation.
   Command : grouping = GROUP student1 by age;
   Dump grouping;

```
(21,{(4,Preethi,Agarwal,21,9848022330,Pune),(1,Rajiv,Reddy,21,9848022337,Hyderabad)})
(22,{(3,Rajesh,Khanna,22,9848022339,Delhi),(2,siddarth,Battacharya,22,9848022338,Kolkata)})
(23,{(6,Archana,Mishra,23,9848022335,Chennai),(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)})
(24,{(8,Bharathi,Nambiayar,24,9848022333,Chennai),(7,Komal,Nayak,24,9848022334,trivendram)})
grunt>
```

4. **JOIN** - To join two or more relations.

   Command : student3 = JOIN student1 BY id, student2 BY id;

   Dump student3;

```
(1,Rajiv,Reddy,21,9848022337,Hyderabad,1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata,2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi,3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune,4,Preethi,Agarwal,21,9848022330,Pune)
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar,5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(6,Archana,Mishra,23,9848022335,Chennai,6,Archana,Mishra,23,9848022335,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram,7,Komal,Nayak,24,9848022334,trivendram)
(8,Bharathi,Nambiayar,24,9848022333,Chennai,8,Bharathi,Nambiayar,24,9848022333,Chennai)
grunt>
```

5. **CROSS** - To create the cross product of two or more relations.

   Command : cross_data = CROSS customers, orders;

   Dump cross_data;

```
2017-08-21 10:44:19,441 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
2017-08-21 10:44:19,441 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2017-08-21 10:44:19,441 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-08-21 10:44:19,441 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-08-21 10:44:19,455 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-08-21 10:44:19,455 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(7,Muffy,24,Indore,10000,103,2008-05-20 00:00:00,4,2060)
(7,Muffy,24,Indore,10000,101,2009-11-20 00:00:00,2,1560)
(7,Muffy,24,Indore,10000,100,2009-10-08 00:00:00,3,1500)
(7,Muffy,24,Indore,10000,102,2009-10-08 00:00:00,3,3000)
(6,Komal,22,MP,4500,103,2008-05-20 00:00:00,4,2060)
(6,Komal,22,MP,4500,101,2009-11-20 00:00:00,2,1560)
(6,Komal,22,MP,4500,100,2009-10-08 00:00:00,3,1500)
(6,Komal,22,MP,4500,102,2009-10-08 00:00:00,3,3000)
(5,Hardik,27,Bhopal,8500,103,2008-05-20 00:00:00,4,2060)
(5,Hardik,27,Bhopal,8500,101,2009-11-20 00:00:00,2,1560)
(5,Hardik,27,Bhopal,8500,100,2009-10-08 00:00:00,3,1500)
(5,Hardik,27,Bhopal,8500,102,2009-10-08 00:00:00,3,3000)
(4,Chaitali,25,Mumbai,6500,103,2008-05-20 00:00:00,4,2060)
(4,Chaitali,25,Mumbai,6500,101,2009-11-20 00:00:00,2,1560)
(4,Chaitali,25,Mumbai,6500,100,2009-10-08 00:00:00,3,1500)
(4,Chaitali,25,Mumbai,6500,102,2009-10-08 00:00:00,3,3000)
(3,kaushik,23,Kota,2000,103,2008-05-20 00:00:00,4,2060)
(3,kaushik,23,Kota,2000,101,2009-11-20 00:00:00,2,1560)
(3,kaushik,23,Kota,2000,100,2009-10-08 00:00:00,3,1500)
(3,kaushik,23,Kota,2000,102,2009-10-08 00:00:00,3,3000)
(2,Khilan,25,Delhi,1500,103,2008-05-20 00:00:00,4,2060)
(2,Khilan,25,Delhi,1500,101,2009-11-20 00:00:00,2,1560)
(2,Khilan,25,Delhi,1500,100,2009-10-08 00:00:00,3,1500)
(2,Khilan,25,Delhi,1500,102,2009-10-08 00:00:00,3,3000)
(1,Ramesh,32,Ahmedabad,2000,103,2008-05-20 00:00:00,4,2060)
(1,Ramesh,32,Ahmedabad,2000,101,2009-11-20 00:00:00,2,1560)
(1,Ramesh,32,Ahmedabad,2000,100,2009-10-08 00:00:00,3,1500)
(1,Ramesh,32,Ahmedabad,2000,102,2009-10-08 00:00:00,3,3000)
grunt>
```

6. **UNION** - To combine two or more relations into a single relation.

   Command : union_out = UNION student1, student2;

   Dump union_out;

```
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune)
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(6,Archana,Mishra,23,9848022335,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram)
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune)
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(6,Archana,Mishra,23,9848022335,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram)
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
grunt>
```

7. **SPLIT** - To split a single relation into two or more relations.
   Command : SPLIT student1 into student_details1 if age<23, student_details2 if (age>=23);
   Dump student_details1;
   Dump student_details2;

```
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune)
grunt>
```

```
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(6,Archana,Mishra,23,9848022335,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram)
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
grunt>
```

8. **FILTER** - To remove unwanted rows from a relation.
   Command : filter_data = FILTER student1 BY city == 'Chennai';
   Dump filter_data;

```
(6,Archana,Mishra,23,9848022335,Chennai)
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
grunt>
```

9. **DISTINCT** - To remove duplicate rows from a relation.
   Command : student4 = LOAD 'student4.txt' USING PigStorage(',') as (id:int, firstname:chararray, lastname:chararray, age:int, phone:chararray, city:chararray);
   distinct_data = DISTINCT student4;
   Dump distinct_data;

```
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune)
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(6,Archana,Mishra,23,9848022335,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram)
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
grunt>
```

10. **FOREACH**- To generate data transformations based on columns of data.

Command : foreach_data = FOREACH student1 GENERATE id,age,city;

Dump foreach_data;

```
(1,21,Hyderabad)
(2,22,Kolkata)
(3,22,Delhi)
(4,21,Pune)
(5,23,Bhuwaneshwar)
(6,23,Chennai)
(7,24,trivendram)
(8,24,Chennai)
grunt>
```

11. **ORDER** - To arrange a relation in a sorted order based on one or more fields (ascending or descending).

Command : order_by_data = ORDER student1 BY age DESC;

Dump order_by_data;

```
(8,Bharathi,Nambiayar,24,9848022333,Chennai)
(7,Komal,Nayak,24,9848022334,trivendram)
(6,Archana,Mishra,23,9848022335,Chennai)
(5,Trupthi,Mohanthy,23,9848022336,Bhuwaneshwar)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(4,Preethi,Agarwal,21,9848022330,Pune)
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
grunt>
```

12. **LIMIT** - To get a limited number of tuples from a relation.

Command : limit_data = LIMIT student1 4;

Dump limit_data;

```
(1,Rajiv,Reddy,21,9848022337,Hyderabad)
(2,siddarth,Battacharya,22,9848022338,Kolkata)
(3,Rajesh,Khanna,22,9848022339,Delhi)
(4,Preethi,Agarwal,21,9848022330,Pune)
grunt>
```