Done By:

Arvind Shankar Raman – AXS170059

**MongoDB Project – Google Store Visitor Data**

BUAN 6320.00#

Contents

[Data Review 3](#_Toc531516932)

[Assumptions/Notes About Data Collections, Attributes and Relationships between Collections 3](#_Toc531516933)

[Physical Database 4](#_Toc531516934)

[Assumptions/Notes About Data Set 4](#_Toc531516935)

[Screen shot of Physical Database objects (Database, Collections and Attributes) 4](#_Toc531516936)

[Data in the Database 5](#_Toc531516937)

[MongoDB Queries/Code 6](#_Toc531516938)

[Query 2 6](#_Toc531516939)

[Question 6](#_Toc531516940)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 6](#_Toc531516941)

[Translation 6](#_Toc531516942)

[Screen Shot of MongoDB Query/Code and Results 6](#_Toc531516943)

[Query 4 7](#_Toc531516944)

[Question 7](#_Toc531516945)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 7](#_Toc531516946)

[Translation 7](#_Toc531516947)

[Screen Shot of MongoDB Query/Code and Results 7](#_Toc531516948)

[Query 5 8](#_Toc531516949)

[Question 8](#_Toc531516950)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 8](#_Toc531516951)

[Translation 8](#_Toc531516952)

[Screen Shot of MongoDB Query/Code and Results 8](#_Toc531516953)

[Query 6 9](#_Toc531516954)

[Question 9](#_Toc531516955)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 9](#_Toc531516956)

[Translation 9](#_Toc531516957)

[Screen Shot of MongoDB Query/Code and Results 9](#_Toc531516958)

[Query 7 10](#_Toc531516959)

[Question 10](#_Toc531516960)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 10](#_Toc531516961)

[Translation 10](#_Toc531516962)

[Screen Shot of MongoDB Query/Code and Results 10](#_Toc531516963)

[Query 8 11](#_Toc531516964)

[Question 11](#_Toc531516965)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 11](#_Toc531516966)

[Visitors from a total of 208 countries visited the store more than once. 11](#_Toc531516967)

[Rows in result: 208. 11](#_Toc531516968)

[Translation 11](#_Toc531516969)

[Screen Shot of MongoDB Query/Code and Results 11](#_Toc531516970)

[Query 1 13](#_Toc531516971)

[Question 13](#_Toc531516972)

[Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result) 13](#_Toc531516973)

[Translation 13](#_Toc531516974)

[Screen Shot of MongoDB Query/Code and Results 13](#_Toc531516975)

# Data Review

## Assumptions/Notes About Data Collections, Attributes and Relationships between Collections

1.The entire google store dataset was imported into MongoDB as a single data collection called ‘gstore’.

2.This collection has a total of 804664 rows after the removal of 20 rows with missing values.

3.The import was done using the following method:

* Imported the gstore.csv file to R.
* Converted the gstore.csv file into a JSON file.
* Imported the JSON file to MongoDB.

Import code implemented in R:



4.The data collection ‘gstore’ is created under the database ‘testg’.

5.Since a single collection houses the entire data, there is no involvement/relationship with any other collection.

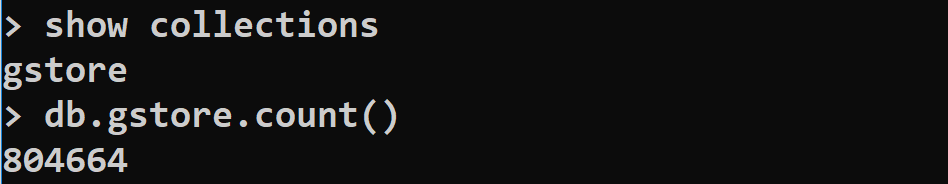
# Physical Database

## Assumptions/Notes About Data Set

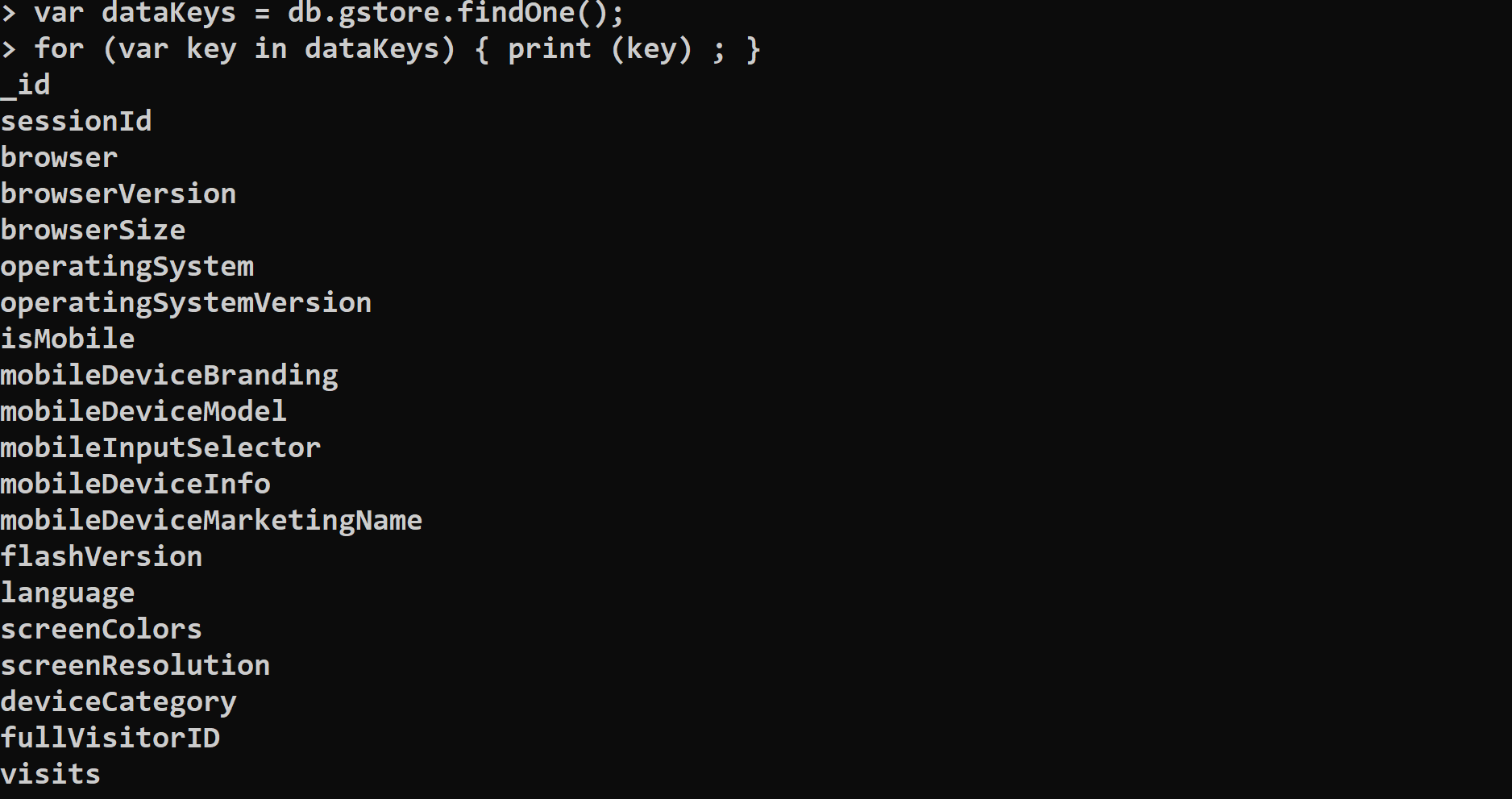
The imported gstore dataset has 43 attributes in total and there are no missing values/NA values present.

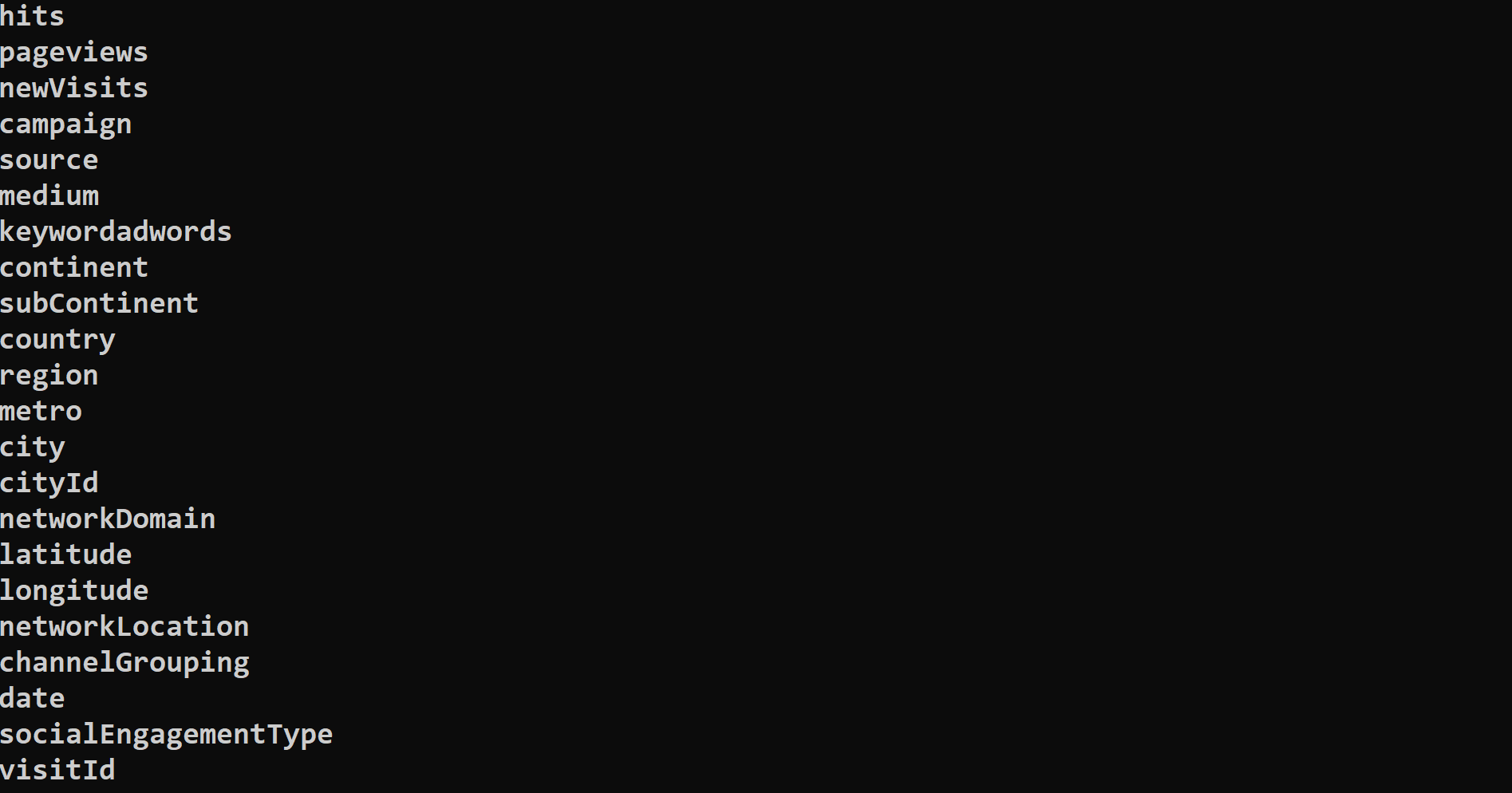
## Screen shot of Physical Database objects (Database, Collections and Attributes)

1.Collections and count of no. of rows



2.Following are the screenshots of the attributes present in the dataset.







# Data in the Database

|  |  |  |
| --- | --- | --- |
| **Collection Name** | **Relationships With Other Collections (if any)** | **# of Rows in Table** |
| gstore | N/A | 804664 |

## MongoDB Queries/Code

## Query 2

### Question

Which operating system (devices) was the most popular amongst store visitors with mobile devices?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

The most popular amongst store visitors with mobile devices is “Android”.

No. of rows in result: 3098.

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

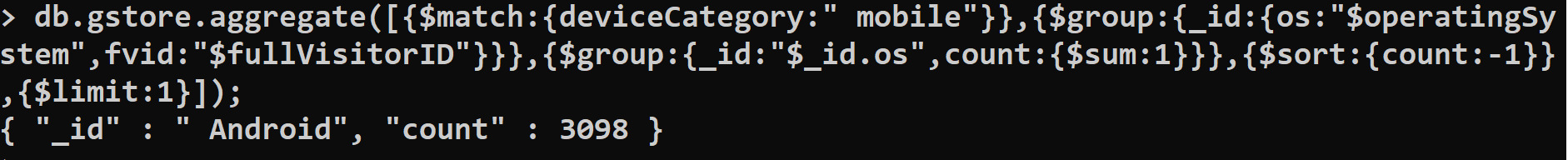
MATCH the “device Category = mobile”.

GROUP by “Operating system”, “full Visitor ID” and get the COUNT to know the no. of users for each operating system.

SORT the result in DESCENDING order and LIMIT the result to the 1st row to know which operating system was the most popular amongst store visitors with mobile devices.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$match:{deviceCategory:" mobile"}},{$group:{\_id:{os:"$operatingSystem",fvid:"$fullVisitorID"}}},{$group:{\_id:"$\_id.os",count:{$sum:1}}},{$sort:{count:-1}},{$limit:1}]);



## Query 4

### Question

Users of which operating system were the least socially engaged?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

The operating system which is least socially engaged is Nokia.

No. of rows in result: 1

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

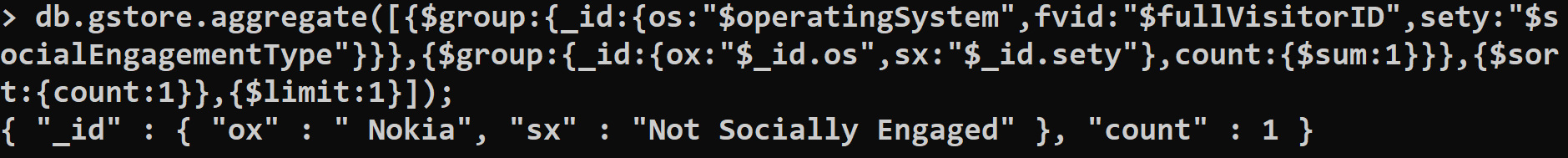
GROUP by “Operating System”, “full Visitor ID” and “Social Engagement Type”.

SORT the result in ASCENDING order since the least no. of visitor ID’s which are not socially engaged are present at the start of the result set.

LIMIT the result to 1st row to know the least socially engaged operating system.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$group:{\_id:{os:"$operatingSystem",fvid:"$fullVisitorID",sety:"$socialEngagementType"}}},{$group:{\_id:{ox:"$\_id.os",sx:"$\_id.sety"},count:{$sum:1}}},{$sort:{count:1}},{$limit:1}]);



## Query 5

### Question

Provide a breakdown of unique visitors by mobile vs nonmobile users

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

Breakdown of unique visitors by mobile vs nonmobile users:

* Mobile users: 3326.
* Non-Mobile users: 3610

No. of rows in result: 2

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

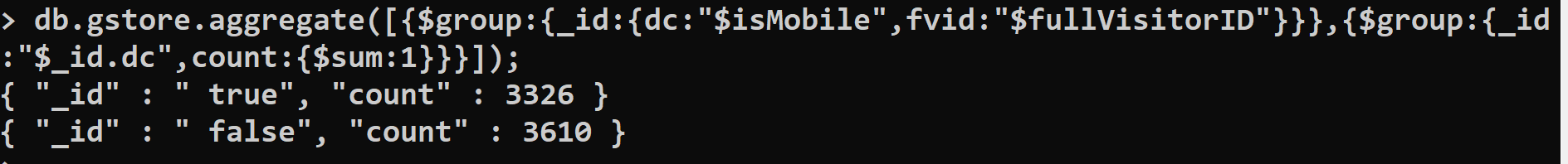
GROUP by “Is Mobile” and “full visitor ID”.

Is Mobile is divided into TRUE for mobile users and FALSE for non-mobile users.

Get a COUNT for no. of unique visitors by mobile (TRUE) and non-mobile(FALSE) to obtain the breakdown of unique visitors by mobile vs non-mobile users.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$group:{\_id:{dc:"$isMobile",fvid:"$fullVisitorID"}}},{$group:{\_id:"$\_id.dc",count:{$sum:1}}}]);



## Query 6

### Question

How many users used only iOS devices to visit the store?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

2917 users used only iOS devices to visit the store.

No. of rows in result: 1

### Translation

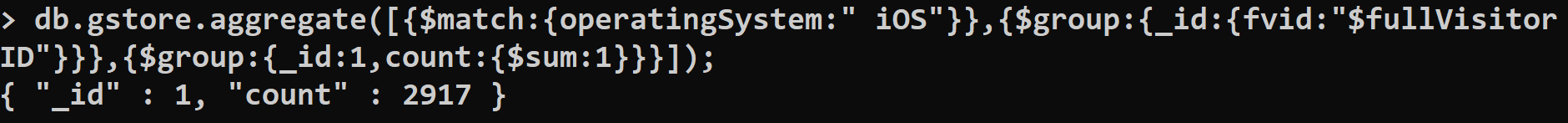
Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

MATCH the “operating System = iOS”.

GROUP by “full Visitor ID” and get the COUNT of the result to know how many users used only iOS devices to visit the store.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$match:{operatingSystem:" iOS"}},{$group:{\_id:{fvid:"$fullVisitorID"}}},{$group:{\_id:1,count:{$sum:1}}}]);



## Query 7

### Question

Which user generated the least amount of hits and when?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

The user with least number of hits is with “Full visit ID: 918000000000000000” and the date is on “2018/04/21”.

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

GROUP by both “full Visitor ID” and “date”.

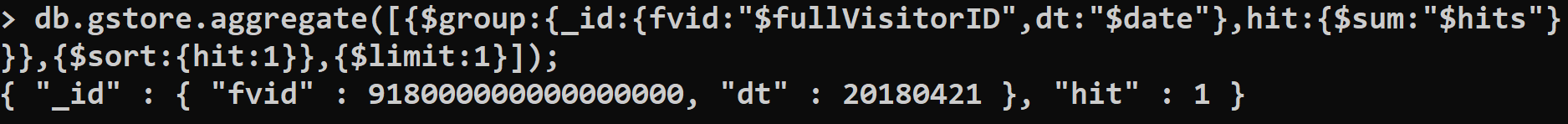
Get the COUNT of the hits.

SORT the COUNT in ASCENDING order.

LIMIT the result to the 1st row to know which user generated the least amount of hits and when.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$group:{\_id:{fvid:"$fullVisitorID",dt:"$date"},hit:{$sum:"$hits"}}},{$sort:{hit:1}},{$limit:1}]);



## Query 8

### Question

Visitors from which country visited the store more than once?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

### Visitors from a total of 208 countries visited the store more than once.

### Rows in result: 208.

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

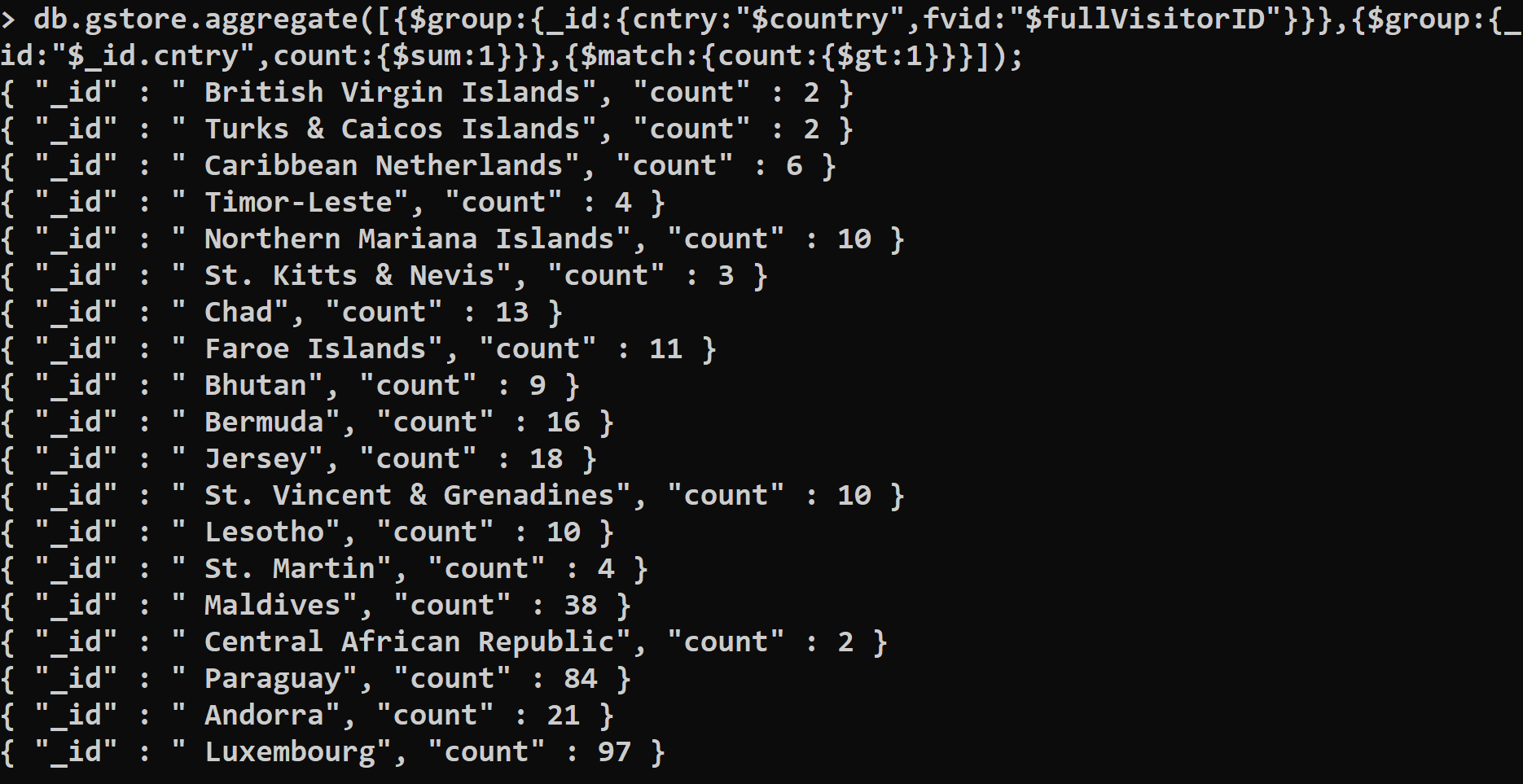
GROUP by the “country” and “full Visitor ID”.

Get the COUNT of visitors from each country.

MATCH the COUNT where COUNT is “GREATER than 1” to know visitors from which country visited the store more than once.

### Screen Shot of MongoDB Query/Code and Results

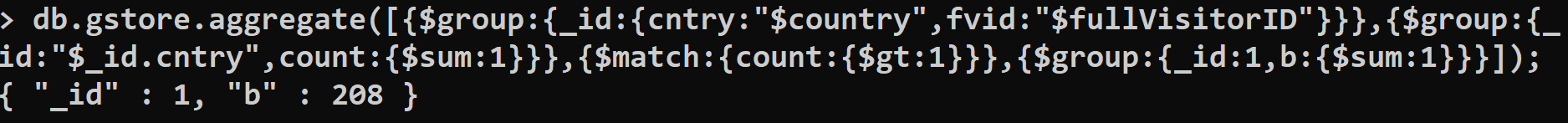
db.gstore.aggregate([{$group:{\_id:{cntry:"$country",fvid:"$fullVisitorID"}}},{$group:{\_id:"$\_id.cntry",count:{$sum:1}}},{$match:{count:{$gt:1}}}]);





**Number of rows in result:**

db.gstore.aggregate([{$group:{\_id:{cntry:"$country",fvid:"$fullVisitorID"}}},{$group:{\_id:"$\_id.cntry",count:{$sum:1}}},{$match:{count:{$gt:1}}},{$group:{\_id:1,b:{$sum:1}}}]);



## Query 1

### Question

Which user had the minimum number of visits and when?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Rows in Result)

The user with minimum number of visits is with “Full visit ID : 70200000000000” and the date is “2017/08/02”.

No. of rows in result: 1

### Translation

Use the collection “gstore” and AGGREGATE all the functions enclosed within the parentheses.

GROUP by “full visitor ID”, SUM the “visits” on date.

SORT the results in ASCENDING order by specifying “visits = 1” and “date =1”.

LIMIT the results to the 1st row to get the user with the minimum no. of visits along with the date.

### Screen Shot of MongoDB Query/Code and Results

db.gstore.aggregate([{$group:{\_id:{fvid:"$fullVisitorID"},vc:{$sum:"$visits"},dt:{$last:"$date"}}},{$sort:{vc:1,dt:1}},{$limit:1}]);

