NITTE EDUCATION TRUST | NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

ACADEMIC YEAR 2020-2021

KNOWLEDGE • CHARACTER • UNITY

BIGDATA LABORATORY

Report on,

Learning Activity II-Programming Assignment

Submitted by,

Anjani Saurav (1NT18IS030)

Submitted to,

Disha DN,

Assistant Professor,
Department of Information Science and Engineering
Nitte Meenakshi Institute of Technology
Bangalore-064

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(An autonomous institution with A+ Grade by NAAC /UGC, Affiliated to Visvesvaraya Technological University, Belgaum, Approved by UGC/AICTE/Govt. of Karnataka)

Yelahanka, Bengaluru-560064

Big data lab- Learning activity II

Table of Contents

Big data lab- Learning activity II

List of Figures

Big data lab- Learning activity II

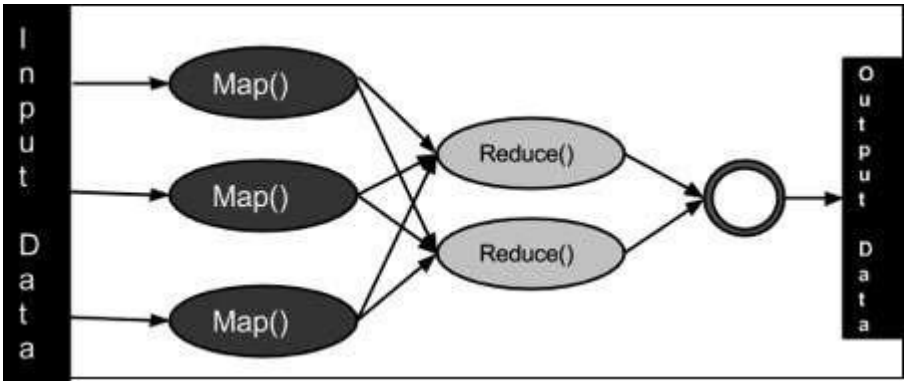
Hadoop and Map Reduce

Hadoop is an open-source framework that allow to store and process big data in a distributed environment across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

MapReduce is a processing technique and a program model for distributed computing based on java.

The MapReduce algorithm contains two important:

1. Map : It takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs).
2. Reduce : It takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job.



MapReduce job −

(Input) <k1, v1> → map → <k2, v2> → reduce → <k3, v3> **(Output).**

The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes.

Big data lab- Learning activity II

Problem Statement

Using the Hadoop Map Reduce framework compute the following :-

1. Total number of employees who are eligible for the pay raise.

2. Total number of cumulative awards the company had this year.

3. How many total awards were obtained by the employee whose salary is 30000 ?

4. Count the number of employees who had paid the Tax.

Big data lab- Learning activity II

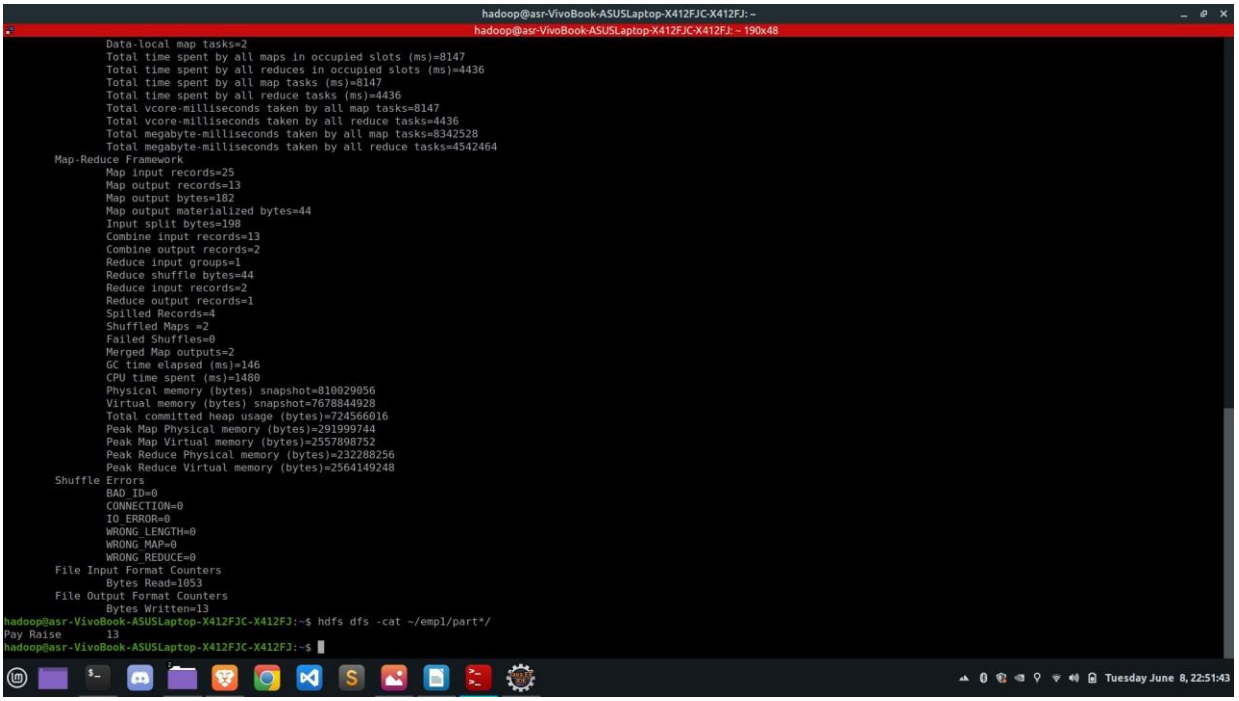
Dataset description

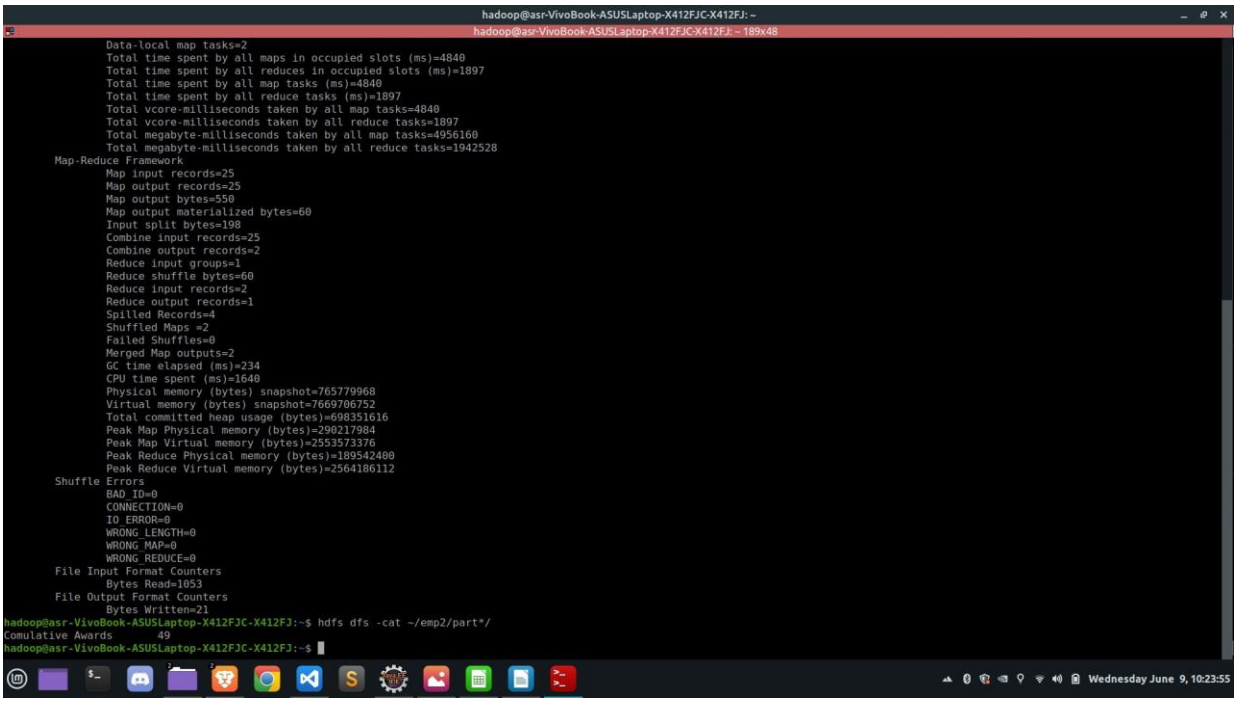
Employee Dataset:

name	ssn	salary	awards	tax_paid	eligible_for_pay_raise
ajay	61301	30000	2	yes	yes
aman	61302	40000	1	yes	no
alok	61303	45000	2	yes	yes
bhavya	61304	55000	4	no	no
divya	61305	30000	3	yes	yes
abhishek	61306	43000	2	no	no
subham	61307	25000	1	yes	no
saini	61308	69000	3	yes	yes
sanket	61309	30000	2	no	yes
supriya	61310	35000	1	yes	no
mishra	61311	42000	3	yes	yes
kaushal	61312	37000	2	no	yes
mallya	61313	28000	2	yes	no
sujan	61314	30000	2	yes	yes
mukund	61315	29000	1	no	no
kshitija	61316	54000	0	yes	no
pandey	61317	30000	0	no	no
rajan	61318	55000	1	yes	no
skrikant	61319	26000	1	yes	no
talpade	61320	30000	1	no	no
raji	61321	30000	2	no	yes
saumya	61322	40000	2	no	yes
pallavi	61323	48000	3	yes	yes
shreya	61324	39000	4	no	yes
vaibhav	61325	60000	4	yes	yes

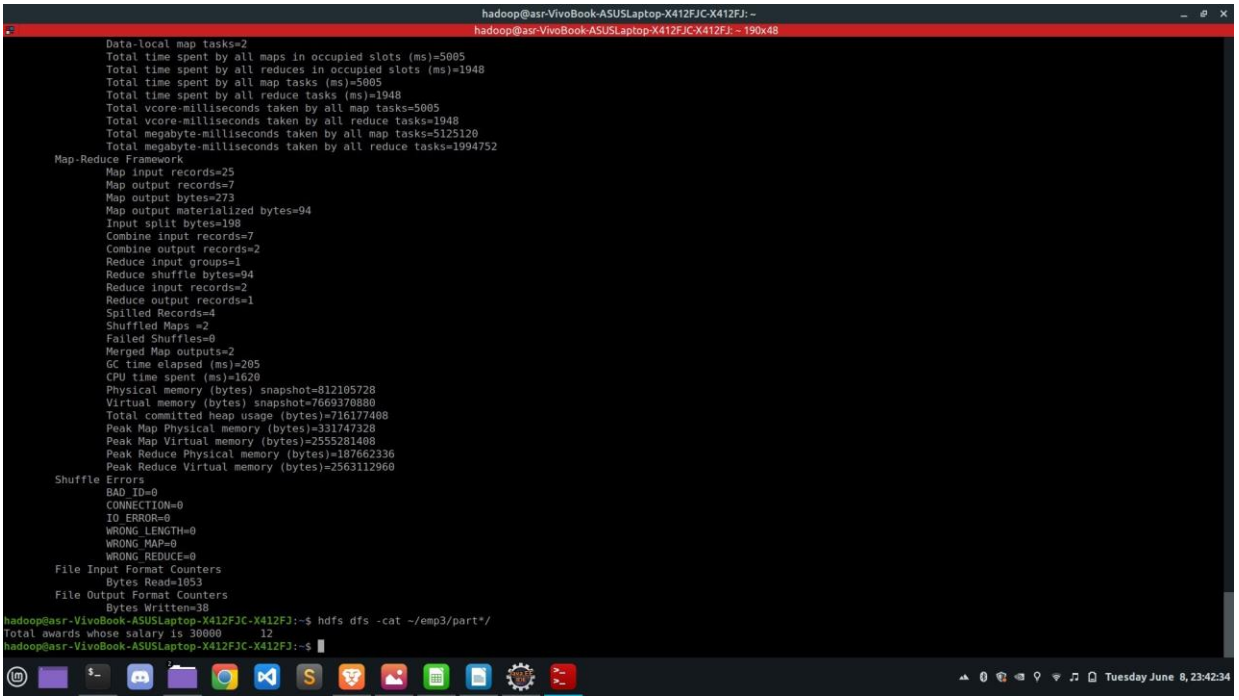
Big data lab- Learning activity II

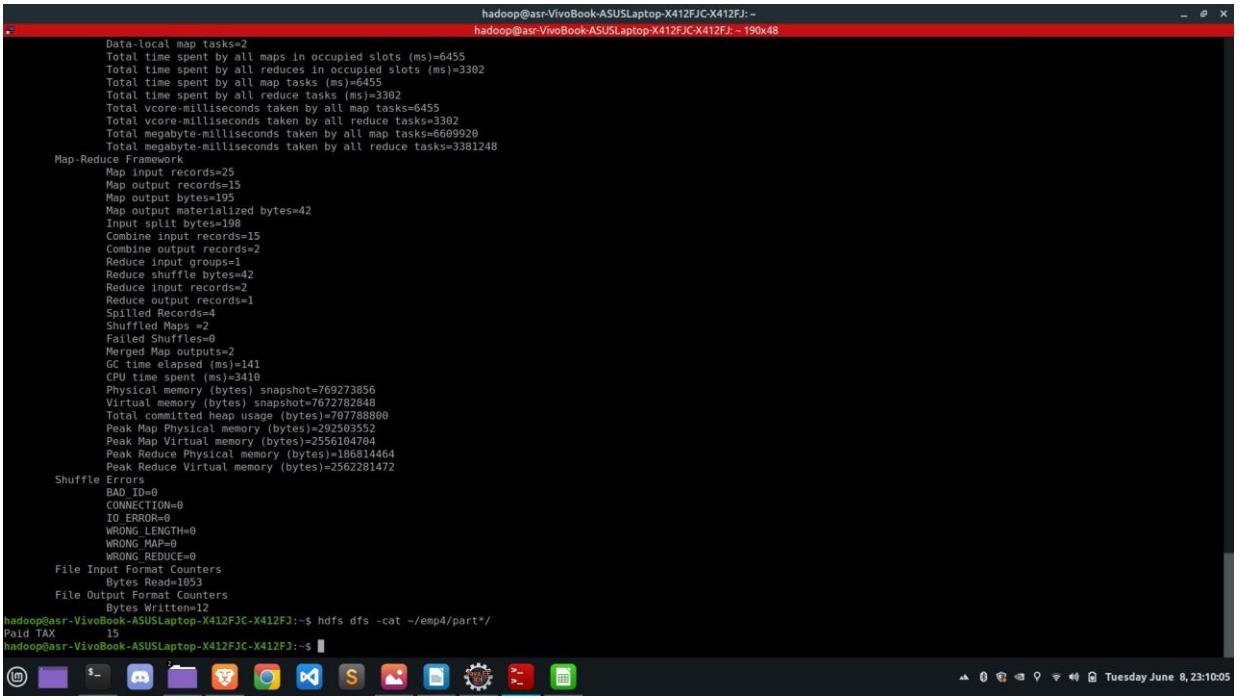
Results and Snapshots





Big data lab- Learning activity II





Big data lab- Learning activity II

Hive

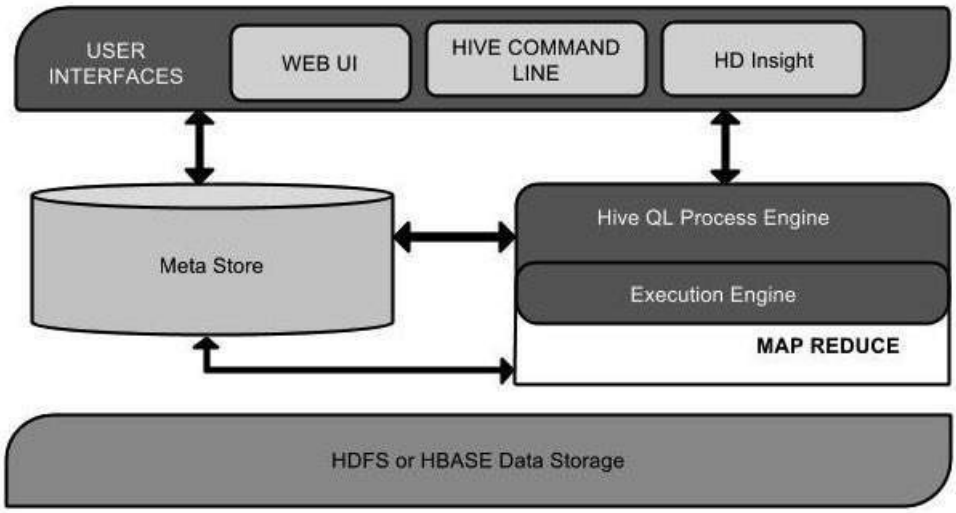
Hive is a data warehouse infrastructure tool to process structured data in Hadoop. It resides on top of Hadoop to summarize Big Data, and makes querying and analyzing easy.

Initially Hive was developed by Facebook, later the Apache Software Foundation took it up and developed it further as an open source under the name Apache Hive. It is used by different companies. For example, Amazon uses it in Amazon Elastic MapReduce.

Features of Hive :-

- It stores schema in a database and processed data into HDFS.
- It is designed for OLAP.
- It provides SQL type language for querying called HiveQL or HQL.
- It is familiar, fast, scalable and extensible.

Architecture of Hive :-



Big data lab- Learning activity II

Dataset description

Employee Dataset :

name	ssn	salary	awards	tax_paid	eligible_for_pay_raise
ajay	61301	30000	2	yes	yes
aman	61302	40000	1	yes	no
alok	61303	45000	2	yes	yes
bhavya	61304	55000	4	no	no
divya	61305	30000	3	yes	yes
abhishek	61306	43000	2	no	no
subham	61307	25000	1	yes	no
saini	61308	69000	3	yes	yes
sanket	61309	30000	2	no	yes
supriya	61310	35000	1	yes	no
mishra	61311	42000	3	yes	yes
kaushal	61312	37000	2	no	yes
mallya	61313	28000	2	yes	no
sujan	61314	30000	2	yes	yes
mukund	61315	29000	1	no	no
kshitija	61316	54000	0	yes	no
pandey	61317	30000	0	no	no
rajan	61318	55000	1	yes	no
skrikant	61319	26000	1	yes	no
talpade	61320	30000	1	no	no
raji	61321	30000	2	no	yes
saumya	61322	40000	2	no	yes
pallavi	61323	48000	3	yes	yes
shreya	61324	39000	4	no	yes
vaibhav	61325	60000	4	yes	yes

Orders Dataset :

order_id	custssn	amount
61	61302	50000
62	61304	30000
63	61304	10000
64	61302	10000
65	61308	8000
66	61309	50000
67	61309	15000
68	61310	16000
69	61306	17000
70	61301	18000
71	61315	12000
72	61316	30000
73	61317	31000
74	61318	35000
75	61319	2100
76	61320	1000
77	61321	16000
78	61322	20000
79	61323	21000
80	61324	12000
81	61306	12000
82	61307	1100
83	61308	501
84	61309	190
85	61310	11111
86	61311	9999
87	61312	9991
88	61313	2390
89	61314	7891
90	61315	65000

Big data lab- Learning activity II

Problem Statement

Using HiveQL language perform following Queries :-

1 Insert 5 records using the INSERT command.

2 Demonstrate the Alter command for the following cases,

2.a Rename the table name to “Emp”.

2.b Rename the column name “Eligible for Pay raise” to “Eligibility”.

3 Count the number of Employees who are eligible for pay raise who had paid the tax.

4 Extract all the users ordered by the Name who had paid the tax but are not eligible for pay raise.

5 Create a separate view containing “SSN and Salary” and call the view as sal_ssn_view.

6 Display count (eligibility) fields grouped by the SSN.

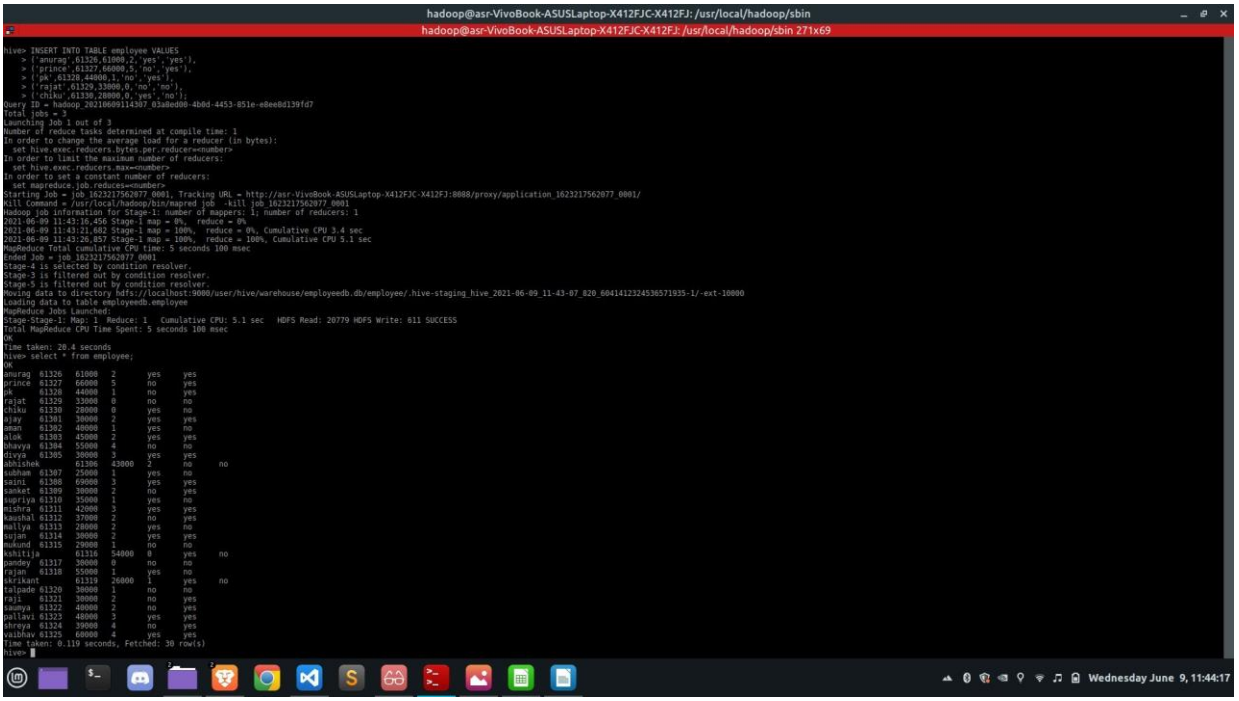
7 Display the (Name, SSN) of employees whose salary is >40000 but < 48000.

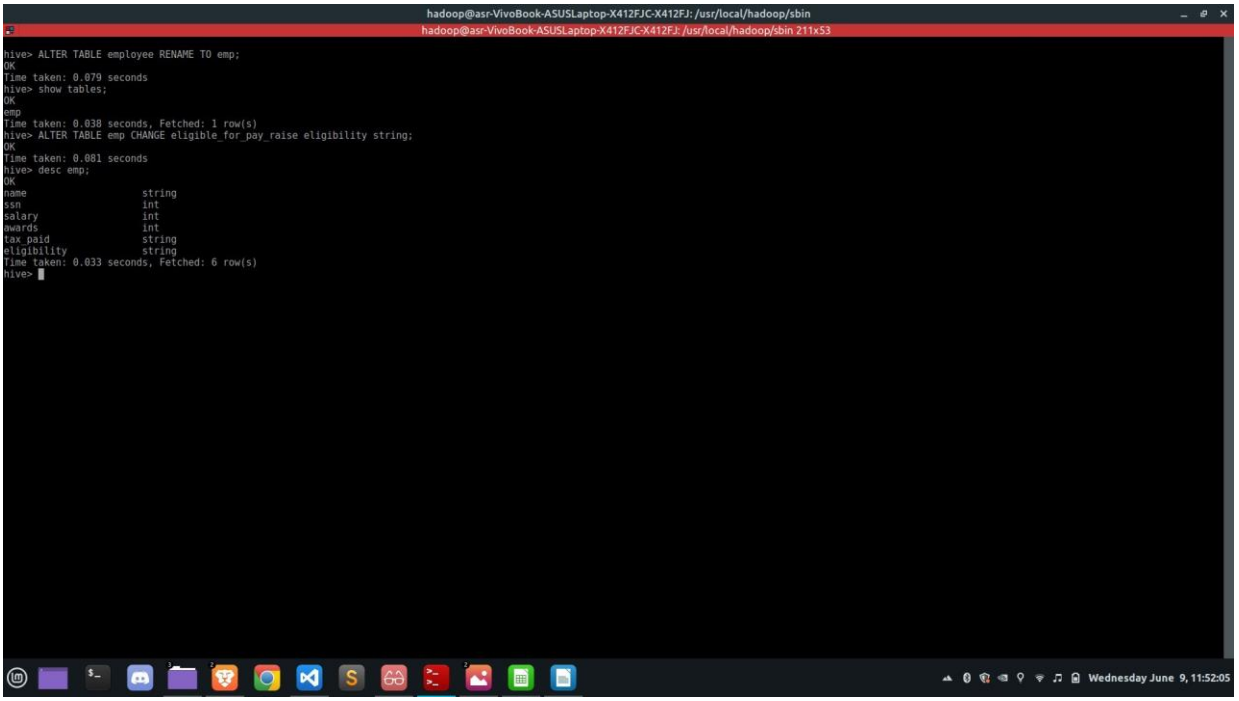
8 Create Another table called orders with the following fields (custssn = SSN in the Employee) and perform the following joins over custssn .

1) Outer 2.Left outer 3.Right outer.

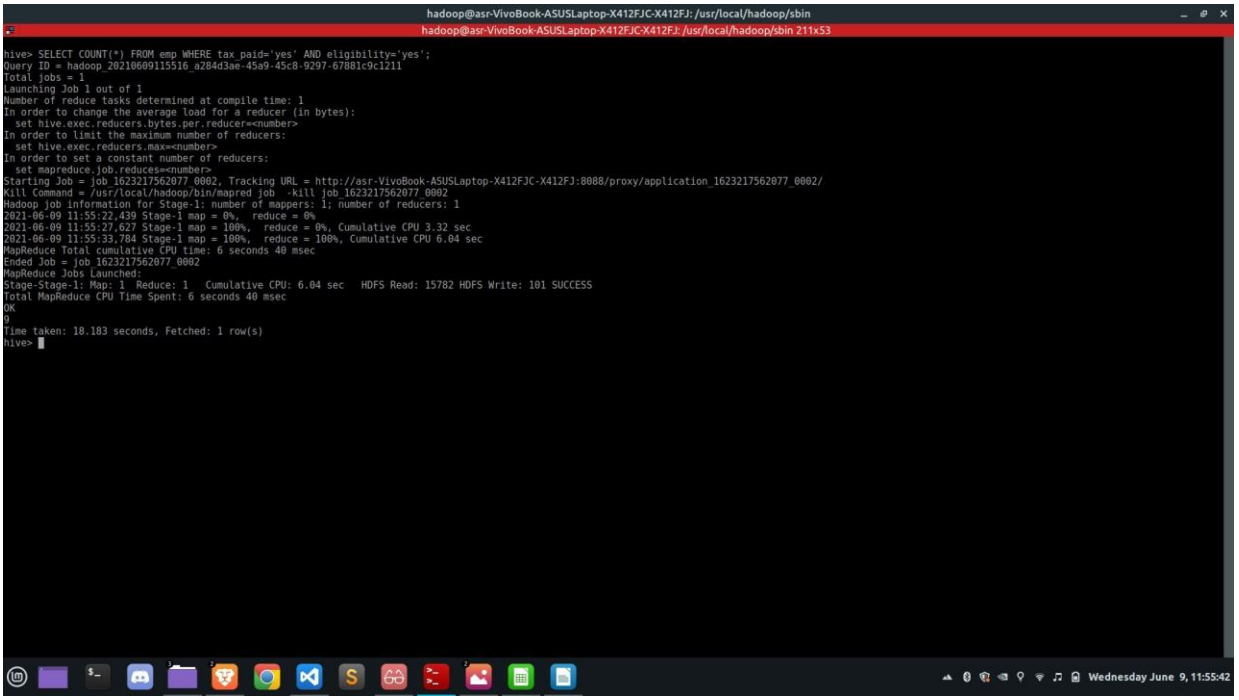
Big data lab- Learning activity II

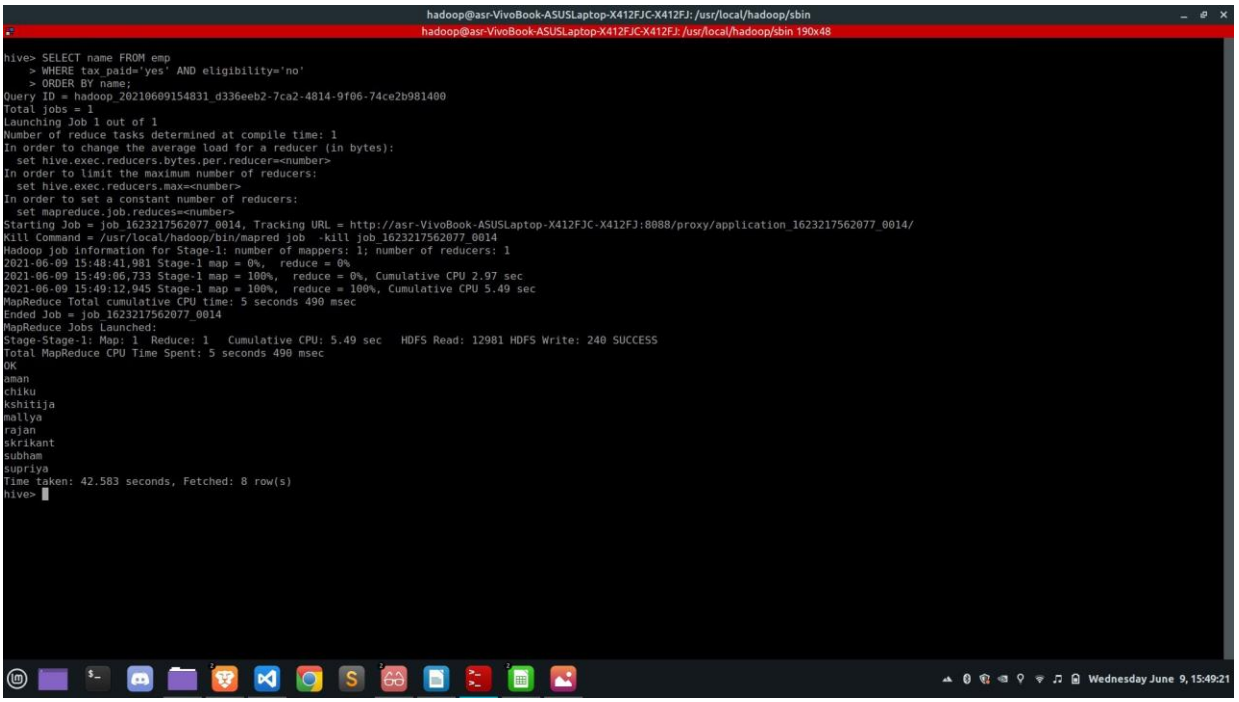
Results and Snapshots



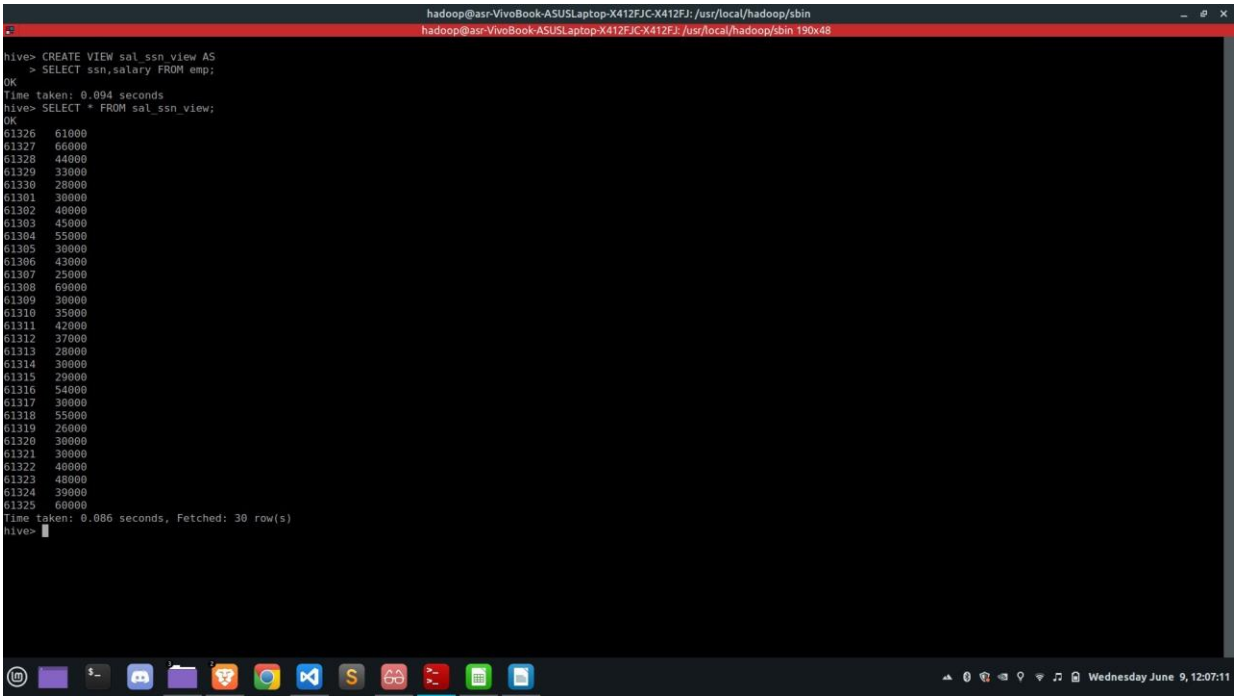


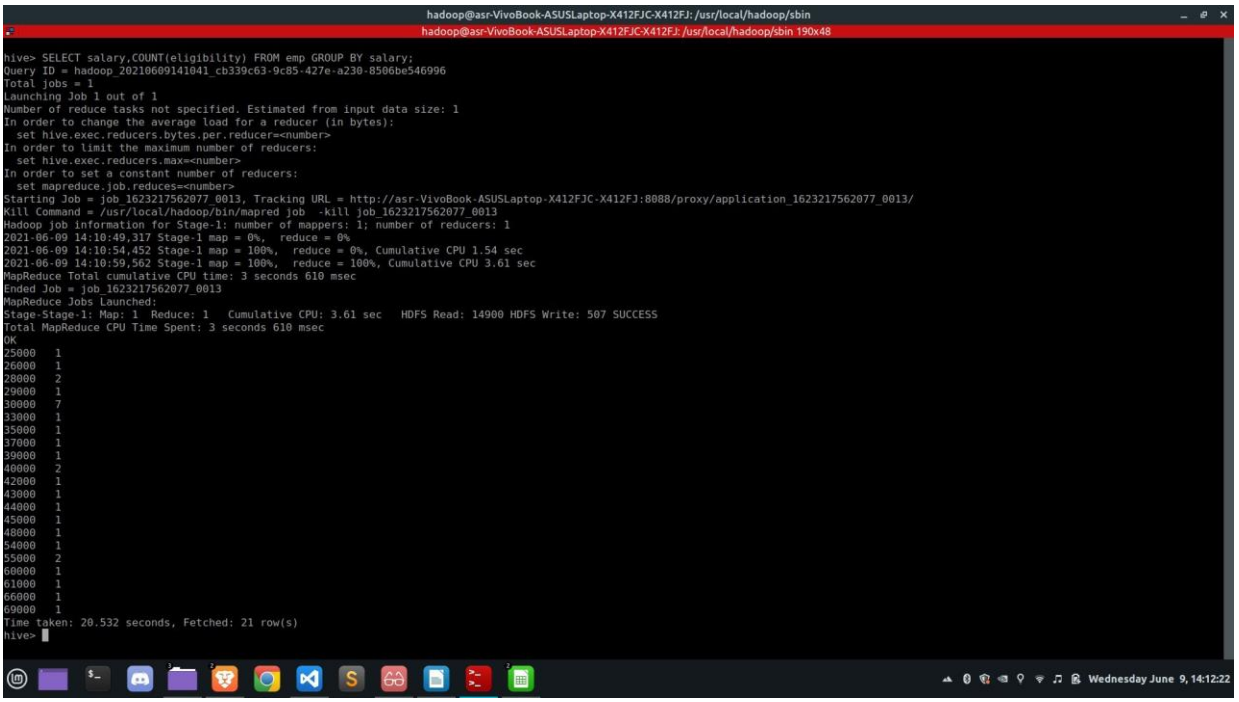
Big data lab- Learning activity II



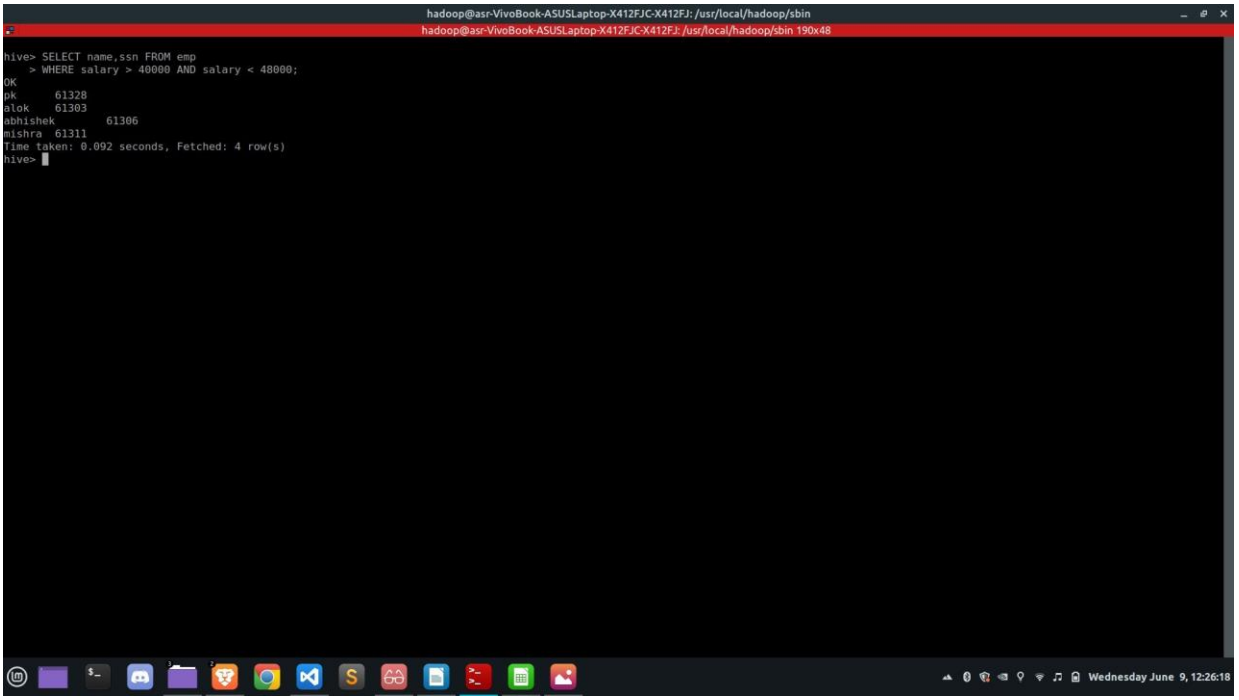


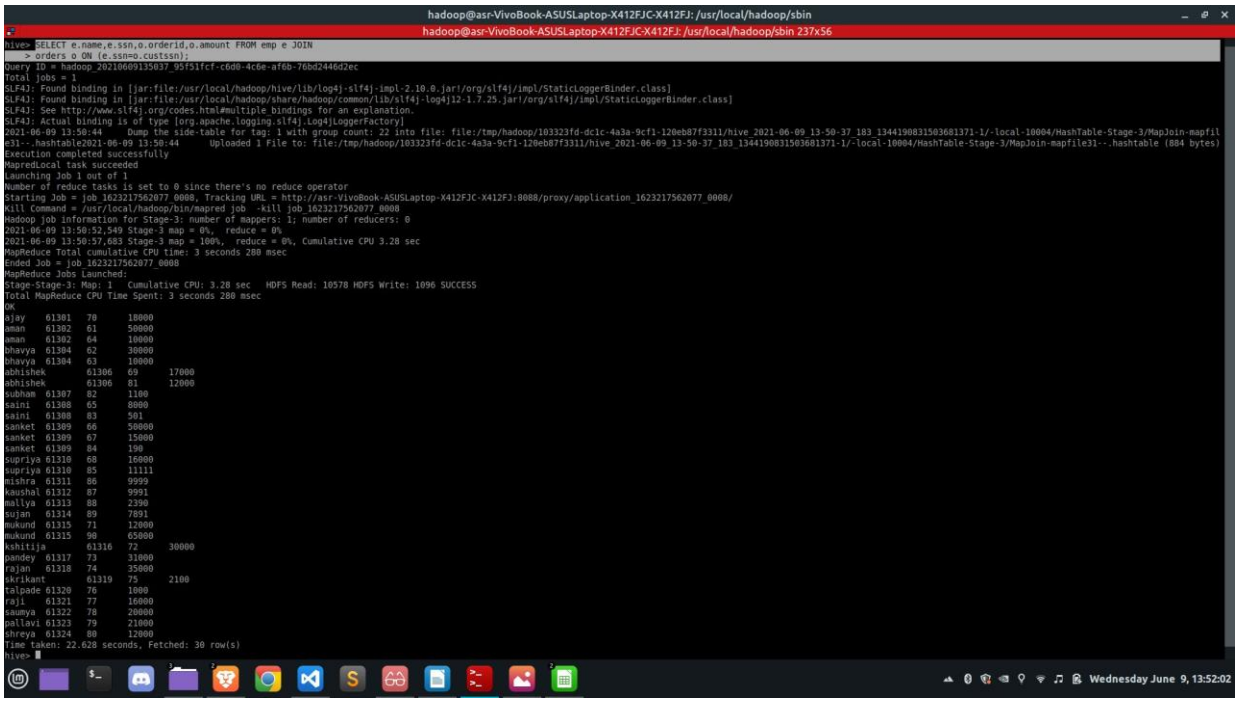
Big data lab- Learning activity II

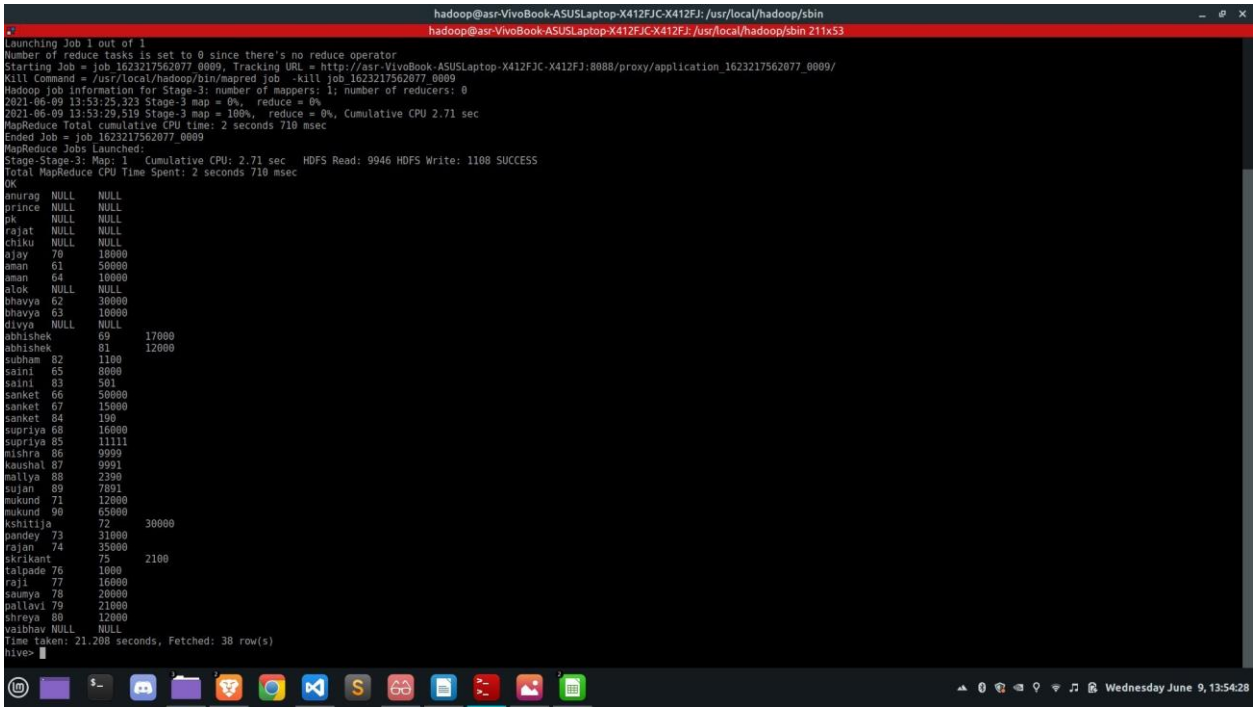


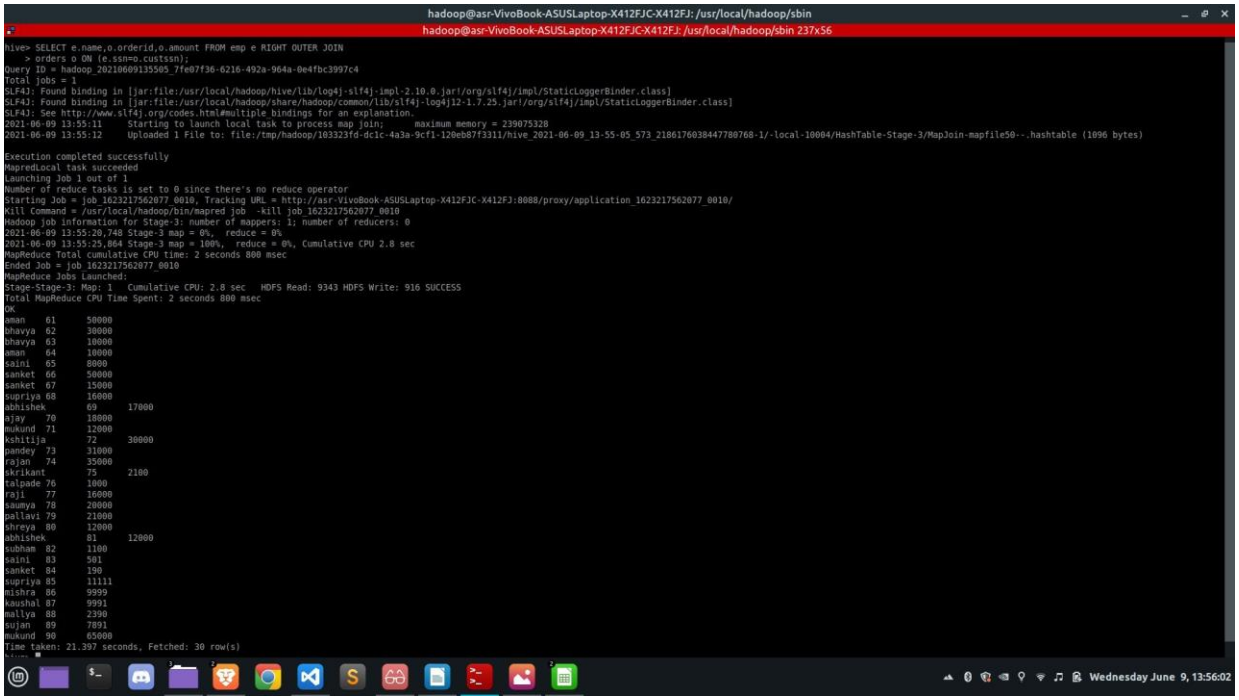


Big data lab- Learning activity II






hadoop@asr-VivoBook-ASUSLaptop-X412FJC-X412FJ: /usr/local/hadoop/sbin
hadoop@asr-VivoBook-ASUSLaptop-X412FJC-X412FJ: /usr/local/hadoop/sbin 211x53
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1623217562077_0009, Tracking URL = http://asr-VivoBook-ASUSLaptop-X412FJC-X412FJ:8088/proxy/application_1623217562077_0009/
Kill Command = /usr/local/hadoop/bin/mapred job -kill job_1623217562077_0009
Hadoop job information for Stage-3: number of mappers: 1; number of reducers: 0
2021-06-09 13:53:25,323 Stage-3 map = 0%, reduce = 0%
2021-06-09 13:53:29,519 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 2.71 sec
MapReduce Total cumulative CPU time: 2 seconds 710 msec
Ended Job = job_1623217562077_0009
MapReduce Jobs Launched:
Stage-Stage-3: Map: 1 Cumulative CPU: 2.71 sec HDFS Read: 9946 HDFS Write: 1108 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 710 msec
OK
anurag NULL NULL
prince NULL NULL
pk NULL NULL
rajat NULL NULL
chiku NULL NULL
ajay 70 18000
aman 61 50000
aman 64 10000
alok NULL NULL
bhavya 62 30000
bhavya 63 10000
divya NULL NULL
abhishek 69 17000
abhishek 81 12000
subham 82 1100
saini 65 8000
saini 83 501
sanket 66 50000
sanket 67 15000
sanket 84 190
supriya 68 16000
supriya 85 11111
mishra 86 9999
kaushal 87 9991
mallya 88 2390
sujan 89 7891
mukund 71 12000
mukund 90 65000
kshitija 72 30000
pandey 73 31000
rajan 74 35000
skrikant 75 2100
talpade 76 1000
raji 77 16000
saumya 78 20000
pallavi 79 21000
shreya 80 12000
vaibhav NULL NULL
Time taken: 21.208 seconds, Fetched: 38 row(s)
hive>
Wednesday June 9, 13:54:28


hadoop@asr-VivoBook-ASUSLaptop-X412FJC-X412FJ: /usr/local/hadoop/sbin
hadoop@asr-VivoBook-ASUSLaptop-X412FJC-X412FJ: /usr/local/hadoop/sbin 237x56
hive> SELECT e.name,o.orderid,o.amount FROM emp e RIGHT OUTER JOIN
> orders o ON (e.ssn=o.custssn);
Query ID = hadoop_20210609135505_7fe07f36-6216-492a-964a-0e4fbc3997c4
Total jobs = 1
SLF4J: Found binding in [jar:file:/usr/local/hadoop/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
2021-06-09 13:55:11 Starting to launch local task to process map join; maximum memory = 239075328
2021-06-09 13:55:12 Uploaded 1 File to: file:/tmp/hadoop/103323fd-dc1c-4a3a-9cf1-120eb87f3311/hive_2021-06-09_13-55-05_573_2186176038447780768-1/-local-10004/HashTable-Stage-3/MapJoin-mapfile50--.hashtable (1096 bytes)
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1623217562077_0010, Tracking URL = http://asr-VivoBook-ASUSLaptop-X412FJC-X412FJ:8088/proxy/application_1623217562077_0010/
Kill Command = /usr/local/hadoop/bin/mapred job -kill job_1623217562077_0010
Hadoop job information for Stage-3: number of mappers: 1; number of reducers: 0
2021-06-09 13:55:20,748 Stage-3 map = 0%, reduce = 0%
2021-06-09 13:55:25,864 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 2.8 sec
MapReduce Total cumulative CPU time: 2 seconds 800 msec
Ended Job = job_1623217562077_0010
MapReduce Jobs Launched:
Stage-Stage-3: Map: 1 Cumulative CPU: 2.8 sec HDFS Read: 9343 HDFS Write: 916 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 800 msec
OK
aman 61 50000
bhavya 62 30000
bhavya 63 10000
aman 64 10000
saini 65 8000
sanket 66 50000
sanket 67 15000
supriya 68 16000
abhishek 69 17000
ajay 70 18000
mukund 71 12000
kshitija 72 30000
pandey 73 31000
rajan 74 35000
skrikant 75 2100
talpade 76 1000
raji 77 16000
saumya 78 20000
pallavi 79 21000
shreya 80 12000
abhishek 81 12000
subham 82 1100
saini 83 501
sanket 84 190
supriya 85 11111
mishra 86 9999
kaushal 87 9991
mallya 88 2390
sujan 89 7891
mukund 90 65000
Time taken: 21.397 seconds, Fetched: 30 row(s)
Wednesday June 9, 13:56:02

Big data lab- Learning activity II

References

1. Hadoop-https://github.com/1NT18IS009/1NT18IS009_abhishek_B_bdLab/tree/master/Programming%20Exercise/Exercise1(hadoop)

2. Hive-https://github.com/1NT18IS009/1NT18IS009_abhishek_B_bdLab/tree/master/Programming%20Exercise/Exercise2(hive)