Started on	Wednesday, 10 November 2021, 2:32 PM
State	Finished
Completed on	Wednesday, 10 November 2021, 4:00 PM
Time taken	1 hour 28 mins
Grade	Not yet graded

Information

A second hand Playing Card seller James is having 10000 assorted cards. With which he would like to build valid decks of cards for reselling. A deck of cards should contain 52 cards with four suits 1.Diamond 2.Clubs 3.Heart and 4.Spade. The details of the cards he is having with him are recorded in a file named deckdetails.csv with 10000 entries. The data of every card is mentioned in the following format [SUIT_NAME RANK INTEGER].For e.g.: SPADE 1. The suit name can be HEART or SPADE or CLUB or DIAMOND respectively and the rank is an integer value ranging from 1 to 13 for the values [A, 2, 3…..J, Q, K]. A sample data set of cards in the file deckdetails.csv is shown below in Fig:01

SPADE 1	DIAMOND 1	DIAMOND 3
SPADE 2	CLUB 4	CLUB 6
SPADE 3	CLUB 5	HEART 8
SPADE 4	HEART 12	HEART 9
DIAMOND 5	CLUB 7	SPADE 6
SPADE 5	CLUB 8	HEART 7
DIAMOND 12	HEART 2	DIAMOND 11
CLUB 12	CLUB 10	DIAMOND 13
CLUB 9	CLUB 11	CLUB 1
SPADE 7	CLUB 13	CLUB 2
DIAMOND 8	HEART 1	HEART 4
DIAMOND 10	HEART 5	CLUB 3
SPADE 9	HEART 3	SPADE 11

Fig:01

Based on the above scenario answer the following.

Question 1
Complete
Marked out of 6

Assume, James process the sample data mentioned in fig:01 using MapReduce paradigm to generate a deck but he found few cards are missing. If the split size of the MapReduce job is 4, then depict the output of the Mapper phase, Shuffle and sort phase and Reduce Phase.

(6 Marks-[Ap/C,2])(CO:4 ; PO:1.4.1)

INPUT	MAPPER	SHUFFLE & SORT	REDUCE	FINAL
SPADE 1	SPADE 1	CLUB 1		
SPADE 2	SPADE 2	CLUB 10		
SPADE 3	SPADE 3	CLUB 11		
SPADE 4	SPADE 4	CLUB 12		
DIAMOND 5		CLUB 13	CLUB-13	
SPADE 5	DIAMOND 5	CLUB 2		
DIAMOND 12	SPADE 5	CLUB 3		
CLUB 12	DIAMOND 12	CLUB 4		CLUB 13
CLUB 9	CLUB 12	CLUB 5		DIAMOND 8
SPADE 7		CLUB 6		HEART 9
DIAMOND 8	CLUB 9	CLUB 7		SPADE 9
DIAMOND 10	SPADE 7	CLUB 8		
SPADE 9	DIAMOND 8	CLUB 9		
DIAMOND 1	DIAMOND 10			
CLUB 4		DIAMOND 1		
CLUB 5	SPADE 9	DIAMOND 10		
HEART 12	DIAMOND 1	DIAMOND 11		
CLUB 7	CLUB 4	DIAMOND 12	DIAMOND-8	
CLUB 8	CLUB 5	DIAMOND 13		
HEART 2		DIAMOND 3		
CLUB 10	HEART 12	DIAMOND 5		
CLUB 11	CLUB 7	DIAMOND 8		
CLUB 13	CLUB 8			
HEART 1	HEART 2	HEART 1		
HEART 5		HEART 12		

HEART 3	CLUB 10	HEART 2	
DIAMOND 3	CLUB 11	HEART 3	
CLUB 6	CLUB 13	HEART 4	HEART-9
HEART 8	HEART 1	HEART 5	
HEART 9		HEART 7	
SPADE 6	HEART 5	HEART 8	
HEART 7	HEART 3	HEART 9	
DIAMOND 11	DIAMOND 3		
DIAMOND 13	CLUB 6	SPADE 1	
CLUB 1		SPADE 11	
CLUB 2	HEART 8	SPADE 2	
HEART 4	HEART 9	SPADE 3	
CLUB 3	SPADE 6	SPADE 4	SPADE-9
SPADE 11	HEART 7	SPADE 5	
		SPADE 6	
	DIAMOND 11	SPADE 7	
	DIAMOND 13	SPADE 9	
	CLUB 1		
	CLUB 2		

Question 2

Complete

Marked out of 2

For finding the missing cards from the deck. The Map Reducer, first sorts and shuffles the input cards in various phases and then outputs the cards in all the four suits. MRUnitTesting is used to validate the execution of MapReduce separately in all its major phases to ensure the correctness of its working. Before testing any process the below given code annotated @Before is exeuted to initialize and set up the environment variables required for testing. Help James in completing the code so that he can set up his testing environment successfully in order to ensure that the execution is working as expected. Also justify the significance of the Code in Line 1 and Line 2

```
@Before
 public void setUp()
 {
 WordCountMapper mapper = new deck_countMapper();
 mapDriver = new MapDriver<LongWritable, Text, Text, IntWritable>();
 mapDriver.setMapper(mapper);
 WordCountReducer reducer = new deck_countReducer();
 reduceDriver = new ReduceDriver<Text, IntWritable, Text, IntWritable>();
 reduceDriver.setReducer(reducer);

 mapReduceDriver = new MapReduceDriver<LongWritable,
 Text, Text,IntWritable, Text, IntWritable>();
 ..............................\ Line 1
 ..............................\ Line 2
 }
```

(2 Marks-[Ap/C,2])(CO:4 ; PO :1.4.1)

Line 1 and Line 2:

mapReduceDriver.setMapper(mapper);

mapReduceDriver.setReducer(reducer);

Significance of the Code in Line 1 and Line 2

when we want to test the MapRedcue job as a whole, we need to test both the mapper and the reducer. Therefore we need to set both the mapper and reducer class to the mapReduceDriver.

Question 3

Complete

Marked out of 2

In local testing, among all possible job configuration parameters, setting the input and output file path to all phases is important as shown below.

```
JobConf conf = new JobConf(getConf(), getClass());
conf.setJobName("DeckCount");
FileInputFormat.addInputPath(conf, new
Path(args[0]));
FileOutputFormat.setOutputPath(conf, new
Path(args[1]));
conf.setOutputKeyClass(Text.class);
conf.setOutputValueClass(IntWritable.class);
conf.setMapperClass(DeckCountMapper.class);
conf.setCombinerClass(DeckCountReducer.class);
conf.setReducerClass(DeckCountReducer.class);
JobClient.runJob(conf);
```

Justify the statement in few lines.

(2 Marks-[An/C,2])(CO:4 ; PO:4.1.1)

Among the possible job configuration parameters, we set the input and output file paths, the mapper, reducer and combiner classes, and the output types (the input typesare determined by the input format, which defaults to TextInputFormat and has Long Writable keys and Text values).

Question **4**

Complete

Marked out of 6

The anatomy of classic Hadoop job run can be listed as a sequence of steps as given below. Map the steps in the given diagram appropriately:

Submit job

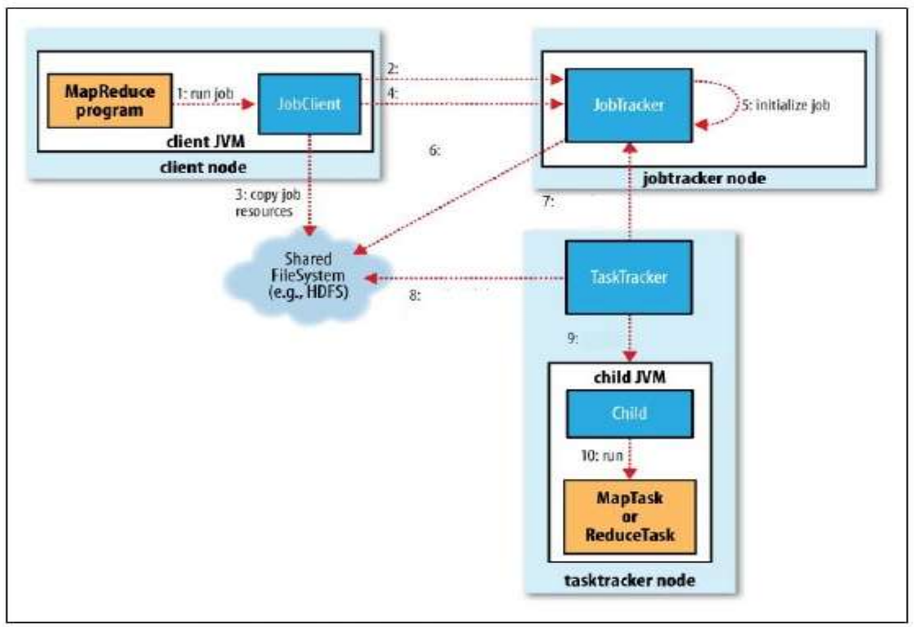
Retrieve job resources

Get new job ID

Launch

Retrieve the input files

Heartbeat (Returning task)



How Hadoop runs a MapReduce job

(6 Marks-[An/C,2])(CO:4 ; PO:4.1.1)

1. run job

2. Get new job ID

3. copy job resource

4. submit job

5. initialize job

6. Retrieve the input files

7.Heartbeat (Returning task)

8. Retrieve job resources

9.launch

10. run

Question **5**

Complete

Marked out of 4

Apache Yarn is also called as Hadoop 2.0 which is mainly used for handling MapReduce, resource management and job scheduling in an efficient way. The following diagram depicts the architecture of Apache Yarn. Identify and Map the suitable components listed below in its appropriate blocks of the diagram:

1) Client

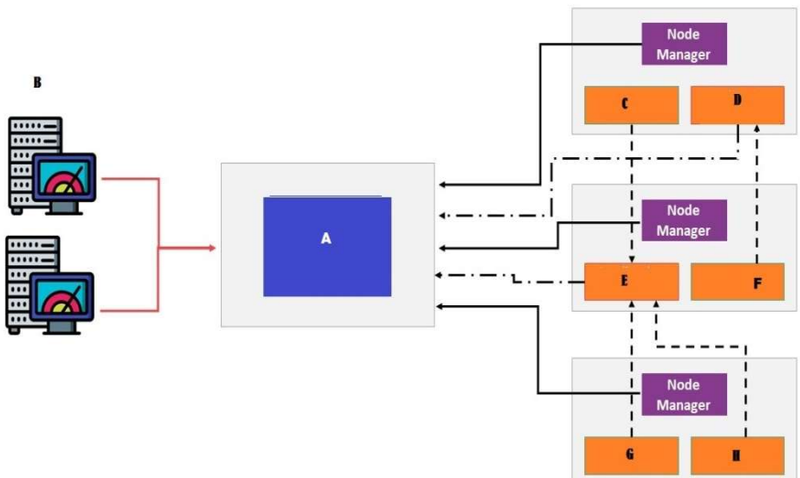
2) Resource Manager

3) Application Master

4) Container

5) Application Manager

6) Mapper Reducer



. (CO:4 ; PO: 1.4.1) (4 Marks -U/C,1])

A - Resource Manager

B - Client

C - Container

D - Application Master

E - Application Master

F - Container

G - Container

H - Container

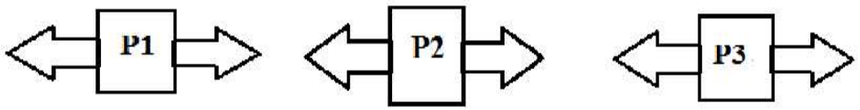
Information

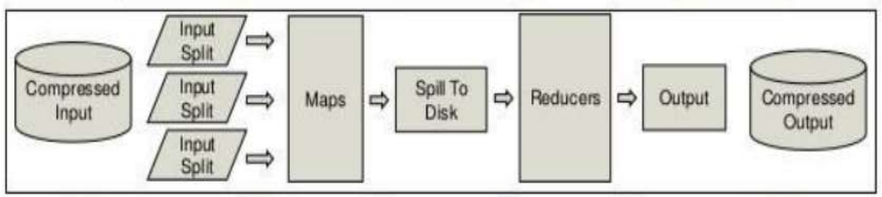
A climate model is a mathematical representation of climate systems based on various factors that impacts the climate of the Earth. Basically, it describes the interaction of various drivers of climate like ocean, sun, atmosphere, etc. to provide an insight into the dynamics of the climate system. It is used to project climate conditions by simulating the climate changes based on factors that affect climate. Regional Climate Model Evaluation System (RCMES) is required for analysis and evaluation of the climate output model against billions of remote sensing data present in various external repositories.

Based on the above scenario answer the following questions

Question 6
Complete
Marked out of 3

NASA's Jet Propulsion Laboratory has developed RCMES with two components such as RCMED (Regional Climate Model Evaluation Database) and RCMET (Regional Climate Model Evaluation Toolkit). They used Hadoop for loading billions of remote sensor data and evaluating them based on different parameters required for modelling climate. While storing the data they have applied a suitable compression technique. Identify the different phases PI, P2, P3 given in the MapReduce pipeline while applying the process of compression.





(3 Marks -[Ap/C,2])(CO-3;PO;1.4.1)

P1 - Map

P2 - Shuffle and sort

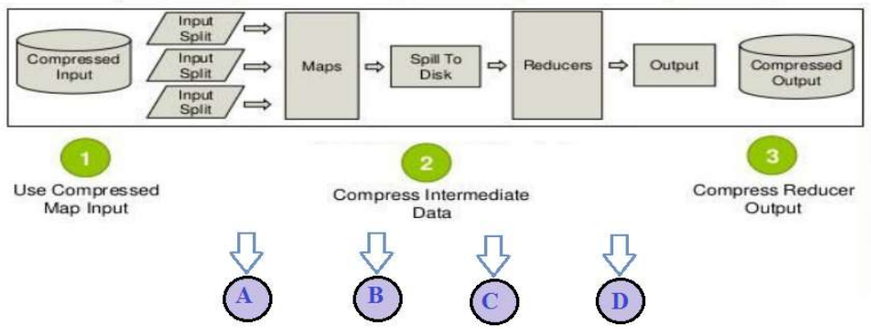
P3 - Reduce

Question 7

Complete

Marked out of 2

Derive the type of Input / Output obtained in A, B, C and D during the process of compressing intermediate data given in the workflow..



(2 Marks -[Ap/C,2]) **(CO:3 ; PO : 1.4.1)**

A - Decompress input

B - Compress output

C - Decompress input

D - Compress output

Question 8
Complete
Marked out of 4

Compression reduces the space needed to store files in the file system and speeds-up the data transfer from one node to another in the network. Match the following compression techniques with its algorithm, tool and its feature.

S. No	Compression Format	Algorithm	Tool	Feature
1	Gzip	LZ4	Lzop	Non Splittable
2	LZO	DEFLATE	NA	Best Compression ratio
3	LZ4	Bzip2	Bzip2	Slower than LZO and LZ4
4	Bzip2	LZO	Gzip	Balanced Compression/Decompression

(4 Marks -[An/C,2]) **(CO:3 ; PO : 2.3.1)**

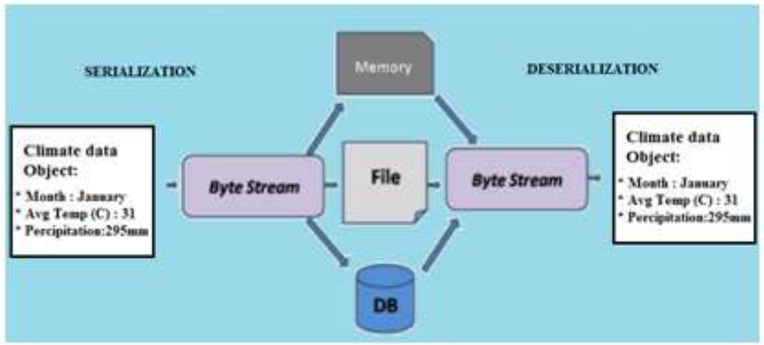
S.NO	Compression Format	Algorithm	Tool	Feature
1	Gzip	deflate	gzip	slower than lzo and lz4
2	LZO	LZO	lzop	non spilttable
3	LZ4	LZ4	na	balanced compression/ decompression
4	Bzip2	bzip2	bzip2	best compression ratio

Question 9

Complete

Marked out of 1

Serialization and deserialization is applied in inter process communication between objects while storing or transmitting them from one file to another.



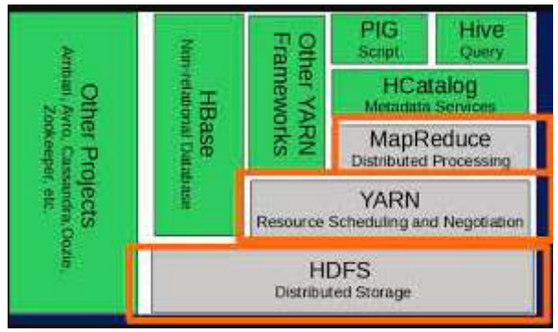
Determine the format to which the data objects are converted while saving the state of the object.

(1 Marks-[An/C,2])(CO:3 ;PO:2.3.1)

JSON FILE FORMAT IS USED WHILE SAVING

Information

Hadoop is an open-source framework to store and process Big Data in a distributed environment. The Hadoop ecosystem contains different sub-projects (tools) that are used to help Hadoop modules.



Question 10
Complete
Marked out of 4

Match the following Hadoop modules based on their data model, architecture Rad-Write modes, Data Access etc:

Hadoop	Data Model	Architecture	R/W	Data Access
HBASE	Data is partitioned	Metastore	Better WRITE	Cell –level
CASSANDRA	Nested relational	Master based	Query	Schema level
HIVE	Datawarehouse Tool	Multi-query approach	Scripts	Row level
PIG	Columns are grouped	Masterless	Better Read	Schema level

(CO:5; PO:4.1.1)(4 Marks-[An/C,2])

Hadoop	Data model	Architecture	R/W	Data Access
HBASE	Columns are grouped	Master based	Better Read	Cell - level
CASSANDRA	Nested Relational	Masterless	Query	schema Level
HIVE	Data Warehouse Tool	Metastore	Better Write	Schema Level
PIG	Data is partitioned	Multi-Query	Scripts	Row Level

Question 11
Complete
Marked out of 6

The following code is used to create table in HBASE.

```
HBaseAdmin admin= new HBaseAdmin(config);
HColumnDescriptor []column;
column= new HColumnDescriptor[2];
column[0]=new HColumnDescriptor("columnFamily1:");
HTableDescriptor desc= new HTableDescriptor(Bytes.toBytes("MyTable"));
desc.addFamily(column[0]);
admin.createTable(desc);
```

Refine the HBASE code with suitable logic to add two more columns
Refine the HBASE code using GET() method to return the corresponding record with the help of Key
Refine the HBASE code using PUT() method to return the corresponding record
(CO:5; PO:4.1.1) (6 Marks-[An/C,3])

HBASE code using GET() method to return the corresponding record with the help of Key

```
Get get = new Get(Bytes.toBytes("row1"));

Result r = htable.get(get);

byte[] b = r.getValue(Bytes.toBytes("cf"), Bytes.toBytes("attr")); // returns current version of value
```

Refine the HBASE code using PUT() method to return the corresponding record

```
Put put = new Put (Bytes.toBytes(row));

put.add(Bytes.toBytes("cf"), Bytes.toBytes("attr1"), Bytes.toBytes(data));

htable.put(put);
```

Question 12
Complete
Marked out of 2

"Create Keyspace Command in cassandra is used to create keyspace in Cassandra.Replication factor is the number of replicas of data placed on different nodes"

Construct a keyspace and determine the replication factor to ensure no single point of failure. Sometimes, the server can be down, or network problem can occur, then other replicas should be able to provide service with no failure.

(CO:5 ; PO:4.1.1) (2 Marks-[AP/C,2])

Create keyspace KEY_NAME with replication={'class':'SimpleStrategy','replication_factor':3};

Information

"Hive is a data warehousing system to store structured data on Hadoop file system.Hive Buckets in hive is used in segregating of hive table-data into multiple files or directories. It is used for efficient querying. Partitions is a way to organizes tables into partitions by dividing tables into different parts based on partition keys"

Based on this answer the following questions

Question 13
Complete
Marked out of 2

Query for Create Dept_bucket with column names such as first_name, job_id, department name ,department id, salary and country is given below

create table Dept_bucket first_name string , job_id int , dept_name string , dept_id int ,salary string ,country string);

Construct four buckets for the “country” attribute.

(CO:5 ; PO:4.1.1)(2 Marks-[AP/C,2])

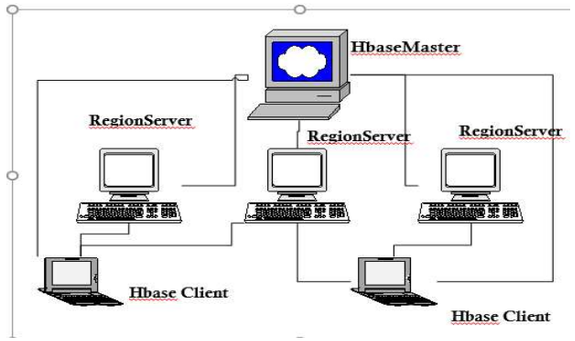
CREATE TABLE Dept_bucket PARTITIONED BY (string ,int,string ,int ,string ,string) CLUSTERED BY (first_name ,job_id ,dept_name,dept_id,salary, country) SORTED BY (country[ASC]) INTO 4 BUCKETS;

Question **14**

Complete

Marked out of 4

Abstract the roles of the HBase master, Region Server and HBase Client given in the architecture below.



(CO:5 ; PO:4.1.1)(4 Marks-[U/C,1])

HBase master

- HMaster is the implementation of a Master server in HBase architecture.
- It acts as a monitoring agent to monitor all Region Server instances present in the cluster and acts as an interface for all the metadata changes.
- In a distributed cluster environment, Master runs on NameNode. Master runs several background threads.

Region Server

- When Region Server receives writes and read requests from the client, it assigns the request to a specific region, where the actual column family resides.
- However, the client can directly contact with HRegion servers, there is no need of HMaster mandatory permission to the client regarding communication with HRegion servers.
- The client requires HMaster help when operations related to metadata and schema changes are required.
- HRegionServer is the Region Server implementation

HBase Client

Client wants to write data and in turn first communicates with Regions server and then regions

Question 15
Complete
Marked out of 2

Determine the PIG Grunt Queries for the following operations.

To specify the number of tuples from the relations.

To extract the data from file system to relation.

To combine the data in a single relation.

To arrange the relations in sorted form

(CO:5 ;PO:4.1.1)(2 Marks-[U/C,1])

- **LIMIT** - To specify the number of tuples from the relations
- **LOAD** - To extract the data from file system to relation0.
- **JOIN** - To combine the data in a single relation.
- **ORDER BY** -To arrange the relations in sorted form

« PREVIOUS ACTIVITY
18CS005_BigDataAnalytics_24-09-21