UNIVERSITY PARTNER

UNIVERSITY OF WOLVERHAMPTON

HERALD COLLEGE KATHMANDU

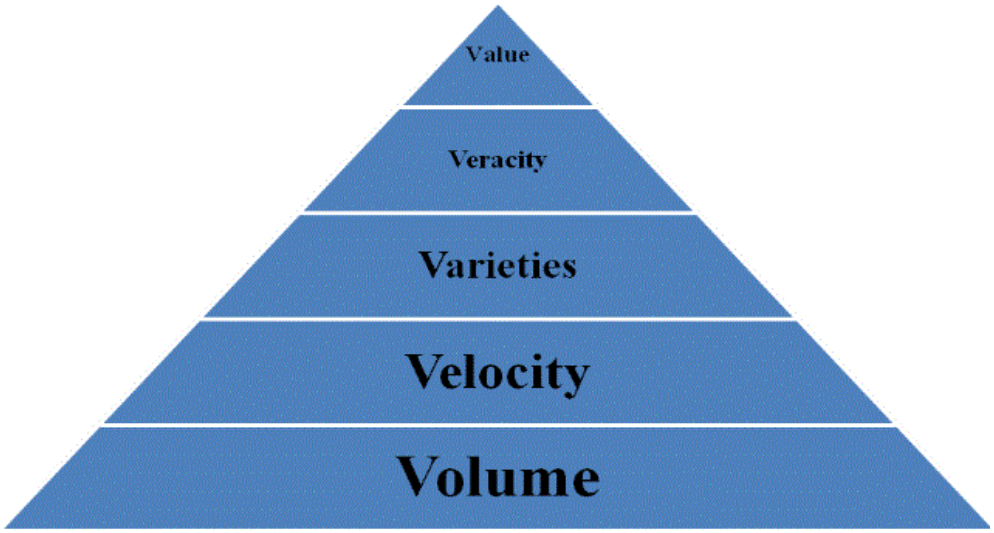
Big Data (6CS030)

Course Work

Student ID	:	2039224
Student Name	:	Tsering Wongdi Sherpa
Module Leader	:	Mr. Jnaneshwar Bohora
Submitted On	:	26th April 2021

1. Introduction to big data

Traditional structures and data warehousing tools are incapable of processing and working with Big Data because it is too large and dynamic. Machines, humans, and mother nature all contribute to the creation of data (Big Data). With the advancement in technology and services, vast amounts of structured, semi-structured, and unstructured data are generated from various sources. Big data can't be acted with standard SQL requests, and it can't be stored with a relational database management system. As a result, several different modular database tools and techniques have emerged. Hadoop is a distributed data analysis framework that is open source (Ishwarappa & Anuradha, 2015). There are five main caricaturists of big data which are given below:



fig; representation of 5V in big data

Volume:

Around 2012, businesses started processing more than three million pieces of data every day, and the amount of data they handle skyrocketed. “This amount has doubled almost every 40 months since then,” Herencia said.

Velocity:

Companies need data to flow efficiently, as close to real-time as possible, in addition to handling it. “Velocity can be more critical than volume because it can offer us a bigger competitive advantage,” the MetLife executive said. It's sometimes preferable

to get a small amount of data in real time rather than a large amount of data at a slow speed."

Variety:

Variety refers to the many forms of data that we can now use. We have various types of data. Structured data is data in which the schema has been specified in advance and obtained from different businesses. However, today's big data encompasses data with no fixed schema, such as audio, video, photographs, and social media notifications.

Veracity:

Anomalies, glitches, noise, and unfiltered data can be present in data contained in databases from various sources. The most important task of Big Data is to ensure the data's worthiness, consistency, and precision.

Value:

This is the pinnacle of the big data pyramid. A broad volume and variety of data that is easy to access, as well as quality analytics that make educated decisions, are all examples of value.

2. Introduction to datasets

2.1. Justification for choice

For this project two education sector related dataset are chosen. The CSV dataset contains the population aged 5 to 25 years old who are currently attending and not attending school, broken down by sex and age. The key goal of using this dataset is to assess and visualize the diversity of students aged 5 to 25 who attend school in different parts of Nepal. The JSON dataset includes the number of schools in each district, zone, ecological area, and growth region for each grade. This dataset was chosen to quantify and visualize the number of schools in each city and district. At last from the analysis of both the dataset new analysis can be taken place which will help to find out the different result of literacy rate, student to school ratio, education quality and economic condition of each region etc.

2.2. CSV datasets

The csv data collection that is used for this course is called "Peoples aged 5 to 25 years of age by attending, going to school, sex and age and not going to school." The following source is available for download from the Nepalese dataset website:
http://data.opennepal.net/content/population-aged-5-25-years-school-attendence-sex-and-age-who-are-currently-going-and-not

2.3. JSON datasets

The used json dataset used for this coursework is named as 'school numbers' which has 1650 collections which contains 6 key value pairs. Which is converted into json using csv to json converter tool the original dataset was named as 'Total number of schools by grade 2012-13' which contains 6 columns and 1550 rows which was downloaded from the Nepali dataset website which link is given below:
http://data.opennepal.net/content/total-number-schools-grade-201213

3. Importing/cleaning datasets

3.1. Cleaning and Importing CSV datasets

The key goal of this coursework is to review different school attendance data of different age groups using the chosen csv dataset. The csv file has been updated or cleaned for a successful outcome based on the objectives set..

A	B	C	D	E	F	G	H	I	J
District	Zone	Ecological	Development Region	Year (BS)	Year (AD)	Age-Group	Indicator	Sub-Indica	Value
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	Currently going to school	Male	2156
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	Currently going to school	Female	2035
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	Not currently going to school	Male	2081
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	Not currently going to school	Female	2052
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	School attendence not Stated	Male	161
Achham	Seti	Hill	Far-Western	2068/69	2011/12	05 Year	School attendence not Stated	Female	165
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	Currently going to school	Male	2996
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	Currently going to school	Female	2788
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	Not currently going to school	Male	1101
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	Not currently going to school	Female	1236
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	School attendence not Stated	Male	87
Achham	Seti	Hill	Far-Western	2068/69	2011/12	06 Year	School attendence not Stated	Female	117
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	Currently going to school	Male	3039
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	Currently going to school	Female	3053
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	Not currently going to school	Male	609
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	Not currently going to school	Female	685
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	School attendence not Stated	Male	59
Achham	Seti	Hill	Far-Western	2068/69	2011/12	07 Year	School attendence not Stated	Female	58

After removing the non-necessary column from dataset:

A	B	C	D	E	F	G	H
District	Zone	Ecological	Development Region	Age-Group	Indicator	Sub-Indicator	Value
Achham	Seti	Hill	Far-Western	05 Year	Currently going to school	Male	2156
Achham	Seti	Hill	Far-Western	05 Year	Currently going to school	Female	2035
Achham	Seti	Hill	Far-Western	05 Year	Not currently going to school	Male	2081
Achham	Seti	Hill	Far-Western	05 Year	Not currently going to school	Female	2052
Achham	Seti	Hill	Far-Western	05 Year	School attendence not Stated	Male	161
Achham	Seti	Hill	Far-Western	05 Year	School attendence not Stated	Female	165
Achham	Seti	Hill	Far-Western	06 Year	Currently going to school	Male	2996
Achham	Seti	Hill	Far-Western	06 Year	Currently going to school	Female	2788
Achham	Seti	Hill	Far-Western	06 Year	Not currently going to school	Male	1101
Achham	Seti	Hill	Far-Western	06 Year	Not currently going to school	Female	1236
Achham	Seti	Hill	Far-Western	06 Year	School attendence not Stated	Male	87
Achham	Seti	Hill	Far-Western	06 Year	School attendence not Stated	Female	117
Achham	Seti	Hill	Far-Western	07 Year	Currently going to school	Male	3039
Achham	Seti	Hill	Far-Western	07 Year	Currently going to school	Female	3053
Achham	Seti	Hill	Far-Western	07 Year	Not currently going to school	Male	609
Achham	Seti	Hill	Far-Western	07 Year	Not currently going to school	Female	685
Achham	Seti	Hill	Far-Western	07 Year	School attendence not Stated	Male	59
Achham	Seti	Hill	Far-Western	07 Year	School attendence not Stated	Female	58
Achham	Seti	Hill	Far-Western	08 Year	Currently going to school	Male	3792

Out of 10 columns only 8 were meaningful so the left 2 were cleaned.

Now, importing cleaned data into oracle SQL developer

3.2. Cleaning/importing JSON datasets

The original dataset was already cleaned and all the fields are required for the analysis so there is no need of cleaning the data.

*schoolnumbers - Notepad

File Edit Format View Help


```
[
  {
    "District": "Taplejung",
    "Zone": "Mechi",
    "Geographical_Region": "Mountain",
    "Development_Region": "Eastern",
    "Grade": 1,
    "Number_of_School": 3
  },
  {
    "District": "Taplejung",
    "Zone": "Mechi",
    "Geographical_Region": "Mountain",
    "Development_Region": "Eastern",
    "Grade": "(1-2)",
    "Number_of_School": 5
  },
  {
    "District": "Taplejung",
    "Zone": "Mechi",
    "Geographical_Region": "Mountain",
    "Development_Region": "Eastern",
    "Grade": "(1-3)",
    "Number_of_School": 58
  },
  {
    "District": "Taplejung",
    "Zone": "Mechi",
    "Geographical_Region": "Mountain",
    "Development_Region": "Eastern",
    "Grade": "(1-4)",
    "Number_of_School": 10
  },
  {
    "District": "Taplejung",
    "Zone": "Mechi",
    "Geographical_Region": "Mountain",
    "Development_Region": "Eastern",
    "Grade": "(1-5)",
    "Number_of_School": 138
  },
  {
    "District": "Taplejung",
    "Zone": "Mechi",
```


3.3. Database selection for datasets

The JSON dataset is used by MongoDB and Spark. Data is saved as documents in MongoDB. These papers are stored in MongoDB JSON format. The embedded fields in JSON documents allow for the saving of related data and data lists into the database instead of the external table. Spark enters a JSON dataset scheme and loads it dynamically as a data frame. On a JSON

file, use SparkSession.read.json to perform this conversion. Oracle and Hadoop are used for the CSV dataset. CSV dataset is structured dataset which can be stored as relational database.

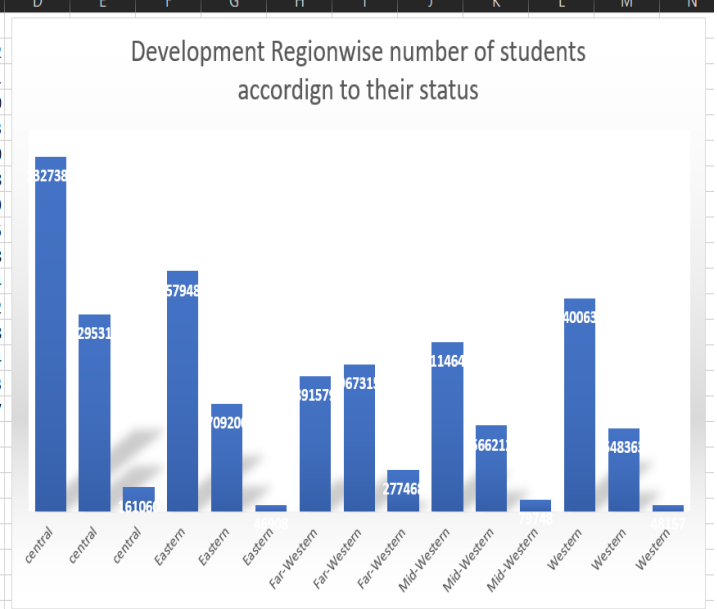
4. Analysis of the data and visualizations

4.1. Techniques used to query the data and its results

Different OLAP query (ROOLUP and CUBE), Wildcard query (like '%') and Distinct query for the analysis of the CSV dataset in Oracle. In Hadoop three different java file were used to find out the total number of students currently going to school for each development region, zone and district. For the JSON dataset in MongoDB functions like find, pretty, count and aggregate were used. In Spark query like select, sum, order by and group by were used at the end from both MongoDB and Spark output result of total number of schools in each development region, zone and district were exported for the visualization.

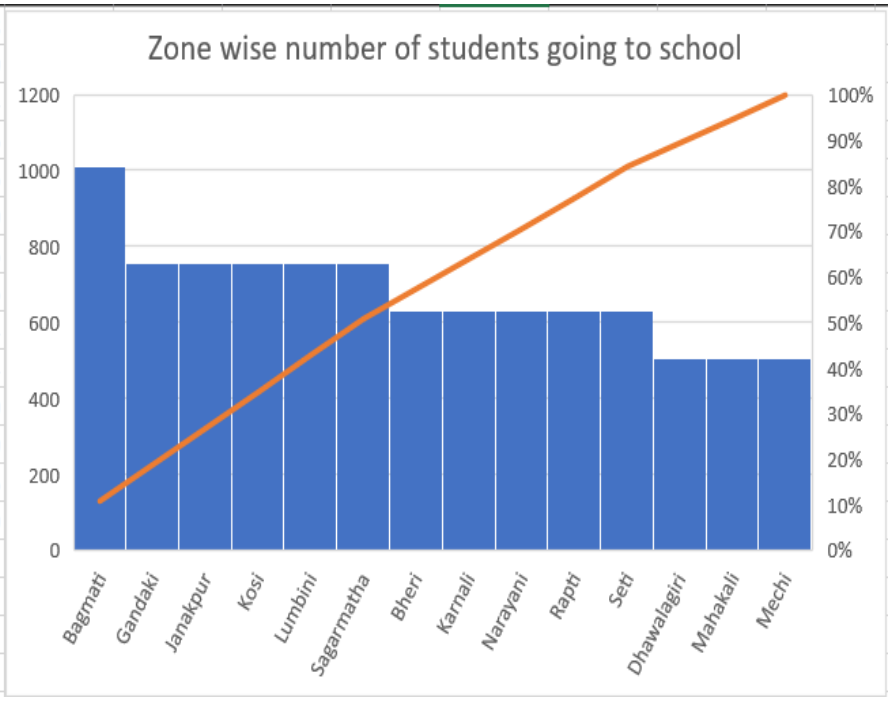
4.2. Visualization of the datasets

Development_Region	Status	Total
central	Currently going to school	2327382
central	Not currently going to school	1295311
central	School attendence not Stated	161060
Eastern	Currently going to school	1579483
Eastern	Not currently going to school	709200
Eastern	School attendence not Stated	46908
Far-Western	Currently going to school	891579
Far-Western	Not currently going to school	967315
Far-Western	School attendence not Stated	277468
Mid-Western	Currently going to school	1114644
Mid-Western	Not currently going to school	566212
Mid-Western	School attendence not Stated	79748
Western	Currently going to school	1400634
Western	Not currently going to school	548363
Western	School attendence not Stated	48157



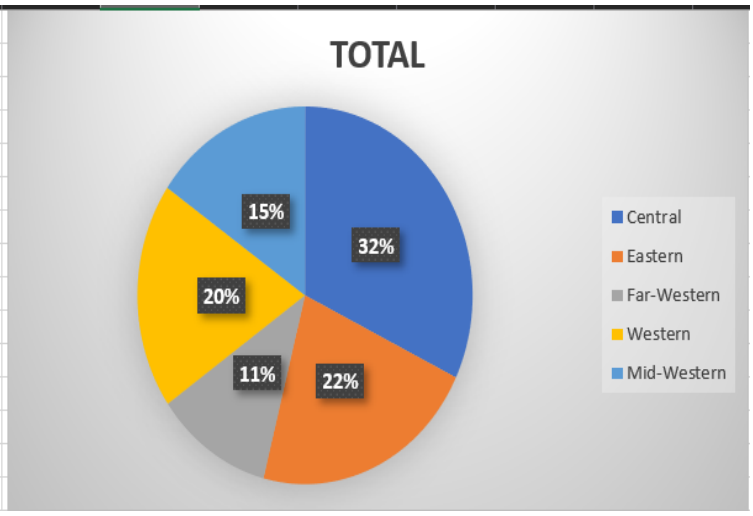
fig; Development region wise number of students according to their status

Bagmati	1008
Bheri	630
Dhawalagi	504
Gandaki	756
Janakpur	756
Karnali	630
Kosi	756
Lumbini	756
Mahakali	504
Mechi	504
Narayani	630
Rapti	630
Sagarmath	756
Seti	630

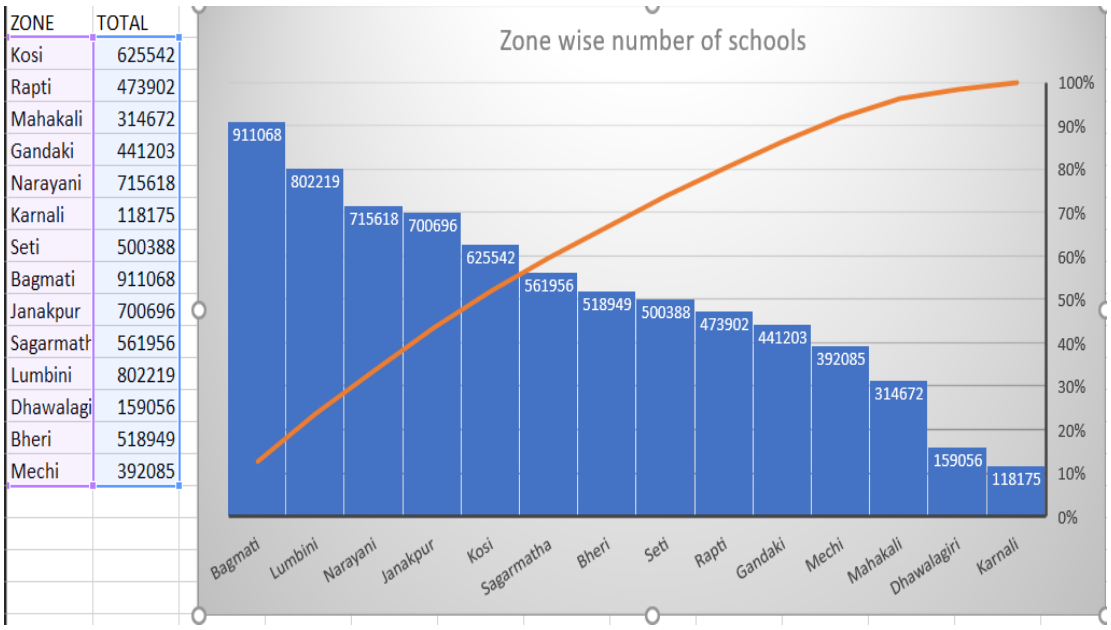


fig; District wise total number of students currently going to school

DEVELOPMENT_REGION	TOTAL
Central	2327382
Eastern	1579583
Far-Western	815060
Western	1402478
Mid-Western	1111026



fig; Development region wise total number of schools



fig; Zone wise number of schools

5. Comparison table

5.1. Pros and cons of using Oracle for big data

Advantages	Disadvantages
It is available on more channels than any of its competitors. It operates on nearly 20 networking protocols as well as over 100 hardware platforms with ease.	One of Oracle's major disadvantages is its difficulty. Oracle is not recommended, particularly if the users are not technically skillful and do not have the technology to deal with the Oracle Database.
With the aid of an Oracle database, a point-in-time recovery can be accomplished quickly.	Oracle products can cost up to ten times more than the mid-range MS SQL server database system.

With transaction monitoring and locking, it increases the efficiency and speed of consideration.	Only if large databases are needed is useful. For small or media companies using smaller data sets, it is not recommended.
It handles several databases in the same transaction with ease.	It is much more complex and demanding when dealing with individual tasks.

5.2. Pros and cons of using MongoDB for big data

Advantages	**Disadvantages**
It is document oriented database. Indexing makes it easy to find records. As a result, it responds to queries quickly. MongoDB is 100 times faster than traditional relational databases.	It does not allow joins in the same way as a relational database does. However, one can use joins features by manually coding it. However, it can slow down execution and have an impact on results.
It has replication and gridFS capabilities. These features contribute to MongoDB's increased data availability. As a result, the efficiency is excellent.	It keeps track of the key names for each value pair. There is also data duplication due to the lack of join capability. As a consequence, memory use increases unnecessarily.
MongoDB has the benefit of being a horizontally scalable database. It is possible to	It can have a maximum document size of 16MB.

scatter a huge amount of data through many computers in order to handle it.	
MongoDB is simpler to set up than RDBMS. For requests, it even has a JavaScript database.	More than 100 level of document nesting are not possible. It does not have transaction support.

5.3. Pros and cons of using Hadoop for big data

Advantages	**Disadvantages**
It is open-source and runs on low-cost commodity hardware, resulting in a cost-effective architecture.	It is a framework written in Java, which is one of the most widely used programming languages, making it more vulnerable so any cyber-criminal can easily hack it.
It's a model with a lot of scalability. A vast volume of data is split from several low-cost machines in a cluster and processed in parallel.	It fails where a large number of small files must be accessed. The Namenode is overburdened with too many tiny images, making it impossible to deal with.
It can efficiently handle any kind of data regardless of its structure, making it extremely versatile.	Its security function is Kerberos, which is difficult to handle. Kerberos lacks storage and network encryption, which leaves us much more worried.

It is faster compared to traditional database management system.	In Hadoop, data is read or written from disk, making in-memory calculations impossible and resulting in computing overhead or high up processing.
Data is mirrored across several DataNodes in a Hadoop cluster, ensuring data stability even though one or more systems fail.	It is mostly intended for dealing with big databases, but it can be effectively used by businesses who generate a large amount of data. When working in a tiny data setting, the productivity suffers.

Conclusion

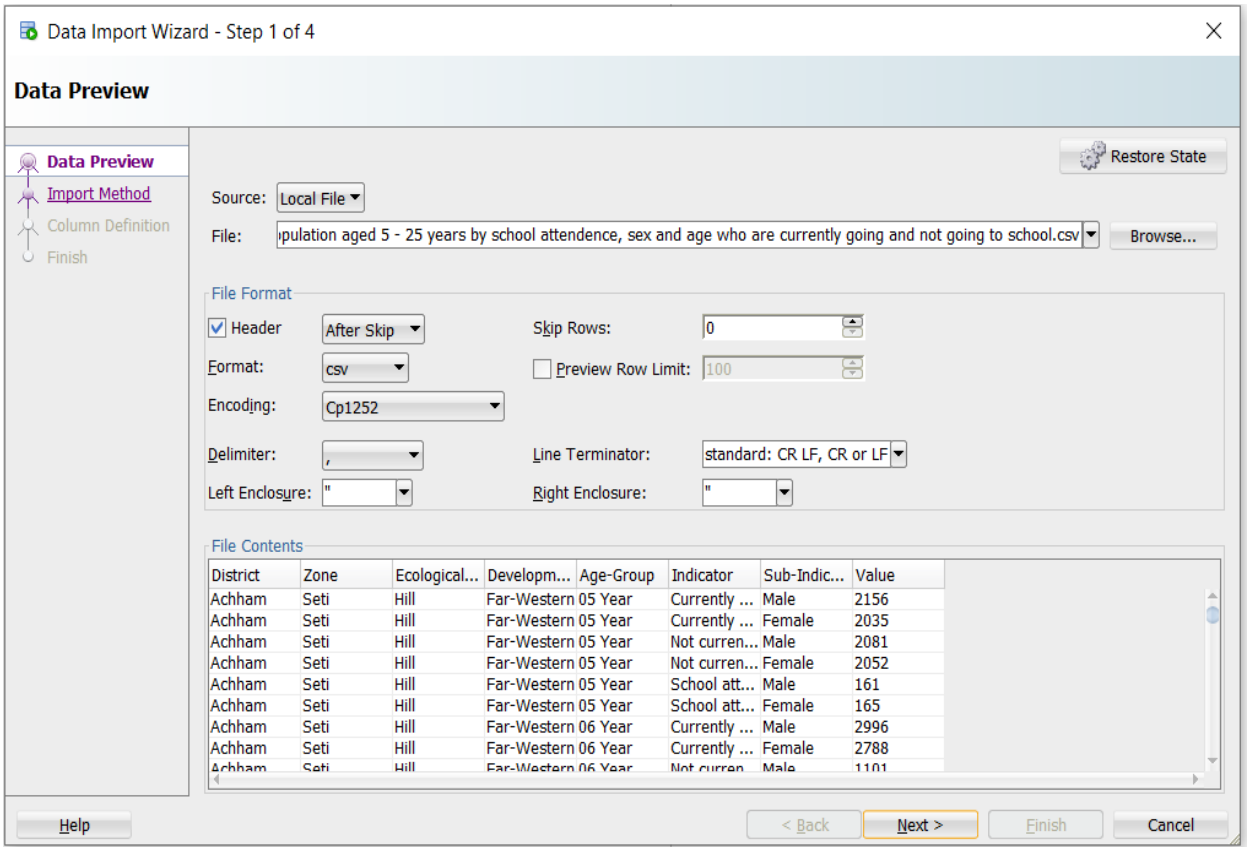
In this project two Nepali education sector related csv and json datasets were taken. After that different database system like Oracle, MongoDB and Hadoop were used to analyze the datasets. Number of schools and students of each district, zone and development region were calculated as a result after analyzing those datasets.

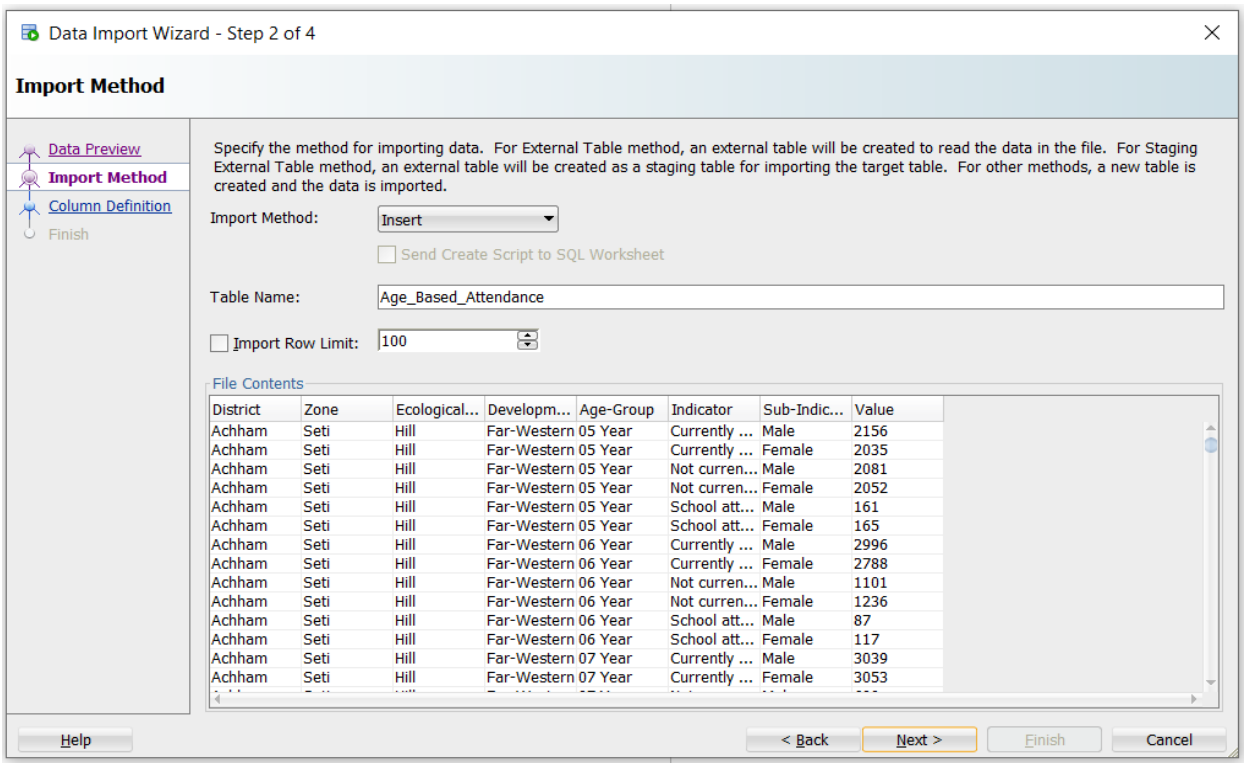
Code Appendix

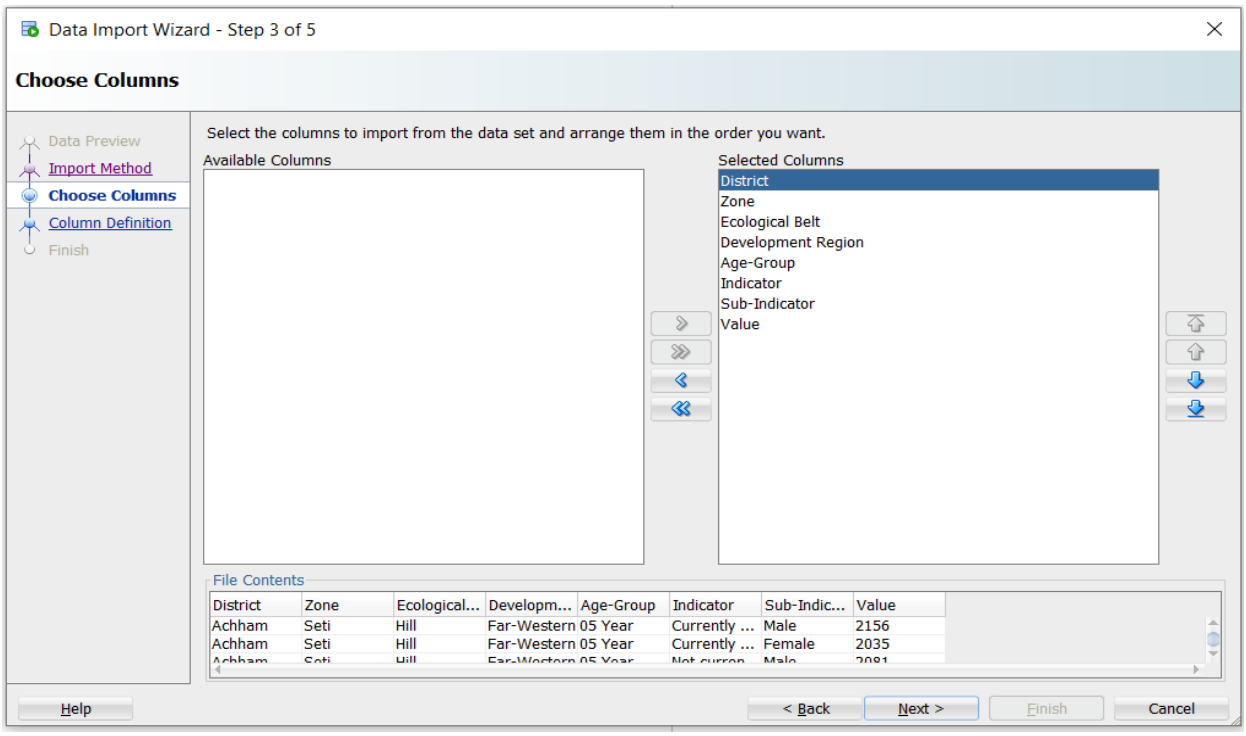
Database Creation and Importing

Importing CSV into Oracle

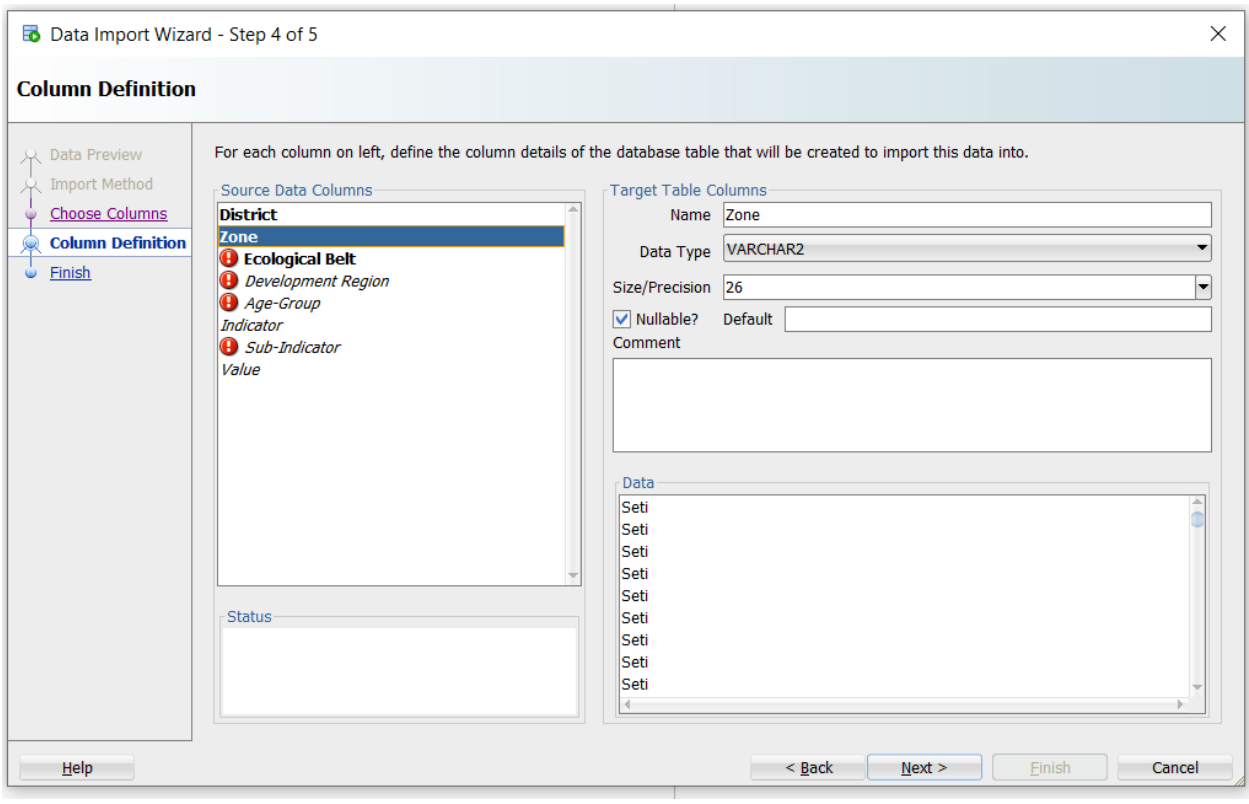
Naming the table Age Based Attendance



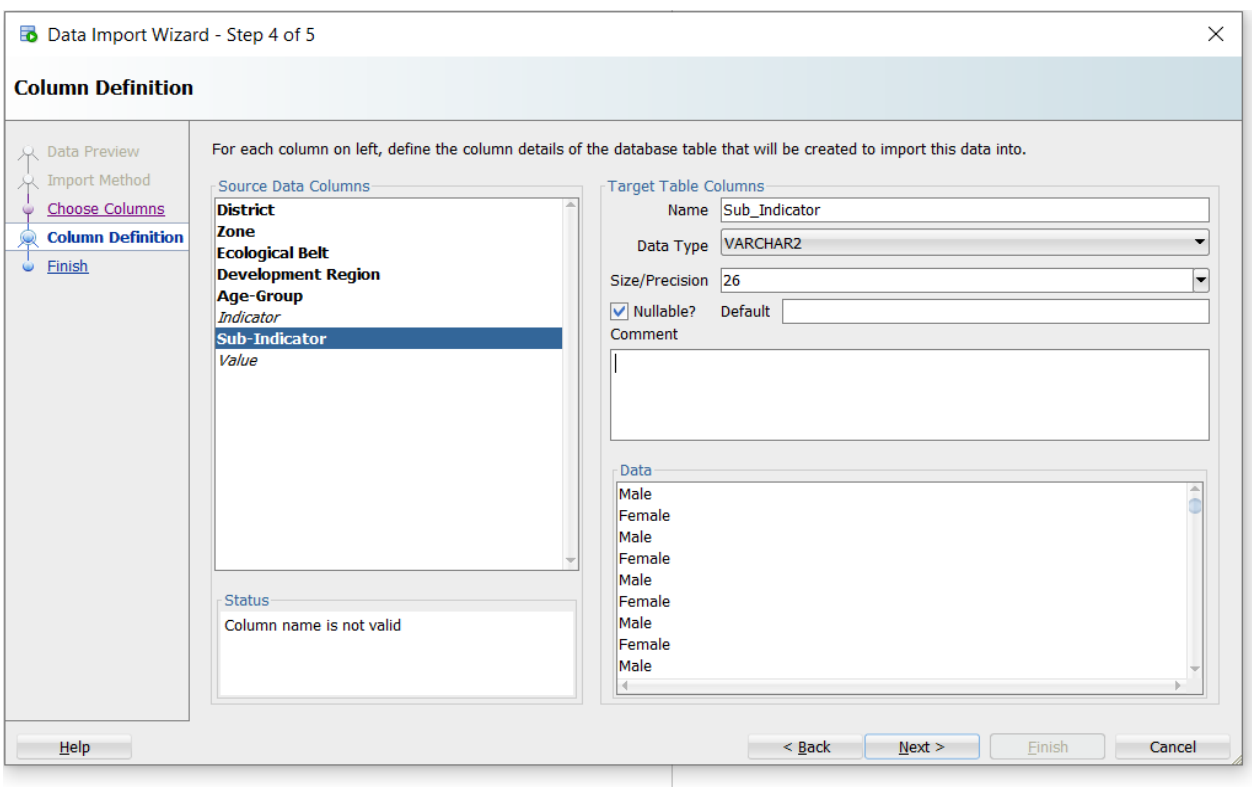


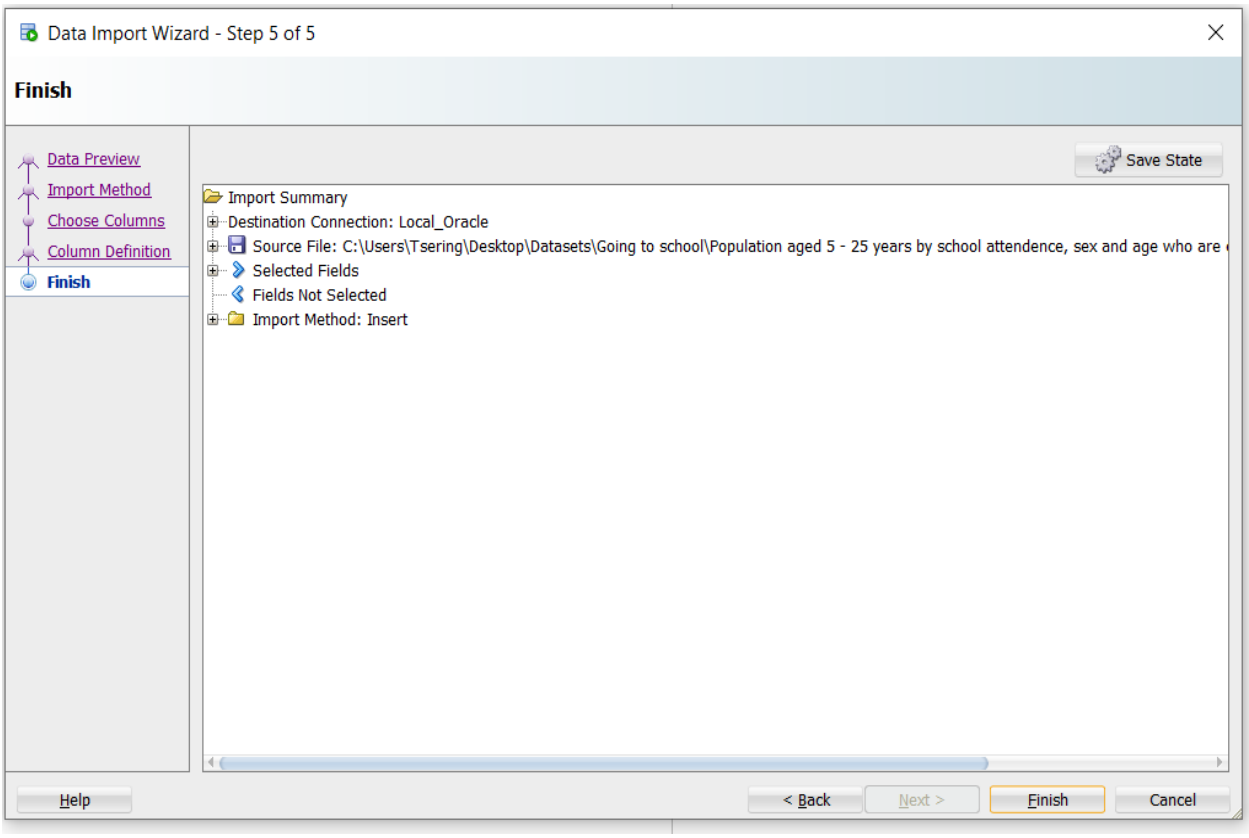


Error occurred while importing tables

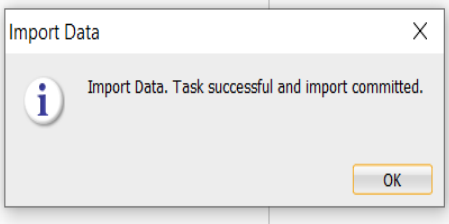


After correcting the error





Data imported successfully



Local_Oracle | AGE_BASED_ATTENDANCE

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DISTRICT	VARCHAR2(26 BYTE)	Yes	(null)	1	(null)
2	ZONE	VARCHAR2(26 BYTE)	Yes	(null)	2	(null)
3	ECOLOGICAL_BELT	VARCHAR2(26 BYTE)	Yes	(null)	3	(null)
4	DEVELOPMENT_REGION	VARCHAR2(26 BYTE)	Yes	(null)	4	(null)
5	AGE_GROUP	NUMBER(38,0)	Yes	(null)	5	(null)
6	INDICATOR	VARCHAR2(128 BYTE)	Yes	(null)	6	(null)
7	SUB_INDICATOR	VARCHAR2(26 BYTE)	Yes	(null)	7	(null)
8	VALUE	NUMBER(38,0)	Yes	(null)	8	(null)

Local_Oracle | AGE_BASED_ATTENDANCE

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. | Filter:

	DISTRICT	ZONE	ECOLOGICAL_BELT	DEVELOPMENT_REGION	AGE_GROUP	INDICATOR	SUB_INDICATOR	VALUE
1	Arghakhanchi	Lumbini	Hill	Western	15	Not currently going to school	Male	188
2	Arghakhanchi	Lumbini	Hill	Western	15	Not currently going to school	Female	276
3	Arghakhanchi	Lumbini	Hill	Western	15	School attendence not Stated	Male	35
4	Arghakhanchi	Lumbini	Hill	Western	15	School attendence not Stated	Female	63
5	Arghakhanchi	Lumbini	Hill	Western	16	Currently going to school	Male	1375
6	Arghakhanchi	Lumbini	Hill	Western	16	Currently going to school	Female	1776
7	Arghakhanchi	Lumbini	Hill	Western	16	Not currently going to school	Male	292
8	Arghakhanchi	Lumbini	Hill	Western	16	Not currently going to school	Female	458
9	Arghakhanchi	Lumbini	Hill	Western	16	School attendence not Stated	Male	37
10	Arghakhanchi	Lumbini	Hill	Western	16	School attendence not Stated	Female	32
11	Arghakhanchi	Lumbini	Hill	Western	17	Currently going to school	Male	921
12	Arghakhanchi	Lumbini	Hill	Western	17	Currently going to school	Female	1219
13	Arghakhanchi	Lumbini	Hill	Western	17	Not currently going to school	Male	353
14	Arghakhanchi	Lumbini	Hill	Western	17	Not currently going to school	Female	686
15	Arghakhanchi	Lumbini	Hill	Western	17	School attendence not Stated	Male	25
16	Arghakhanchi	Lumbini	Hill	Western	17	School attendence not Stated	Female	45
17	Arghakhanchi	Lumbini	Hill	Western	18	Currently going to school	Male	551
18	Arghakhanchi	Lumbini	Hill	Western	18	Currently going to school	Female	832
19	Arghakhanchi	Lumbini	Hill	Western	18	Not currently going to school	Male	457
20	Arghakhanchi	Lumbini	Hill	Western	18	Not currently going to school	Female	1064
21	Arghakhanchi	Lumbini	Hill	Western	18	School attendence not Stated	Male	20
22	Arghakhanchi	Lumbini	Hill	Western	18	School attendence not Stated	Female	57
23	Arghakhanchi	Lumbini	Hill	Western	19	Currently going to school	Male	200
24	Arghakhanchi	Lumbini	Hill	Western	19	Currently going to school	Female	359
25	Arghakhanchi	Lumbini	Hill	Western	19	Not currently going to school	Male	313
26	Arghakhanchi	Lumbini	Hill	Western	19	Not currently going to school	Female	832
27	Arghakhanchi	Lumbini	Hill	Western	19	School attendence not Stated	Male	10
28	Arghakhanchi	Lumbini	Hill	Western	19	School attendence not Stated	Female	35
29	Arghakhanchi	Lumbini	Hill	Western	20	Currently going to school	Male	96
30	Arghakhanchi	Lumbini	Hill	Western	20	Currently going to school	Female	161
31	Arghakhanchi	Lumbini	Hill	Western	20	Not currently going to school	Male	401
32	Arghakhanchi	Lumbini	Hill	Western	20	Not currently going to school	Female	1001
33	Arghakhanchi	Lumbini	Hill	Western	20	School attendence not Stated	Male	26
34	Arghakhanchi	Lumbini	Hill	Western	20	School attendence not Stated	Female	31

Importing JSON Dataset into MongoDB

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Datasets/Going to school$ sudo mongoimport --jsonAr
ray --db SchoolDB --collection SchoolCollection --file schoolnumbers.json
2021-04-24T14:29:51.361+0545    connected to: localhost
2021-04-24T14:29:51.599+0545    imported 1650 documents
```

Importing into Hadoop

Importing CSV

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Desktop/Datasets/Going\ to\ scho
ol/Population\ aged\ 5\ -\ 25\ years\ by\ school\ attendence\,\ sex\ and\ age\ who\ are\ currently\ going\ and\ not\
going\ to\ school.csv .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
'Population aged 5 - 25 years by school attendence, sex and age who are currently going and not going to school.csv'
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Desktop/Datasets/Going\ to\ scho
ol/schoolnumbers.json .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendanceCount.jav
a .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendance.java
cp: missing destination file operand after '/home/2039224_hadoop/Downloads/PopAttendance.java'
Try 'cp --help' for more information.
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendance.java .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
 PopAttendanceCount.java
 PopAttendance.java
'Population aged 5 - 25 years by school attendence, sex and age who are currently going and not going to school.csv'
 schoolnumbers.json
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$
```

Importing json dataset with spark

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ pyspark
Python 3.8.5 (default, Jan 27 2021, 15:41:15)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
21/04/23 01:26:12 WARN Utils: Your hostname, tsering-Inspiron-15-3567 resolves to a loopback address: 127.0.1.1; using 192.168.1.9
4 instead (on interface wlp2s0)
21/04/23 01:26:12 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/home/2039224_hadoop/spark/jars/spark-unsafe_2.12-3.1
.1.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
21/04/23 01:26:12 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes wher
e applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.1.1
      /_/

Using Python version 3.8.5 (default, Jan 27 2021 15:41:15)
Spark context Web UI available at http://192.168.1.94:4040
Spark context available as 'sc' (master = local[*], app id = local-1619120474471).
SparkSession available as 'spark'.
>>> school_numbers = spark.read.json("schoolnumbers.json", multiLine = "True")
```

```
>>> school_numbers = spark.read.json("schoolnumbers.json", multiLine = "True")
>>> school_numbers.show()
+------------------+---------+------------------+------+----------------+-----+
|Development_Region| District|Geographical_Region| Grade|Number_of_School| Zone|
+------------------+---------+------------------+------+----------------+-----+
|           Eastern|Taplejung|          Mountain|     1|               3|Mechi|
|           Eastern|Taplejung|          Mountain| (1-2)|               5|Mechi|
|           Eastern|Taplejung|          Mountain| (1-3)|              58|Mechi|
|           Eastern|Taplejung|          Mountain| (1-4)|              10|Mechi|
|           Eastern|Taplejung|          Mountain| (1-5)|             138|Mechi|
|           Eastern|Taplejung|          Mountain| (1-6)|               2|Mechi|
|           Eastern|Taplejung|          Mountain| (1-7)|              13|Mechi|
|           Eastern|Taplejung|          Mountain| (1-8)|              53|Mechi|
|           Eastern|Taplejung|          Mountain| (1-9)|               2|Mechi|
|           Eastern|Taplejung|          Mountain|(1-10)|              25|Mechi|
|           Eastern|Taplejung|          Mountain|(1-11)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(1-12)|              29|Mechi|
|           Eastern|Taplejung|          Mountain| (6-7)|               0|Mechi|
|           Eastern|Taplejung|          Mountain| (6-8)|               0|Mechi|
|           Eastern|Taplejung|          Mountain| (6-9)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(6-10)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(6-11)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(6-12)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(9-10)|               0|Mechi|
|           Eastern|Taplejung|          Mountain|(9-11)|               0|Mechi|
+------------------+---------+------------------+------+----------------+-----+
only showing top 20 rows

|           Central|Sindhupalchok|          Mountain|(6-12)|               0|Bagmati|
|           Central|Sindhupalchok|          Mountain|(9-10)|               0|Bagmati|
+------------------+-------------+------------------+------+----------------+-------+
only showing top 20 rows

>>>
```

Oracle Query

OLAP query

- **CUBE**

Local_Oracle | AGE_BASED_ATTENDANCE

Worksheet | Query Builder

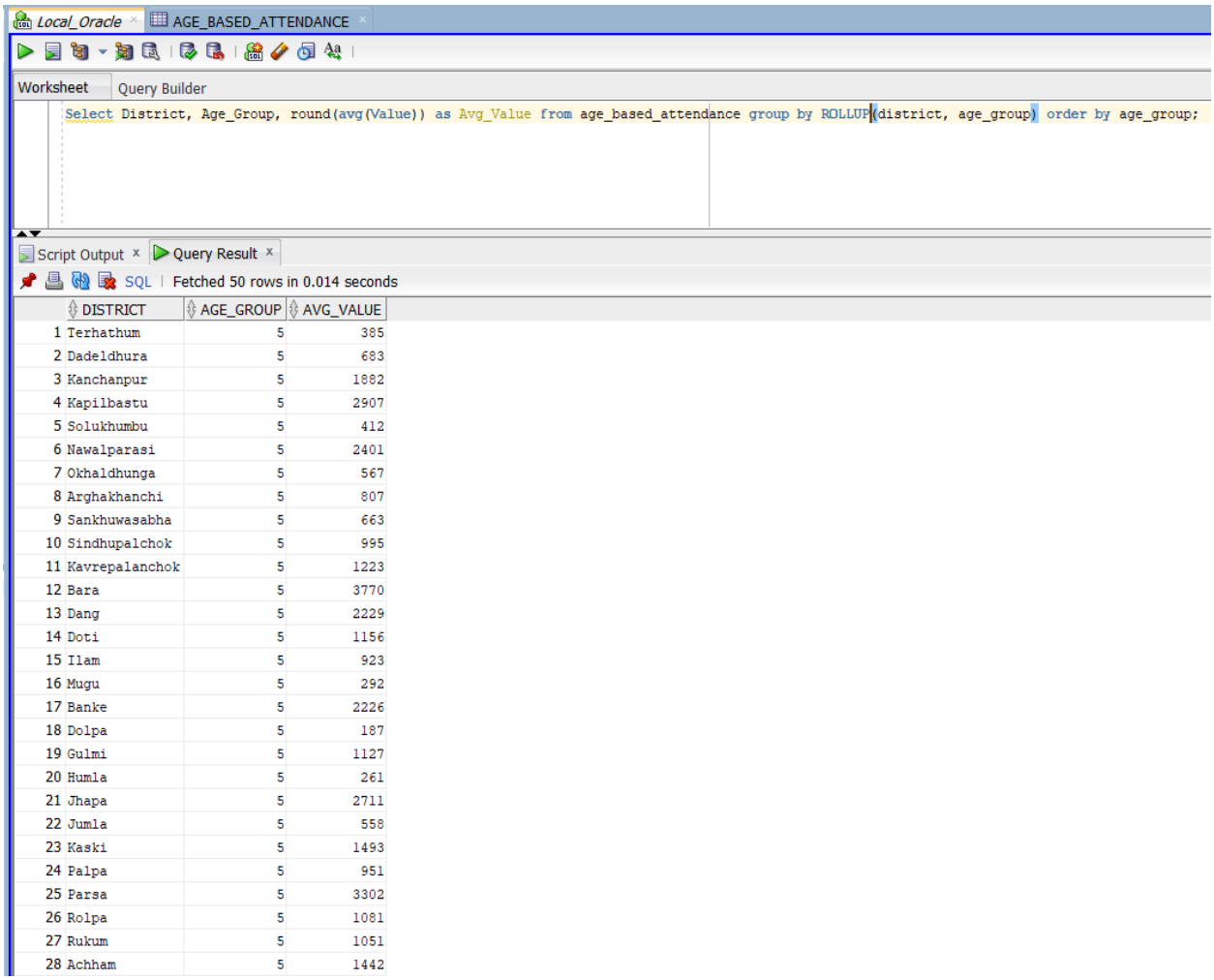
```
Select District, Age_Group, round(avg(Value)) as Avg_Value from age_based_attendance group by CUBE(district, age_group) order by age_group;
```

Query Result

SQL | All Rows Fetched: 1672 in 0.071 seconds

	DISTRICT	AGE_GROUP	AVG_VALUE
1	Achham	5	1442
2	Arghakhanchi	5	807
3	Baglung	5	1094
4	Baitadi	5	1281
5	Bajhang	5	1095
6	Bajura	5	765
7	Banke	5	2226
8	Bara	5	3770
9	Bardiya	5	1631
10	Bhaktapur	5	758
11	Bhojpur	5	704
12	Chitawan	5	1704
13	Dadeldhura	5	683
14	Dailekh	5	1413
15	Dang	5	2229
16	Darchula	5	628
17	Dhading	5	1254
18	Dhankuta	5	560
19	Dhanusa	5	3779
20	Dolakha	5	656
21	Dolpa	5	187
22	Doti	5	1156
23	Gorkha	5	889
24	Gulmi	5	1127
25	Humla	5	261
26	Ilam	5	923
27	Jajarkot	5	970
28	Jhapa	5	2711
29	Jumla	5	558
30	Kailali	5	3246
31	Kalikot	5	775
32	Kanchanpur	5	1882

- **ROLLUP**



	DISTRICT	AGE_GROUP	AVG_VALUE
1	Terhathum	5	385
2	Dadeldhura	5	683
3	Kanchanpur	5	1882
4	Kapilbastu	5	2907
5	Solukhumbu	5	412
6	Nawalparasi	5	2401
7	Okhaldhunga	5	567
8	Arghakhanchi	5	807
9	Sankhuwasabha	5	663
10	Sindhupalchok	5	995
11	Kavrepalanchok	5	1223
12	Bara	5	3770
13	Dang	5	2229
14	Doti	5	1156
15	Ilam	5	923
16	Mugu	5	292
17	Banke	5	2226
18	Dolpa	5	187
19	Gulmi	5	1127
20	Humla	5	261
21	Jhapa	5	2711
22	Jumla	5	558
23	Kaski	5	1493
24	Palpa	5	951
25	Parsa	5	3302
26	Rolpa	5	1081
27	Rukum	5	1051
28	Achham	5	1442

ii) Wildcard query

- **Like “ %”**

Local_Oracle | AGE_BASED_ATTENDANCE

Worksheet | Query Builder

```
Select District, Age_Group, sub_indicator from age_based_attendance where sub_indicator like 'F%';
```

Script Output | Query Result

SQL | Fetched 50 rows in 0.004 seconds

	DISTRICT	AGE_GROUP	SUB_INDICATOR
1	Arghakhanchi	15	Female
2	Arghakhanchi	15	Female
3	Arghakhanchi	16	Female
4	Arghakhanchi	16	Female
5	Arghakhanchi	16	Female
6	Arghakhanchi	17	Female
7	Arghakhanchi	17	Female
8	Arghakhanchi	17	Female
9	Arghakhanchi	18	Female
10	Arghakhanchi	18	Female
11	Arghakhanchi	18	Female
12	Arghakhanchi	19	Female
13	Arghakhanchi	19	Female
14	Arghakhanchi	19	Female
15	Arghakhanchi	20	Female
16	Arghakhanchi	20	Female
17	Arghakhanchi	20	Female
18	Arghakhanchi	21	Female
19	Arghakhanchi	21	Female
20	Arghakhanchi	21	Female
21	Arghakhanchi	22	Female

Development region wise number of students currently going to school

Local_Oracle.sql

SQL Worksheet | History

Worksheet | Query Builder

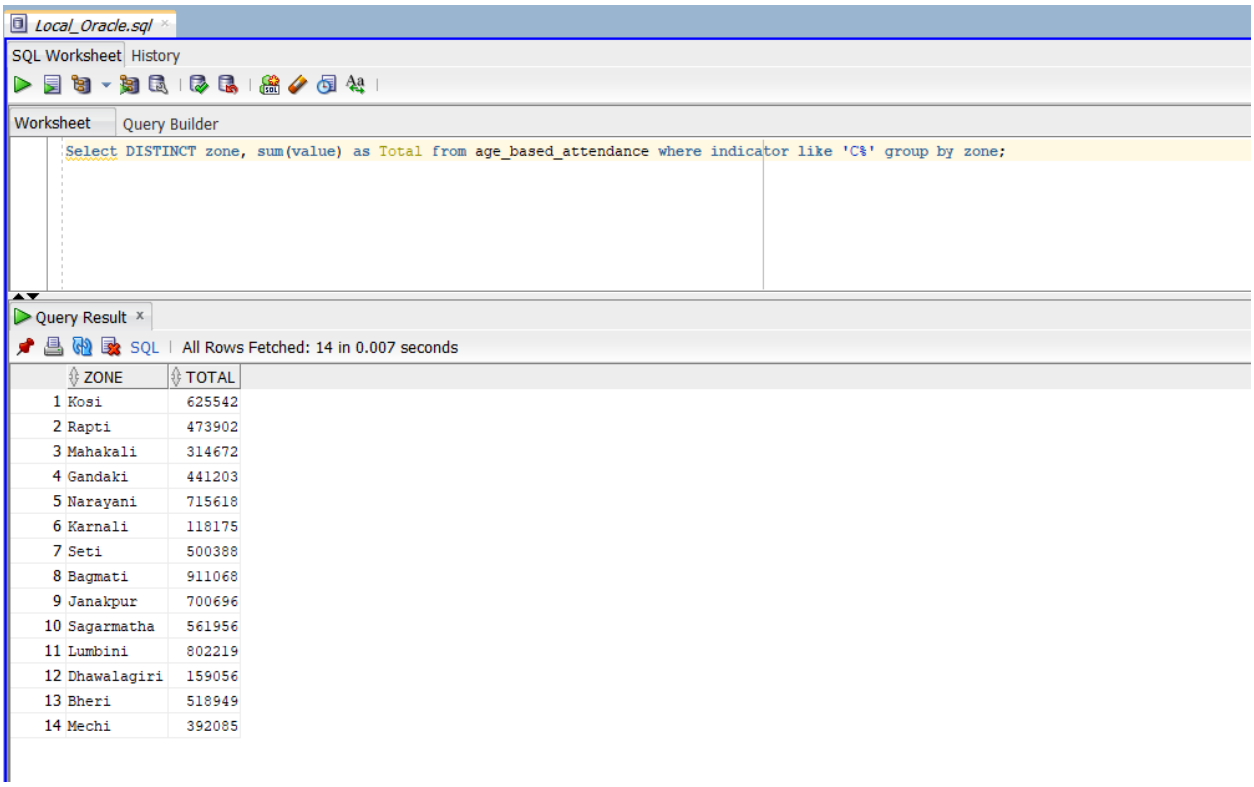
```
Select DISTINCT development_region, sum(value) as Total from age_based_attendance where indicator like 'C%' group by development_region;
```

Script Output | Query Result

SQL | All Rows Fetched: 5 in 0.005 seconds

	DEVELOPMENT_REGION	TOTAL
1	Central	2327382
2	Eastern	1579583
3	Far-Western	815060
4	Western	1402478
5	Mid-Western	1111026

Zone wise number of students currently going to school



	ZONE	TOTAL
1	Kosi	625542
2	Rapti	473902
3	Mahakali	314672
4	Gandaki	441203
5	Narayani	715618
6	Karnali	118175
7	Seti	500388
8	Bagmati	911068
9	Janakpur	700696
10	Sagarmatha	561956
11	Lumbini	802219
12	Dhawalagiri	159056
13	Bheri	518949
14	Mechi	392085

District wise number of students currently going to school

Local_Oracle.sql

SQL Worksheet | History

Worksheet | Query Builder

```
Select DISTINCT district, sum(value) as Total from age_based_attendance where indicator like 'C%' group by district;
```

Script Output | Query Result

SQL | Fetched 50 rows in 0.011 seconds

	DISTRICT	TOTAL
1	Achham	84511
2	Bara	167346
3	Darchula	41668
4	Dhanusa	175320
5	Kapilbastu	144139
6	Mustang	2594
7	Rasuwa	12864
8	Rukum	71445
9	Bhaktapur	67208
10	Gorkha	81934
11	Kailali	242955
12	Kanchanpur	143943
13	Kathmandu	363316
14	Khotang	67206
15	Lalitpur	98518
16	Nuwakot	77337
17	Parsa	141526
18	Ramechhap	64855
19	Siraha	156910
20	Bajhang	61908
21	Banke	129029
22	Chitawan	153179
23	Dadeldhura	48000
24	Dang	172602
25	Doti	65344
26	Gulmi	89906
27	Ilam	81118
28	Kalikot	45929
29	Mahottari	141689
30	Morang	247377
31	Nawalparasi	188008
32	Rautahat	135619
33	Sindhupalchok	80988
34	Tanahu	96696
35	Dolpa	9487

MongoDB Query

- Retrieving all the documents from the collection named SchoolCollection and arranging them in an easy-to-read format using pretty function.

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
> db.SchoolCollection.find().pretty()
{
        "_id" : ObjectId("6083da836fbb08dba1840537"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-4)",
        "Number_of_School" : 10
}
{
        "_id" : ObjectId("6083da836fbb08dba1840538"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-5)",
        "Number_of_School" : 138
}
{
        "_id" : ObjectId("6083da836fbb08dba1840539"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-2)",
        "Number_of_School" : 5
}
{
        "_id" : ObjectId("6083da836fbb08dba184053a"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-6)",
        "Number_of_School" : 2
}
{
        "_id" : ObjectId("6083da836fbb08dba184053b"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-7)",
        "Number_of_School" : 13
}
{
        "_id" : ObjectId("6083da836fbb08dba184053c"),
        "District" : "Taplejung",
        "Zone" : "Mechi",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-8)",
        "Number_of_School" : 53
```

- Finding the documents having district Solukhumbu.

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
> db.SchoolCollection.find({"District":"Solukhumbu"}).pretty()
{
        "_id" : ObjectId("6083da836fbb08dba1840563"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-2)",
        "Number_of_School" : 3
}
{
        "_id" : ObjectId("6083da836fbb08dba1840564"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-3)",
        "Number_of_School" : 53
}
{
        "_id" : ObjectId("6083da836fbb08dba1840565"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-6)",
        "Number_of_School" : 13
}
{
        "_id" : ObjectId("6083da836fbb08dba1840566"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : 1,
        "Number_of_School" : 7
}
{
        "_id" : ObjectId("6083da836fbb08dba1840567"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-5)",
        "Number_of_School" : 105
}
{
        "_id" : ObjectId("6083da836fbb08dba1840568"),
        "District" : "Solukhumbu",
        "Zone" : "Sagarmatha",
        "Geographical_Region" : "Mountain",
        "Development_Region" : "Eastern",
        "Grade" : "(1-8)",
        "Number_of_School" : 33
}
```

- Finding documents whose district is Solukhumbu and grade 1 to 11:

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
> db.SchoolCollection.find({"District":"Solukhumbu"},{"Grade":"(1-11)"}).pretty()
{ "_id" : ObjectId("6083da836fbb08dba1840563"), "Grade" : "(1-2)" }
{ "_id" : ObjectId("6083da836fbb08dba1840564"), "Grade" : "(1-3)" }
{ "_id" : ObjectId("6083da836fbb08dba1840565"), "Grade" : "(1-6)" }
{ "_id" : ObjectId("6083da836fbb08dba1840566"), "Grade" : 1 }
{ "_id" : ObjectId("6083da836fbb08dba1840567"), "Grade" : "(1-5)" }
{ "_id" : ObjectId("6083da836fbb08dba1840568"), "Grade" : "(1-8)" }
{ "_id" : ObjectId("6083da836fbb08dba1840569"), "Grade" : "(1-7)" }
{ "_id" : ObjectId("6083da836fbb08dba184056a"), "Grade" : "(1-9)" }
{ "_id" : ObjectId("6083da836fbb08dba184056b"), "Grade" : "(1-10)" }
{ "_id" : ObjectId("6083da836fbb08dba184056c"), "Grade" : "(1-11)" }
{ "_id" : ObjectId("6083da836fbb08dba184056d"), "Grade" : "(1-12)" }
{ "_id" : ObjectId("6083da836fbb08dba184056e"), "Grade" : "(6-7)" }
{ "_id" : ObjectId("6083da836fbb08dba184056f"), "Grade" : "(6-8)" }
{ "_id" : ObjectId("6083da836fbb08dba1840570"), "Grade" : "(6-9)" }
{ "_id" : ObjectId("6083da836fbb08dba1840571"), "Grade" : "(6-10)" }
{ "_id" : ObjectId("6083da836fbb08dba1840572"), "Grade" : "(6-11)" }
{ "_id" : ObjectId("6083da836fbb08dba1840573"), "Grade" : "(6-12)" }
{ "_id" : ObjectId("6083da836fbb08dba1840574"), "Grade" : "(9-10)" }
{ "_id" : ObjectId("6083da836fbb08dba1840575"), "Grade" : "(9-12)" }
{ "_id" : ObjectId("6083da836fbb08dba1840576"), "Grade" : "(9-11)" }
Type "it" for more
```

- Finding the total number (count) of documents whose zone is Sagarmatha and district which contains regular expression ‘khumbu’

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
> db.SchoolCollection.find({"Zone":"Sagarmatha"},{District:{$regex:/khumbu/}}).count()
132
```

- Aggregate using sum to find out the number of schools based on geographies, zones and districts.

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
> db.SchoolCollection.aggregate([{$group:{_id:"$Zone", TotalSchool: {$sum:"$Number_of_School"}}}]);
{ "_id" : "Mahakali", "TotalSchool" : 1623 }
{ "_id" : "Lumbini", "TotalSchool" : 3377 }
{ "_id" : "Kosi", "TotalSchool" : 2885 }
{ "_id" : "Janakpur", "TotalSchool" : 3065 }
{ "_id" : "Sagarmatha", "TotalSchool" : 2596 }
{ "_id" : "Mechi", "TotalSchool" : 1968 }
{ "_id" : "Rapti", "TotalSchool" : 2215 }
{ "_id" : "Seti", "TotalSchool" : 2372 }
{ "_id" : "Bagmati", "TotalSchool" : 4773 }
{ "_id" : "Narayani", "TotalSchool" : 2446 }
{ "_id" : "Gandaki", "TotalSchool" : 2966 }
{ "_id" : "Dhawalagiri", "TotalSchool" : 1306 }
{ "_id" : "Karnali", "TotalSchool" : 851 }
{ "_id" : "Bheri", "TotalSchool" : 2339 }
> db.SchoolCollection.aggregate([{$group:{_id:"$District", TotalSchool: {$sum:"$Number_of_School"}}}]
);
{ "_id" : "Kanchanpur", "TotalSchool" : 430 }
{ "_id" : "Kailali", "TotalSchool" : 747 }
{ "_id" : "Dadeldhura", "TotalSchool" : 272 }
{ "_id" : "Doti", "TotalSchool" : 399 }
{ "_id" : "Bajura", "TotalSchool" : 267 }
{ "_id" : "Baitadi", "TotalSchool" : 556 }
{ "_id" : "Bardiya", "TotalSchool" : 350 }
{ "_id" : "Banke", "TotalSchool" : 452 }
{ "_id" : "Salyan", "TotalSchool" : 481 }
{ "_id" : "Rautahat", "TotalSchool" : 487 }
{ "_id" : "Sarlahi", "TotalSchool" : 745 }
{ "_id" : "Ilam", "TotalSchool" : 509 }
{ "_id" : "Dang", "TotalSchool" : 521 }
{ "_id" : "Dhanusa", "TotalSchool" : 400 }
{ "_id" : "Kavre", "TotalSchool" : 696 }
{ "_id" : "Makwanpur", "TotalSchool" : 606 }
{ "_id" : "Kapilbastu", "TotalSchool" : 511 }
{ "_id" : "Dolakha", "TotalSchool" : 431 }
{ "_id" : "Siraha", "TotalSchool" : 469 }
{ "_id" : "Chitawan", "TotalSchool" : 543 }
Type "it" for more
```

```
> db.SchoolCollection.aggregate([{$group:{_id:"$Zone", TotalSchool: {$sum:"$Number_of_School"}}}, {$o
ut:"ZoneWiseSchool"}]);
> db.ZoneWiseSchool.find().pretty()
{ "_id" : "Mahakali", "TotalSchool" : 1623 }
{ "_id" : "Lumbini", "TotalSchool" : 3377 }
{ "_id" : "Kosi", "TotalSchool" : 2885 }
{ "_id" : "Janakpur", "TotalSchool" : 3065 }
{ "_id" : "Sagarmatha", "TotalSchool" : 2596 }
{ "_id" : "Mechi", "TotalSchool" : 1968 }
{ "_id" : "Rapti", "TotalSchool" : 2215 }
{ "_id" : "Seti", "TotalSchool" : 2372 }
{ "_id" : "Bagmati", "TotalSchool" : 4773 }
{ "_id" : "Narayani", "TotalSchool" : 2446 }
{ "_id" : "Gandaki", "TotalSchool" : 2966 }
{ "_id" : "Dhawalagiri", "TotalSchool" : 1306 }
{ "_id" : "Karnali", "TotalSchool" : 851 }
{ "_id" : "Bheri", "TotalSchool" : 2339 }
> db.SchoolCollection.aggregate([{$group:{_id:"$District", TotalSchool: {$sum:"$Number_of_School"}}},
 {$out:"DistrictWiseSchool"}]);
> db.Dis
db.DistrictWiseSchool      db.disableFreeMonitoring(
> db.DistrictWiseSchool.find().count()
75
> db.DistrictWiseSchool.findOne()
{ "_id" : "Kanchanpur", "TotalSchool" : 430 }
>
```

- Using findOne() method that returns only one document.

```
> db.DistrictWiseSchool.findOne()
{ "_id" : "Kanchanpur", "TotalSchool" : 430 }
>
```

- Exporting collections into json dataset using mongoexport command

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Datasets/Going to school
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Datasets/Going to school$ mongoexport --collection=Dis
trictWiseSchool --db=SchoolDB --out=exported/district.json
2021-04-24T15:54:41.939+0545    connected to: localhost
2021-04-24T15:54:41.940+0545    exported 75 records
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Datasets/Going to school$ mongoexport --collection=Zon
eWiseSchool --db=SchoolDB --out=exported/zone.json
2021-04-24T15:55:00.121+0545    connected to: localhost
2021-04-24T15:55:00.121+0545    exported 14 records
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Datasets/Going to school$ ls
 district.json
'Education Level wise Net Enrolment Rate (2012-13).csv'
 exported
'Population aged 5 - 25 years by school attendence, sex and age who are currently going and not going to
 school.csv'
 primer-dataset.json
 schoolnumbers.json
'Total number of schools by grade 2012-13.csv'
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Datasets/Going to school$
```

```
2039224_hadoop@tsering-Inspiron-15-3567: ~
2039224_hadoop@tsering-Inspiron-15-3567:~$ mongoexport --collection=GeographyWis
eSchool --db=SchoolDB --out=exported/zone.json
2021-04-24T16:03:18.655+0545    connected to: localhost
2021-04-24T16:03:18.656+0545    exported 3 records
2039224_hadoop@tsering-Inspiron-15-3567:~$
```

Hadoop Query

Using the java class for MapReduce

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Desktop/Datasets/Going\ to\ scho
ol/Population\ aged\ 5\ -\ 25\ years\ by\ school\ attendence\,\ sex\ and\ age\ who\ are\ currently\ going\ and\ not\
going\ to\ school.csv .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
'Population aged 5 - 25 years by school attendence, sex and age who are currently going and not going to school.csv'
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Desktop/Datasets/Going\ to\ scho
ol/schoolnumbers.json .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendanceCount.jav
a .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendance.java
cp: missing destination file operand after '/home/2039224_hadoop/Downloads/PopAttendance.java'
Try 'cp --help' for more information.
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ cp /home/2039224_hadoop/Downloads/PopAttendance.java .
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
 PopAttendanceCount.java
 PopAttendance.java
'Population aged 5 - 25 years by school attendence, sex and age who are currently going and not going to school.csv'
 schoolnumbers.json
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$
```

Running the java file to find district wise student numbers currently going to school

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ javac -classpath $(hadoop classpath) PopAttendanceDistrict.java
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hadoop jar PopulationAttendanceDistrict.jar PopAttendanceDistrict pro
tfolio_input/Population_Attendance.csv portfolio_output
JAR does not exist or is not a normal file: /home/2039224_hadoop/Desktop/Portfolio/PopulationAttendanceDistrict.jar
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ jar cf PopulationAttendanceDistrict.jar PopAttendanceDistrict*.class
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hadoop jar PopulationAttendanceDistrict.jar PopAttendanceDistrict pro
tfolio_input/Population_Attendance.csv portfolio_output
2021-04-23 00:59:01,759 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2021-04-23 00:59:02,055 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool i
nterface and execute your application with ToolRunner to remedy this.
2021-04-23 00:59:02,097 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/2039224_ha
doop/.staging/job_1619117249026_0002
2021-04-23 00:59:02,311 INFO input.FileInputFormat: Total input files to process : 1
2021-04-23 00:59:02,497 INFO mapreduce.JobSubmitter: number of splits:1
2021-04-23 00:59:02,784 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1619117249026_0002
2021-04-23 00:59:02,786 INFO mapreduce.JobSubmitter: Executing with tokens: []
2021-04-23 00:59:02,937 INFO conf.Configuration: resource-types.xml not found
2021-04-23 00:59:02,937 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2021-04-23 00:59:02,986 INFO impl.YarnClientImpl: Submitted application application_1619117249026_0002
2021-04-23 00:59:03,019 INFO mapreduce.Job: The url to track the job: http://tsering-Inspiron-15-3567:8088/proxy/application_16191
17249026_0002/
2021-04-23 00:59:03,020 INFO mapreduce.Job: Running job: job_1619117249026_0002
2021-04-23 00:59:09,161 INFO mapreduce.Job: Job job_1619117249026_0002 running in uber mode : false
2021-04-23 00:59:09,161 INFO mapreduce.Job:  map 0% reduce 0%
2021-04-23 00:59:14,227 INFO mapreduce.Job:  map 100% reduce 0%
2021-04-23 00:59:19,268 INFO mapreduce.Job:  map 100% reduce 100%
2021-04-23 00:59:19,284 INFO mapreduce.Job: Job job_1619117249026_0002 completed successfully
2021-04-23 00:59:19,384 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=116178
                FILE: Number of bytes written=701317
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=685962
                HDFS: Number of bytes written=1072
                HDFS: Number of read operations=8
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=2484
                Total time spent by all reduces in occupied slots (ms)=2483
                Total time spent by all map tasks (ms)=2484
                Total time spent by all reduce tasks (ms)=2483
                Total vcore-milliseconds taken by all map tasks=2484
                Total vcore-milliseconds taken by all reduce tasks=2483
                Total megabyte-milliseconds taken by all map tasks=2543616
                Total megabyte-milliseconds taken by all reduce tasks=2542592
        Map-Reduce Framework
                Map input records=9450
                Map output records=9450
                Map output bytes=97272
                Map output materialized bytes=116178
                Input split bytes=148
```

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ javac -classpath $(hadoop classpath) PopAttend
ance.java
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ jar cf PopulationAttendance.jar PopAttendance*
.class
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
'PopAttendance$csvReducer.class'   PopAttendanceCount.java     PopulationAttendance.jar
'PopAttendance$PopMapper.class'    PopAttendance.java           schoolnumbers.json
 PopAttendance.class               Population_Attendance.csv
```

Output result of district wise student numbers currently going to school

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -cat portfolio_output/part-r-00000
Achham,126,0
Arghakhanchi,126,0
Baglung,126,0
Baitadi,126,0
Bajhang,126,0
Bajura,126,0
Banke,126,0
Bara,126,0
Bardiya,126,0
Bhaktapur,126,0
Bhojpur,126,0
Chitawan,126,0
Dadeldhura,126,0
Dailekh,126,0
Dang,126,0
Darchula,126,0
Dhading,126,0
Dhankuta,126,0
Dhanusa,126,0
Dolakha,126,0
Dolpa,126,0
Doti,126,0
Gorkha,126,0
Gulmi,126,0
Humla,126,0
Ilam,126,0
Jajarkot,126,0
Jhapa,126,0
Jumla,126,0
Kailali,126,0
Kalikot,126,0
Kanchanpur,126,0
Kapilbastu,126,0
Kaski,126,0
Kathmandu,126,0
Kavrepalanchok,126,0
Khotang,126,0
Lalitpur,126,0
Lamjung,126,0
Mahottari,126,0
Makwanpur,126,0
Manang,126,0
Morang,126,0
Mugu,126,0
Mustang,126,0
Myagdi,126,0
Nawalparasi,126,0
Nuwakot,126,0
Okhaldhunga,126,0
Palpa,126,0
Panchthar,126,0
```

Running the java file to find age wise student numbers currently going to school

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -get portfolio_output/part-r-00000 districtWiseSchoolNo.csv
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ javac -classpath $(hadoop classpath) PopAttendanceAge.java
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ jar cf PopulationAttendanceAge.jar PopAttendanceAge*.class
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hadoop jar PopulationAttendanceAge.jar PopAttendanceAge protfolio_input/Population_Attendance.csv portfolio_output
2021-04-23 01:02:12,021 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2021-04-23 01:02:12,264 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2021-04-23 01:02:12,296 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/2039224_hadoop/.staging/job_1619117249026_0003
2021-04-23 01:02:12,520 INFO input.FileInputFormat: Total input files to process : 1
2021-04-23 01:02:12,673 INFO mapreduce.JobSubmitter: number of splits:1
2021-04-23 01:02:12,972 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1619117249026_0003
2021-04-23 01:02:12,973 INFO mapreduce.JobSubmitter: Executing with tokens: []
2021-04-23 01:02:13,128 INFO conf.Configuration: resource-types.xml not found
2021-04-23 01:02:13,128 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2021-04-23 01:02:13,186 INFO impl.YarnClientImpl: Submitted application application_1619117249026_0003
2021-04-23 01:02:13,222 INFO mapreduce.Job: The url to track the job: http://tsering-Inspiron-15-3567:8088/proxy/application_1619117249026_0003/
2021-04-23 01:02:13,223 INFO mapreduce.Job: Running job: job_1619117249026_0003
2021-04-23 01:02:20,396 INFO mapreduce.Job: Job job_1619117249026_0003 running in uber mode : false
2021-04-23 01:02:20,398 INFO mapreduce.Job:  map 0% reduce 0%
2021-04-23 01:02:25,502 INFO mapreduce.Job:  map 100% reduce 0%
2021-04-23 01:02:30,556 INFO mapreduce.Job:  map 100% reduce 100%
2021-04-23 01:02:30,565 INFO mapreduce.Job: Job job_1619117249026_0003 completed successfully
2021-04-23 01:02:30,657 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=63906
                FILE: Number of bytes written=596753
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=685962
                HDFS: Number of bytes written=184
                HDFS: Number of read operations=8
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=2938
                Total time spent by all reduces in occupied slots (ms)=2459
                Total time spent by all map tasks (ms)=2938
                Total time spent by all reduce tasks (ms)=2459
                Total vcore-milliseconds taken by all map tasks=2938
                Total vcore-milliseconds taken by all reduce tasks=2459
                Total megabyte-milliseconds taken by all map tasks=3008512
                Total megabyte-milliseconds taken by all reduce tasks=2518016
        Map-Reduce Framework
                Map input records=9450
                Map output records=9450
                Map output bytes=45000
                Map output materialized bytes=63906
                Input split bytes=148
                Combine input records=0
                Combine output records=0
                Reduce input groups=21
```

Output of number of student currently going to school based on age

```
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -cat portfolio_output/part-r-00000
10,450,0
11,450,0
12,450,0
13,450,0
14,450,0
15,450,0
16,450,0
17,450,0
18,450,0
19,450,0
20,450,0
21,450,0
22,450,0
23,450,0
24,450,0
25,450,0
5,450,0
6,450,0
7,450,0
8,450,0
9,450,0
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -get portfolio_output/part-r-00000 ageWiseSchoolNo.csv
```

Running the java file to find zone wise number of studnent currently going to school

```
2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -put Population_Attendance.csv
protfolio_input
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ javac -classpath $(hadoop classpath) Pop
Attendance.java
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ jar cf PopulationAttendance.jar PopAtten
dance*.class
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ ls
 District_attendance.csv           PopAttendance.class       Population_Attendance.csv
'PopAttendance$csvReducer.class'   PopAttendanceCount.java   PopulationAttendance.jar
'PopAttendance$PopMapper.class'    PopAttendance.java        schoolnumbers.json
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hadoop jar PopulationAttendance.jar PopA
ttendance protfolio_input/Population_Attendance.csv portfolio_output
2021-04-23 00:39:37,358 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2021-04-23 00:39:37,688 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not pe
rformed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2021-04-23 00:39:37,740 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/h
adoop-yarn/staging/2039224_hadoop/.staging/job_1619117249026_0001
2021-04-23 00:39:38,014 INFO input.FileInputFormat: Total input files to process : 1
2021-04-23 00:39:39,062 INFO mapreduce.JobSubmitter: number of splits:1
2021-04-23 00:39:39,762 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1619117249026_000
1
2021-04-23 00:39:39,766 INFO mapreduce.JobSubmitter: Executing with tokens: []
2021-04-23 00:39:39,982 INFO conf.Configuration: resource-types.xml not found
2021-04-23 00:39:39,982 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2021-04-23 00:39:40,429 INFO impl.YarnClientImpl: Submitted application application_1619117249026_000
1
2021-04-23 00:39:40,472 INFO mapreduce.Job: The url to track the job: http://tsering-Inspiron-15-3567
:8088/proxy/application_1619117249026_0001/
2021-04-23 00:39:40,472 INFO mapreduce.Job: Running job: job_1619117249026_0001
2021-04-23 00:39:48,660 INFO mapreduce.Job: Job job_1619117249026_0001 running in uber mode : false
2021-04-23 00:39:48,663 INFO mapreduce.Job:  map 0% reduce 0%
2021-04-23 00:39:52,740 INFO mapreduce.Job:  map 100% reduce 0%
2021-04-23 00:39:57,777 INFO mapreduce.Job:  map 100% reduce 100%
2021-04-23 00:39:58,809 INFO mapreduce.Job: Job job_1619117249026_0001 completed successfully
2021-04-23 00:39:58,979 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=111894
                FILE: Number of bytes written=692717
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=685962
                HDFS: Number of bytes written=195
                HDFS: Number of read operations=8
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=2454
                Total time spent by all reduces in occupied slots (ms)=2616
                Total time spent by all map tasks (ms)=2454
                Total time spent by all reduce tasks (ms)=2616
```

Output of zone wise number of student currently going to school

```
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$ hdfs dfs -cat portfolio_output/part-r-00
000
Bagmati,1008,0
Bheri,630,0
Dhawalagiri,504,0
Gandaki,756,0
Janakpur,756,0
Karnali,630,0
Kosi,756,0
Lumbini,756,0
Mahakali,504,0
Mechi,504,0
Narayani,630,0
Rapti,630,0
Sagarmatha,756,0
Seti,630,0
2039224_hadoop@tsering-Inspiron-15-3567:~/Desktop/Portfolio$
```

Spark Query

Printing schema and creating table

```
>>> school_numbers.printSchema()
root
 |-- Development_Region: string (nullable = true)
 |-- District: string (nullable = true)
 |-- Geographical_Region: string (nullable = true)
 |-- Grade: string (nullable = true)
 |-- Number_of_School: long (nullable = true)
 |-- Zone: string (nullable = true)

>>> school_numbers.createOrReplaceTempView("School_Number")
>>> sql = spark.sql("SELECT * FROM School_Number ORDER BY Zone ASC")
>>> sql.show()
+------------------+-------------+-------------------+------+----------------+-------+
|Development_Region|     District|Geographical_Region| Grade|Number_of_School|   Zone|
+------------------+-------------+-------------------+------+----------------+-------+
|           Central|Sindhupalchok|           Mountain|     1|               5|Bagmati|
|           Central|Sindhupalchok|           Mountain|(9-12)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-2)|              11|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-3)|              80|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-4)|              39|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-5)|             217|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-6)|               4|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-7)|              20|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-8)|              72|Bagmati|
|           Central|Sindhupalchok|           Mountain| (1-9)|               4|Bagmati|
|           Central|Sindhupalchok|           Mountain|(1-10)|              71|Bagmati|
|           Central|Sindhupalchok|           Mountain|(1-11)|               7|Bagmati|
|           Central|Sindhupalchok|           Mountain|(1-12)|              46|Bagmati|
|           Central|Sindhupalchok|           Mountain| (6-7)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain| (6-8)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain| (6-9)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain|(6-10)|               1|Bagmati|
|           Central|Sindhupalchok|           Mountain|(6-11)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain|(6-12)|               0|Bagmati|
|           Central|Sindhupalchok|           Mountain|(9-10)|               0|Bagmati|
+------------------+-------------+-------------------+------+----------------+-------+
only showing top 20 rows

>>>
```

Total number of schools according to zone, district, grade and development region

2039224_hadoop@tsering-Inspiron-15-3567: ~/Desktop/Portfolio

```
>>> sql = spark.sql("SELECT Zone, SUM(Number_of_School) AS Total_School FROM School_Number GROUP BY Zone").show()
+-----------+------------+
|       Zone|Total_School|
+-----------+------------+
|   Narayani|        2446|
|       Seti|        2372|
|      Rapti|        2215|
|       Kosi|        2885|
|    Lumbini|        3377|
|      Mechi|        1968|
|    Bagmati|        4773|
|      Bheri|        2339|
| Sagarmatha|        2596|
|    Karnali|         851|
|   Mahakali|        1623|
|   Janakpur|        3065|
|    Gandaki|        2966|
|Dhawalagiri|        1306|
+-----------+------------+

>>> sql = spark.sql("SELECT District, SUM(Number_of_School) AS Total_School FROM School_Number GROUP BY District").show()
+-----------+------------+
|   District|Total_School|
+-----------+------------+
|   Udayapur|         501|
|       Ilam|         509|
|  Mahottari|         415|
|    Bajhang|         457|
|     Morang|         707|
|    Lamjung|         424|
|      Jumla|         155|
|    Sarlahi|         745|
|     Siraha|         469|
|      Palpa|         496|
| Kapilbastu|         511|
|     Bajura|         267|
|     Tanahu|         654|
|   Sindhuli|         581|
|      Humla|         135|
|     Parbat|         369|
|      Rukum|         399|
|Okhaldhunga|         368|
|    Baglung|         594|
|  Rupandehi|         597|
+-----------+------------+
only showing top 20 rows
```

```
>>> sql = spark.sql("SELECT Grade, SUM(Number_of_School) AS Total_School FROM School_Number GROUP BY Grade").show()
+-------+------------+
|  Grade|Total_School|
+-------+------------+
|  (6-9)|           2|
|  (1-6)|         565|
|  (1-5)|       12234|
| (6-12)|         112|
| (1-10)|        4663|
| (9-10)|           2|
| (1-11)|         210|
|  (6-8)|           8|
|  (1-7)|        1165|
|  (6-7)|           7|
|  (1-2)|        1017|
|  (1-8)|        4289|
|(11-12)|         295|
|      1|         388|
| (9-11)|           0|
| (6-11)|           3|
|  (1-3)|        4864|
|  (1-9)|         394|
| (6-10)|          54|
| (1-12)|        2975|
+-------+------------+
only showing top 20 rows

>>> sql = spark.sql("SELECT Development_Region, SUM(Number_of_School) AS Total_School FROM School_Number GROUP BY Development_Regi
on").show()
+------------------+------------+
|Development_Region|Total_School|
+------------------+------------+
|           Eastern|        7449|
|       Mid-Western|        5405|
|       Far-Western|        3995|
|           Central|       10284|
|           Western|        7649|
+------------------+------------+

>>>
```