**Question 1 – it**

**Instructions/Data Description/Output Requirements**

* Details - Duration 40 minutes
  + Data set is in folder /user/cloudera/data\_prep/crime
  + Structure of data (ID,Case Number,Date,Block,IUCR,Primary Type,Description,Location Description,Arrest,Domestic,Beat,District,Ward,Community Area,FBI Code,X Coordinate,Y Coordinate,Year,Updated On,Latitude,Longitude,Location)
  + File format - text file
  + Delimiter - “,”
  + Get monthly count of primary crime type, sorted by month in ascending and number of crimes per type in descending order
  + Store the result in HDFS path /user/cloudera/DA\_SPARK/question1/output
  + Output File Format: TEXT
  + Output Columns: month in YYYY-MM format, crime\_count, crime\_type
  + Output Delimiter: \t (tab delimited)
  + Output Compression: gzip

KEYWORDS: CSVREADOPTIONS, TEXTWRITEOPTIONS, DATE, GROUPBY, ORDERBY

**Question 1A – it**

**Instructions/Data Description/Output Requirements**

* Details - Duration 15 to 20 minutes
  + Data is available in HDFS file system under /user/cloudera/data\_prep/crime
  + Structure of data (ID,Case Number,Date,Block,IUCR,Primary Type,Description,Location Description,Arrest,Domestic,Beat,District,Ward,Community Area,FBI Code,X Coordinate,Y Coordinate,Year,Updated On,Latitude,Longitude,Location)
  + File format - text file
  + Delimiter - “,” (use regex while splitting split(",(?=(?:[^\"]\*\"[^\"]\*\")\*[^\"]\*$)", -1), as there are some fields with comma and enclosed using double quotes.
  + Get top 3 crime types based on number of incidents in RESIDENCE area using “Location Description”
  + Store the result in HDFS path /user/cloudera/DA\_SPARK/question1A/output
  + Output Fields: crime\_type, num\_incidents
  + Output File Format: JSON
  + Output Delimiter: N/A
  + Output Compression: No

KEYWORDS: CSVREADOPTIONS, JSONWRITEOPTIONS, FILTER, GROUPBY, ORDERBY, LIMIT

**Question 2 – it**

**Instructions/Data Description/Output Requirements**

* Details - Duration 15 to 20 minutes
  + Data is available in local file system /home/cloudera/data\_prep/retail\_db
  + Source directories: orders and customers
  + Source delimiter: comma (“,”)
  + Source Columns - orders - order\_id, order\_date, order\_customer\_id, order\_status
  + Source Columns - customers - customer\_id, customer\_fname, customer\_lname and many more
  + Get the customers who have not placed any orders, sorted by customer\_lname and then customer\_fname
  + Target Columns: customer\_lname, customer\_fname
  + Number of files - 1
  + Target Directory: /user/cloudera/DA\_SPARK/question2/output
  + Target File Format: TEXT
  + Target Delimiter: commaspace(“, ”)
  + Compression: N/A

KEYWORDS: LOCALFSREAD, TEXTWRITEOPTIONS, JOIN, GROUPBY, ORDERBY

**Question 3 – it**

**Instructions/Data Description/Output Requirements**

* Details - Duration 20 minutes
  + Data is available in HDFS /user/cloudera/data\_prep/randomtextwriter
  + Get word count for the input data using space as delimiter (for each word, we need to get how many times it is repeated in the entire input data set), ordered by most frequent words
  + Number of executors should be 10
  + Executor memory should be 3 GB
  + Executor cores should be 20 in total (2 per executor)
  + Number of output files should be 8
  + Avro dependency details: groupId -> com.databricks, artifactId -> spark-avro\_2.11, version -> 4.0.0
  + Target Directory: /user/cloudera/DA\_SPARK/question3/output
  + Target File Format: Avro
  + Target fields: word, count
  + Compression: N/A or default

**Question 4 – it**

**Instructions/Data Description/Output Requirements**

Duration: 20 to 30 minutes

* Tables are to be used in hive database - /user/cloudera/data\_prep/retail\_db\_txt
  + orders
  + order\_items
  + customers
* Time to create database and tables need not be counted.
* Make sure to go back to Spark SQL module and create tables and load data
* Get details of top 5 customers by revenue for each month
* We need to get all the details of the customer along with month and revenue\_per\_month
* Data need to be sorted by month in ascending order and revenue per month in descending order
* Create table top5\_customers\_per\_month in path - /user/cloudera/DA\_SPARK/question4/output
* Insert the output into the newly created table

**Question 5 – nd**

**Instructions/Data Description/Output Requirements**

* Join the data at hdfs location /user/cloudera/data\_prep/retail\_db/orders/ & /user/cloudera/ data\_prep /retail\_db/customers\_3cols/
* to find out customers whose orders status is like "pending"
* Schema for customer File - Customer\_id,customer\_fname,customer\_lname
* Schema for Order File -Order\_id,order\_date,order\_customer\_id,order\_status
* Output Requirement:
* Output should have customer\_id,customer\_fname,order\_id and order\_status.
* Result should be saved in /user/cloudera/DA\_SPARK/question5/output

**Question 6 – nd**

**Instructions/Data Description/Output Requirements**

* Count number of customers grouped by customer city and customer first name where customer\_fname is like "Mary" and order the results by customer first name and save the result as text file.
* Input folder is- /user/cloudera/data\_prep/retail\_db/customer\_part\_parquet
* Output Requirement:
* Result should have customer\_city,customer\_fname, count and output should be saved in - /user/cloudera/DA\_SPARK/question6/output
* as text file with fields separated by pipe character

**Question 7 – nd**

**Instructions/Data Description/Output Requirements**

* Find all customers that lives 'Brownsville' city and save the result into HDFS.
* Input folder is - /user/cloudera/data\_prep/retail\_db/customer\_part\_text\_tab – it is tab delimited file – input schema is customer\_id,customer\_fname,customer\_city
* Output Requirement:
* Result should be saved in /user/cloudera/DA\_SPARK/question7/output
* Output file should be saved in Json format

**Question 8 – nd**

**Instructions/Data Description/Output Requirements**

* Get products from metastore table named retail\_db\_txt.product\_replica whose product\_price > 100 and save the results in HDFS in parquet format.
* Output Requirement:
* Result should be saved in /user/cloudera/DA\_SPARK/question8/output as parquet file
* Files should be saved in Gzip compression.

**Question 9 – nd**

**Instructions/Data Description/Output Requirements**

* Get all customers who have placed order of amount more than 200.
* Input files are tab delimeted files placed at below HDFS location:
* $HDFS\_DP/retail\_db/orders\_tab
* $HDFS\_DP/retail\_db/customers\_tab
* $HDFS\_DP/retail\_db/order\_items\_tab

Schema for customers File

* Customer\_id,customer\_fname,customer\_lname,customer\_email,customer\_password,customer\_street,customer\_city,customer\_state,customer\_zipcode

Schema for Orders File

* Order\_id,order\_date,order\_customer\_id,order\_status

Schema for Order\_Items File

* Order\_item\_id,Order\_item\_order\_id,order\_item\_product\_id,Order\_item\_quantity,Order\_item\_subtotal,Order\_item\_product\_price

Output Requirements:

* Output should be placed in below HDFS Location
* /user/cloudera/DA\_SPARK/question9/output
* Output file should be comma seperated file with customer\_fname,customer\_lname,customer\_city,order\_amount

**Question 10 – nd**

**Instructions/Data Description/Output Requirements**

* Hive database retail\_db\_txt
* hive-table customers\_hive

Instructions:

* Get Customers from metastore table named "customers\_hive" whose fname is like "Rich" and save the results in HDFS in text format.
* Output Requirement:
* Result should be saved in
* /user/cloudera/DA\_SPARK/question10/output
* as text file.
* Output should contain only customer\_fname, customer\_lname and customer \_city
* customer\_fname and customer\_lname should seperated by tab with customer\_city seperated by colon
* order by customer \_city,customer\_lname and customer\_fname

Sample Output

* Richard Plaza:Francisco
* Rich Smith:Chicago

**Question 11 – nd**

**Instructions/Data Description/Output Requirements**

* Provided tab delimited file, get total numbers customers in each state whose first name starts with 'M' and save results in HDFS in json format.Input Schems is customer\_id,customer\_fname,customer\_state
* Input folder
* $HDFS\_DP/retail\_db/customers\_part2\_tab
* Output Requirement:
* Result should be saved in /user/cloudera/DA\_SPARK/question11/output
* Output should have state name followed by total number of customers in that state.
* order by total number of customers in that state descending

**Question 12 – nd**

**Instructions/Data Description/Output Requirements**

* Provided a meta-store table named retail\_db\_txt.products
* consisting of product details, find the most expensive product in each category.
* Output Requirement:
* Output should have
* product\_category\_id ,product\_name,product\_price,rank.
* Result should be saved in
* /user/cloudera/DA\_SPARK/question12/output
* as pipe delimited text file
* order by product\_category\_id, product\_name