Declaration: I have viewed the final version of the assignment that is to be submitted and it is my original work.
Signature: Irfan, Andrew, Nicholas, Minjae

IAB206 - Group 60: Assessment Task 3

This report was written for the purpose of familiarizing the key stakeholders of this project, with the dataset provided to us for analysis. To make the final analysis provided in *Task 15,* simpler, this report will include possible trend predictions, as well as possible points of interest.

To start with, the dataset is a set of old tweets from approximately 10 years ago. The aim of this analysis will be to realise any possible findings with this data, as seen in the future tasks provided.

To organise this dataset, separate 'fields' have been provided. These fields each describe a value, with the value being dependant on the field. For example, the field 'created_at' will provide a value that is a string of characters. Some fields are have fields within, called objects. The main field that will be analysed is predicted to be the 'user' object, as this object has many distinct values.

In order to analyse the data intensively, the dataset is run through code called an 'aggregation pipeline', wherein the raw data acts as input, and the 'pipeline' filters the raw data into something that can be analysed efficiently. These 'aggregation pipelines' will be discussed individually in their respective tasks, as well as their value output and scenario.

In the findings seen below, it may be noted that specific trends may appear. For example, a possible trend that may be present in the findings could be a large discrepancy between follower and friend count. These trends will be discussed in detail in the analysis report of *Task 15*.

Tasks 2-14 should be completed using Mongo Shell
Task 2: Describe a scenario and write a query to summarise (summarise would mean finding the mean, median, and other value of interest) a field of your choice from the entire dataset.

Task 3: Describe a scenario and write a query that uses $match operator to match documents against two conditions (e.g., age greater than 20 and year less than 2010) and then uses $project operator to project any four fields.

Task 4: Describe a scenario and write a query that has the following elements:

- Matches the documents against a given condition (e.g. number of employees greater than 500)
- Groups the documents using a logical _id field
- Provides aggregated information for each group (you may use $max, $min, $avg, $sum)

Task 5: Describe a scenario and write a query that skips through some documents in the third stage of the aggregation pipeline.

Task 6: Describe a scenario and write a query that sorts documents in the third stage of the aggregation pipeline.

Task 7: Describe a scenario and write a query that uses $bucket operator and limits results to a certain number of documents in the aggregation pipeline.

Task 8: Describe a scenario and write a query to reshape a document in your dataset such that the names of two fields is displayed as a fieldname of your choice in the output.

Task 9: Describe a scenario and write a query that uses any two of these functions: $concat, $substr, $toLower, $toUpper

Task 10: Describe a scenario and write a query that uses any two of these functions: $add, $divide, $mod, $multiply, $subtract

Task 11: Describe a scenario and write a query that uses $redact, $$descend and $$prune command

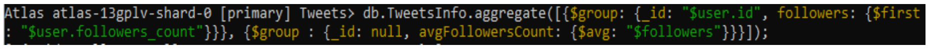
Task 12: Describe a scenario and write a query that uses the $graphlookup operator. Limit your search to 20 documents.

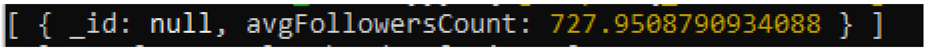
Task 13: Think from the perspective of a data analyst. You want to present some interesting information from the dataset, which may be useful for the key stakeholders. Describe a scenario and write a query using the aggregation framework of MongoDB that outputs this information. The query should not be similar to any queries you have written in Tasks 2-12 but may include any operator used in the prior tasks.

Task 14: Describe a scenario and write a query to create a simple map-reduce function for the dataset provided to you.

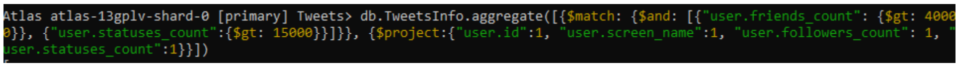
Task no	2

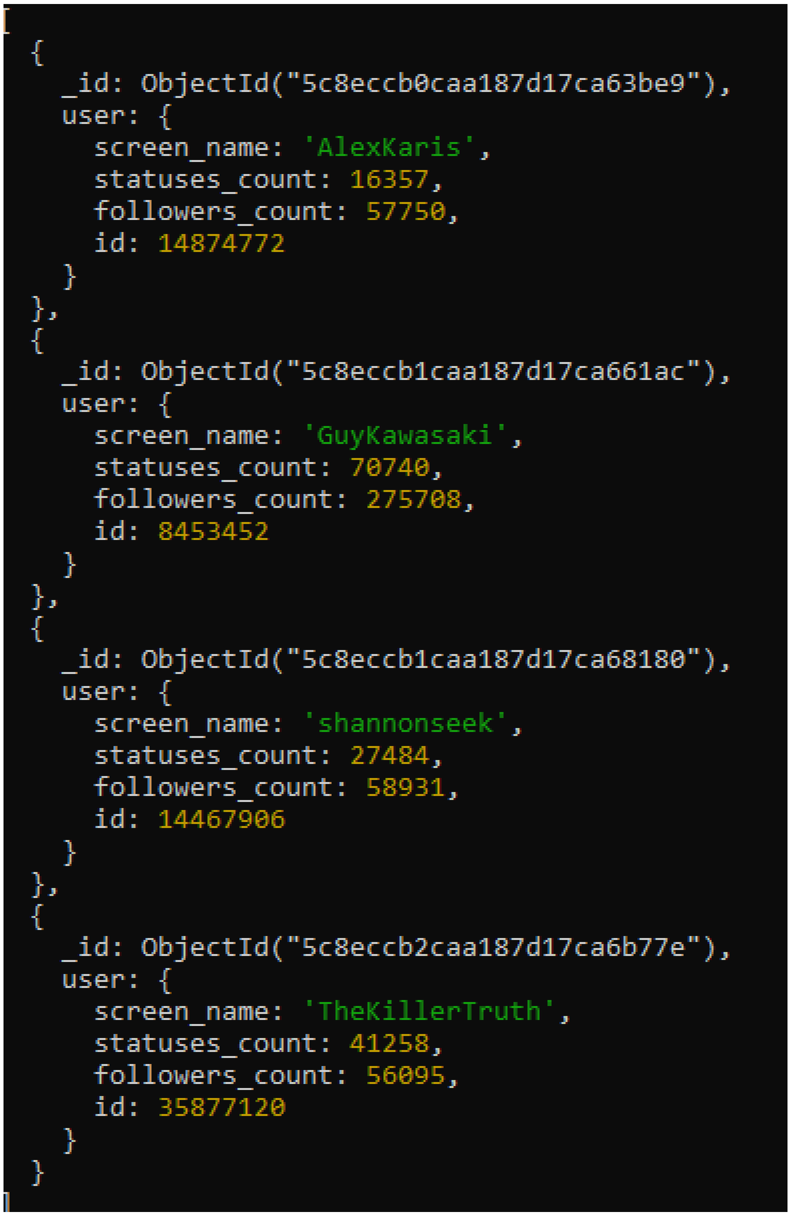
Scenario Description	The the database consists of numerous tweets made by the users. Each tweets has different amount of influence and a major factor that contributes to it is the creator's followers count. Therefore, we want to calculate the average followers count which can be used to estimate influence.
Value Proposition	Stakeholders can compare an account's followers count to this value to determine the amount of influence that the account has and whether it is worth investing.

Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$group: {_id: "$user.id", followers: {$first`
	`: "$user.followers_count"}}}, {$group : {_id: null, avgFollowersCount: {$avg: "$followers"}}}]);`

Output screenshot	 `[ { _id: null, avgFollowersCount: 727.9508790934088 } ]`

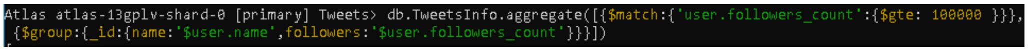
Task no	3
Scenario Description	Stakeholders may be interested in accounts with a lot of social interactions. Thus, we want to get accounts that have many friends and statuses and present the relevant account information to the stakeholders.
Value Proposition	Tweets made by user with numerous social connections can have greater influence. Stakeholder can use the result to easily find the information of such account.

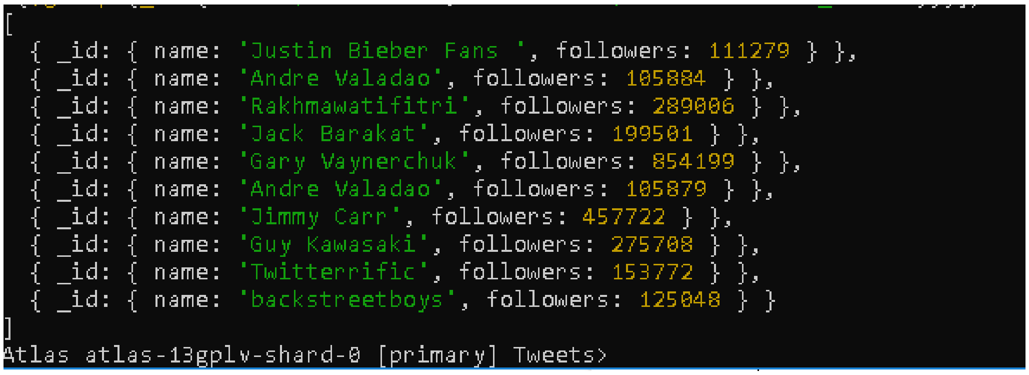
Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$match: {$and: [{"user.friends_count": {$gt: 4000`
	`0}}, {"user.statuses_count":{$gt: 15000}}]}}, {$project:{"user.id":1, "user.screen_name":1, "user.followers_count": 1, "`
	`user.statuses_count":1}}])`

Output screenshot	

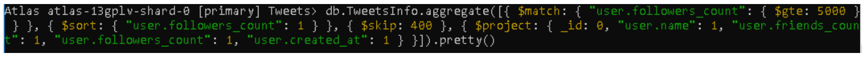
```
[
  {
    _id: ObjectId("5c8eccb0caa187d17ca63be9"),
    user: {
      screen_name: 'AlexKaris',
      statuses_count: 16357,
      followers_count: 57750,
      id: 14874772
    }
  },
  {
    _id: ObjectId("5c8eccb1caa187d17ca661ac"),
    user: {
      screen_name: 'GuyKawasaki',
      statuses_count: 70740,
      followers_count: 275708,
      id: 8453452
    }
  },
  {
    _id: ObjectId("5c8eccb1caa187d17ca68180"),
    user: {
      screen_name: 'shannonseek',
      statuses_count: 27484,
      followers_count: 58931,
      id: 14467906
    }
  },
  {
    _id: ObjectId("5c8eccb2caa187d17ca6b77e"),
    user: {
      screen_name: 'TheKillerTruth',
      statuses_count: 41258,
      followers_count: 56095,
      id: 35877120
    }
  }
]
```

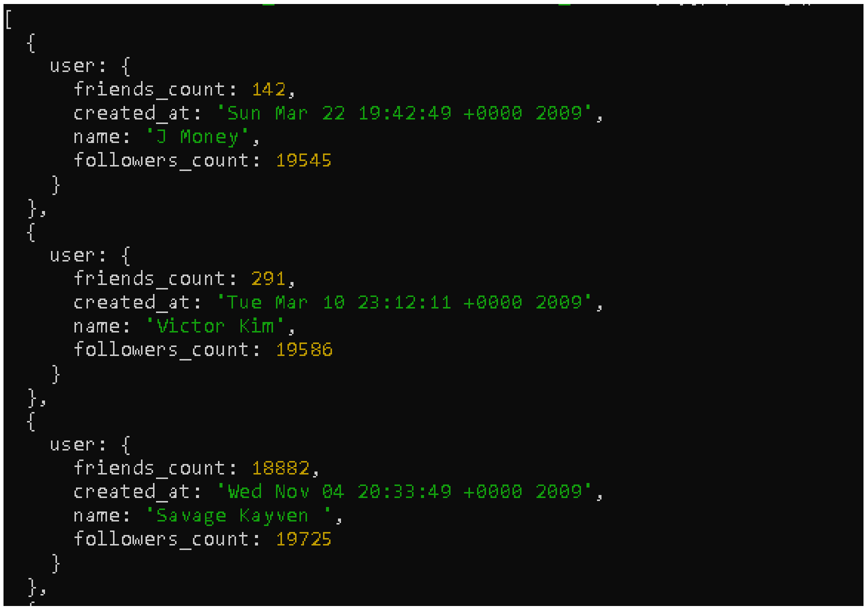
Task no	4
Scenario Description	I wanted to find and group people that have more or equal to 100000 of followers after removing their friends. Display their user IDs .
Value Proposition	Stakeholders can easily find influencers which got huge amount of followers. They can advertise some other products by contact and collaborate with them.

Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$match:{'user.followers_count':{$gte: 100000 }}},`
	` {$group:{_id:{name:'$user.name',followers:'$user.followers_count'}}}])`

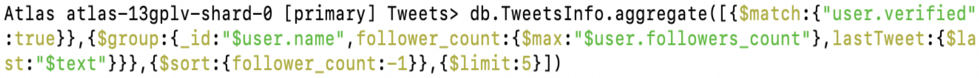
Output screenshot	
	`[`
	`  { _id: { name: 'Justin Bieber Fans ', followers: 111279 } },`
	`  { _id: { name: 'Andre Valadao', followers: 105884 } },`
	`  { _id: { name: 'Rakhmawatifitri', followers: 289006 } },`
	`  { _id: { name: 'Jack Barakat', followers: 199501 } },`
	`  { _id: { name: 'Gary Vaynerchuk', followers: 854199 } },`
	`  { _id: { name: 'Andre Valadao', followers: 105879 } },`
	`  { _id: { name: 'Jimmy Carr', followers: 457722 } },`
	`  { _id: { name: 'Guy Kawasaki', followers: 275708 } },`
	`  { _id: { name: 'Twitterrific', followers: 153772 } },`
	`  { _id: { name: 'backstreetboys', followers: 125048 } }`
	`]`
	`Atlas atlas-13gplv-shard-0 [primary] Tweets>`

Task no	5
Scenario Description	I want to find users that have greater than or equal to 5000 followers, displaying the user's name, their friend count, and their follower count. I will also want to be sorting the list by follower count in descending order, skipping the first 500 tweets in an effort to find more popular users.
Value Proposition	Get users that have 5000 followers or more,

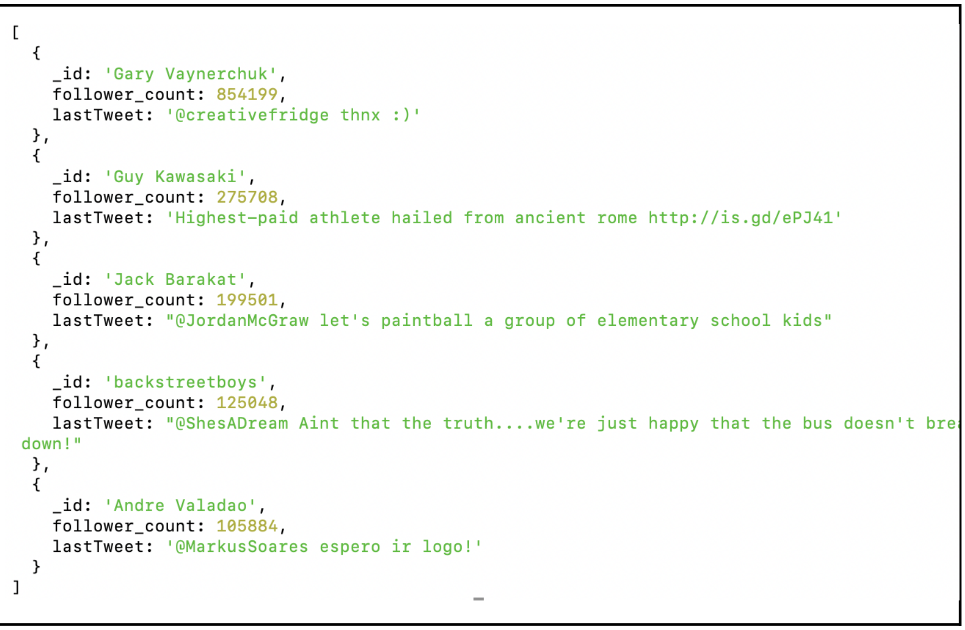
Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{ $match: { "user.followers_count": { $gte: 5000 }`
	` } }, { $sort: { "user.followers_count": 1 } }, { $skip: 400 }, { $project: { _id: 0, "user.name": 1, "user.friends_coun`
	`t": 1, "user.followers_count": 1, "user.created_at": 1 } }]).pretty()`

Output screenshot	

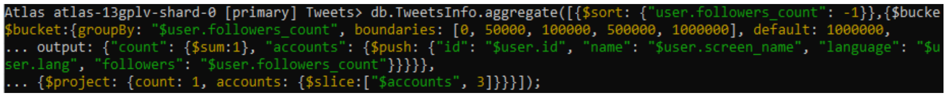
```
[
  {
    user: {
      friends_count: 142,
      created_at: 'Sun Mar 22 19:42:49 +0000 2009',
      name: 'J Money',
      followers_count: 19545
    }
  },
  {
    user: {
      friends_count: 291,
      created_at: 'Tue Mar 10 23:12:11 +0000 2009',
      name: 'Victor Kim',
      followers_count: 19586
    }
  },
  {
    user: {
      friends_count: 18882,
      created_at: 'Wed Nov 04 20:33:49 +0000 2009',
      name: 'Savage Kayven ',
      followers_count: 19725
    }
  },
```

Task no	6
Scenario Description	I would like to find out who the most followed verified users are in the dataset. I can extract each user's name and follower count, and then sort in descending order to find the top 5 users of this criteria.
Value Proposition	This query could be useful, as advertisers might want to contact brand-friendly verified users with large followings to promote their products, or gain insight into the interests of twitter users (i.e why do they follow these people?)
Query screenshot	

```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$match:{"user.verified"
:true}},{$group:{_id:"$user.name",follower_count:{$max:"$user.followers_count"},lastTweet:{$la
st:"$text"}}},{$sort:{follower_count:-1}},{$limit:5}])
```

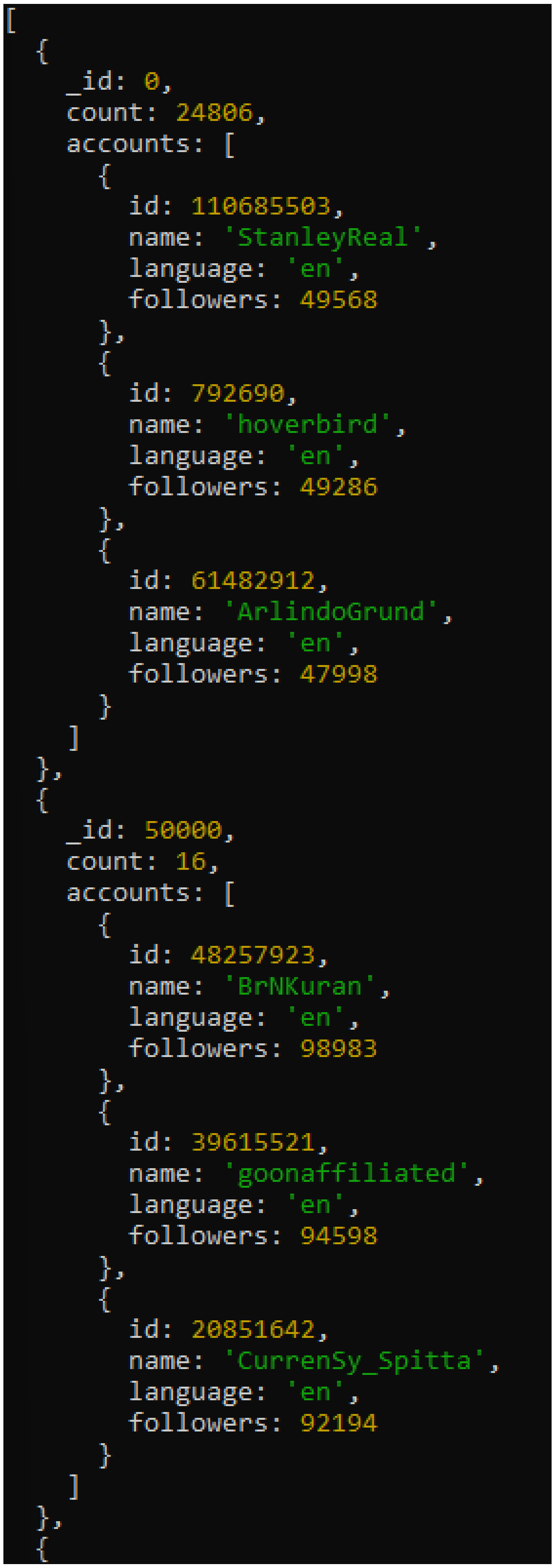
Output screenshot	

```
[
  {
    _id: 'Gary Vaynerchuk',
    follower_count: 854199,
    lastTweet: '@creativefridge thnx :)'
  },
  {
    _id: 'Guy Kawasaki',
    follower_count: 275708,
    lastTweet: 'Highest-paid athlete hailed from ancient rome http://is.gd/ePJ41'
  },
  {
    _id: 'Jack Barakat',
    follower_count: 199501,
    lastTweet: "@JordanMcGraw let's paintball a group of elementary school kids"
  },
  {
    _id: 'backstreetboys',
    follower_count: 125048,
    lastTweet: "@ShesADream Aint that the truth....we're just happy that the bus doesn't bre
 down!"
  },
  {
    _id: 'Andre Valadao',
    follower_count: 105884,
    lastTweet: '@MarkusSoares espero ir logo!'
  }
]
```

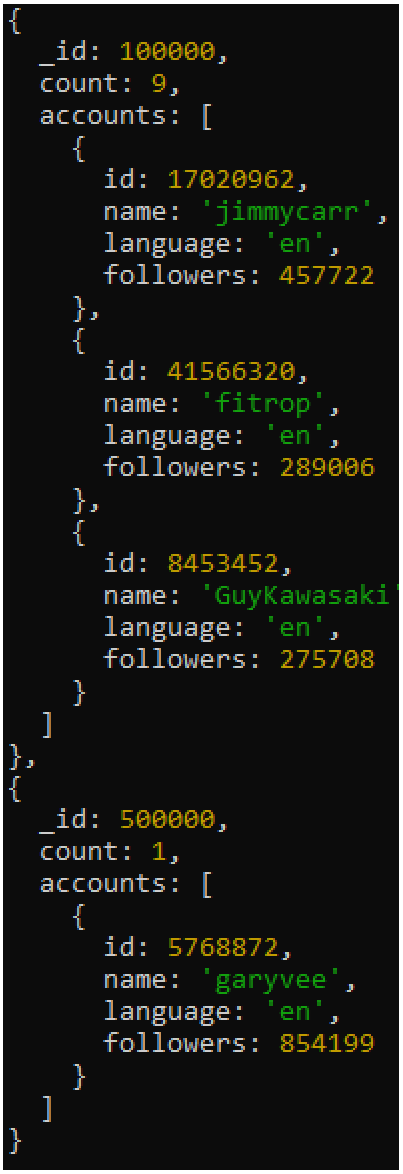
Task no	7
Scenario Description	Stakeholders may be interested in how accounts are distributed according to the followers count. Thus, we need to group the accounts based on the number of followers and present the number of accounts and relevant information of the top 3 most followed accounts in each bucket to the stakeholders.
Value Proposition	Stakeholder can use the data to find out the distribution of accounts and information of 3 accounts with the most followers within each category which can be used if they are looking to target different community.
Query screenshot	

```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$sort: {"user.followers_count": -1}},{$bucke
$bucket:{groupBy: "$user.followers_count", boundaries: [0, 50000, 100000, 500000, 1000000], default: 1000000,
... output: {"count": {$sum:1}, "accounts": {$push: {"id": "$user.id", "name": "$user.screen_name", "language": "$u
ser.lang", "followers": "$user.followers_count"}}}}},
... {$project: {count: 1, accounts: {$slice:["$accounts", 3]}}}]);
```

Output screenshot



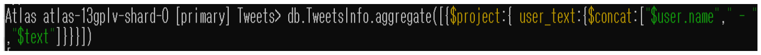
```
[
  {
    _id: 0,
    count: 24806,
    accounts: [
      {
        id: 110685503,
        name: 'StanleyReal',
        language: 'en',
        followers: 49568
      },
      {
        id: 792690,
        name: 'hoverbird',
        language: 'en',
        followers: 49286
      },
      {
        id: 61482912,
        name: 'ArlindoGrund',
        language: 'en',
        followers: 47998
      }
    ]
  },
  {
    _id: 50000,
    count: 16,
    accounts: [
      {
        id: 48257923,
        name: 'BrNKuran',
        language: 'en',
        followers: 98983
      },
      {
        id: 39615521,
        name: 'goonaffiliated',
        language: 'en',
        followers: 94598
      },
      {
        id: 20851642,
        name: 'CurrenSy_Spitta',
        language: 'en',
        followers: 92194
      }
    ]
  },
  {
```

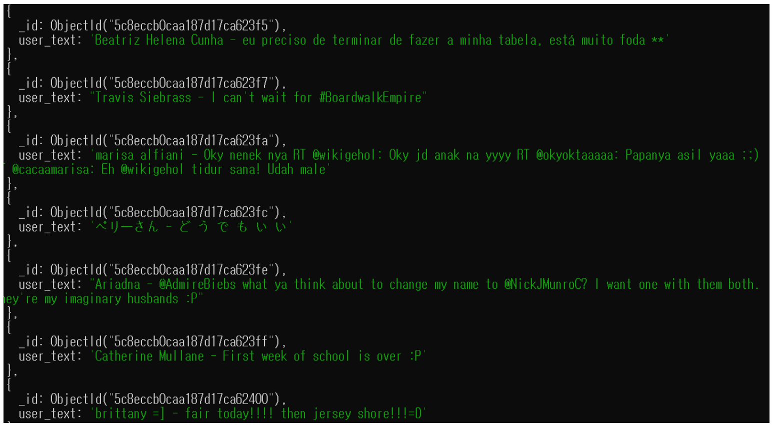
	

```
{
  _id: 100000,
  count: 9,
  accounts: [
    {
      id: 17020962,
      name: 'jimmycarr',
      language: 'en',
      followers: 457722
    },
    {
      id: 41566320,
      name: 'fitrop',
      language: 'en',
      followers: 289006
    },
    {
      id: 8453452,
      name: 'GuyKawasaki'
      language: 'en',
      followers: 275708
    }
  ]
},
{
  _id: 500000,
  count: 1,
  accounts: [
    {
      id: 5768872,
      name: 'garyvee',
      language: 'en',
      followers: 854199
    }
  ]
}
```

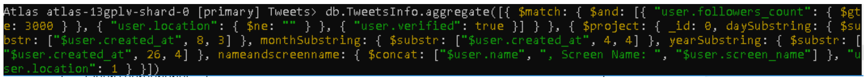
Task no	8
Scenario Description	I wanted to display the user.name field and text field at once by the field name of user_text. So I decided to combine them by having “ - “ operator in the middle to distinguish them.
Value Proposition	In the database file, text and the user.name was too far away so that we have to scroll down a lot in order to find who wrote

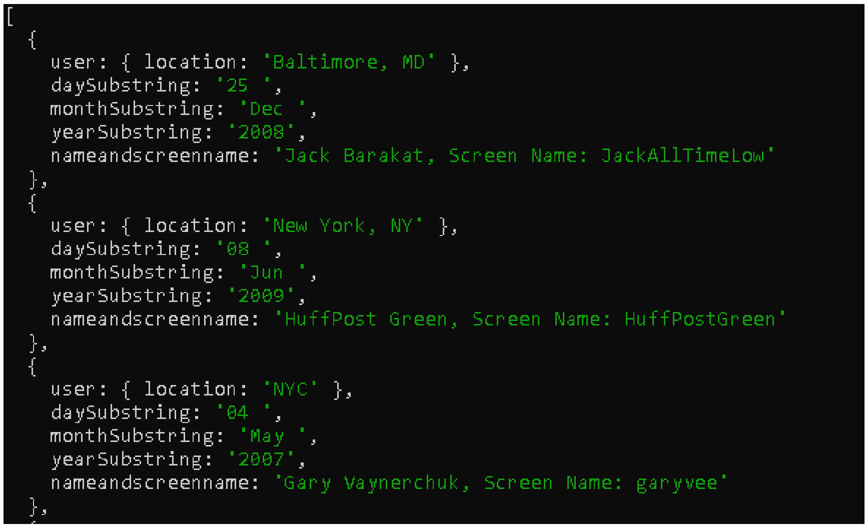
	the text. But combine them so that we don’t need to waste our time to find them.

Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$project:{ user_text:{$concat:["$user.name"," - "`
	`,"$text"]}}}])`

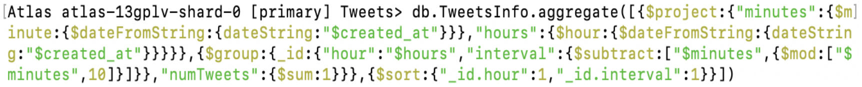
Output screenshot	
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623f5"),`
	`  user_text: 'Beatriz Helena Cunha - eu preciso de terminar de fazer a minha tabela, está muito foda **'`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623f7"),`
	`  user_text: "Travis Siebrass - I can't wait for #BoardwalkEmpire"`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623fa"),`
	`  user_text: 'marisa alfiani - Oky nenek nya RT @wikigehol: Oky jd anak na yyyy RT @okyoktaaaaa: Papanya asil yaaa ;;)`
	` @cacaamarisa: Eh @wikigehol tidur sana! Udah male'`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623fc"),`
	`  user_text: 'ペリーさん - ど う で も い い'`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623fe"),`
	`  user_text: "Ariadna - @AdmireBiebs what ya think about to change my name to @NickJMunroC? I want one with them both.`
	`ey're my imaginary husbands :P"`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca623ff"),`
	`  user_text: 'Catherine Mullane - First week of school is over :P'`
	`},`
	`{`
	`  _id: ObjectId("5c8eccb0caa187d17ca62400"),`
	`  user_text: 'brittany =] - fair today!!!! then jersey shore!!!=D'`

Task no	9
Scenario Description	I want to find a user that has their account verified, has at least 3000 followers, and has a location that is not an empty string. I also want to make a field, "nameandscreenname" that concatenantes the users name and their respective screen name, with a ', Screen Name: ' delimiter phrase. Next, I want to create substrings of the date the account was created, by their respective DateTime fields (e.g. days = Fri, month = Nov, year = 2008). Finally, I want to display the field, "nameandscreenname", the day, month, and the year it was created, and the location.
Value Proposition	Get users that have their account verified, have 3000 followers or more, and have a location that is not an empty string.

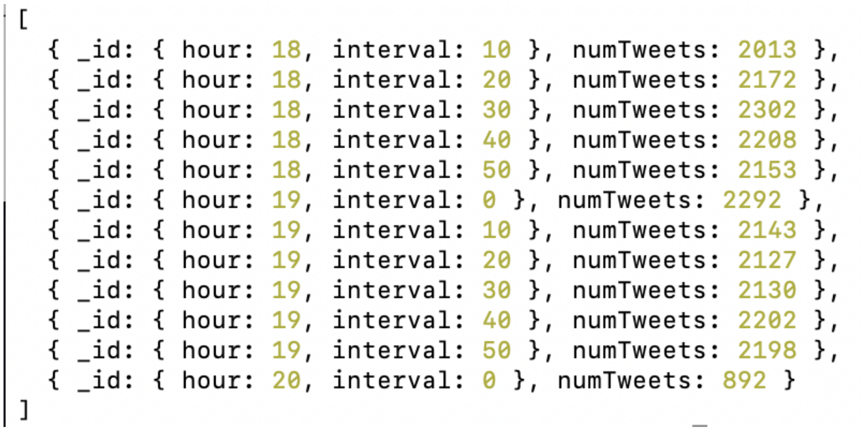
Query screenshot	
	`Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{ $match: { $and: [{ "user.followers_count": { $gt`
	`e: 3000 } }, { "user.location": { $ne: "" } }, { "user.verified": true }] } }, { $project: { _id: 0, daySubstring: { $su`
	`bstr: ["$user.created_at", 8, 3] }, monthSubstring: { $substr: ["$user.created_at", 4, 4] }, yearSubstring: { $substr: [`
	`"$user.created_at", 26, 4] }, nameandscreenname: { $concat: ["$user.name", ", Screen Name: ", "$user.screen_name"] }, "u`
	`ser.location": 1 } }])`

Output screenshot	

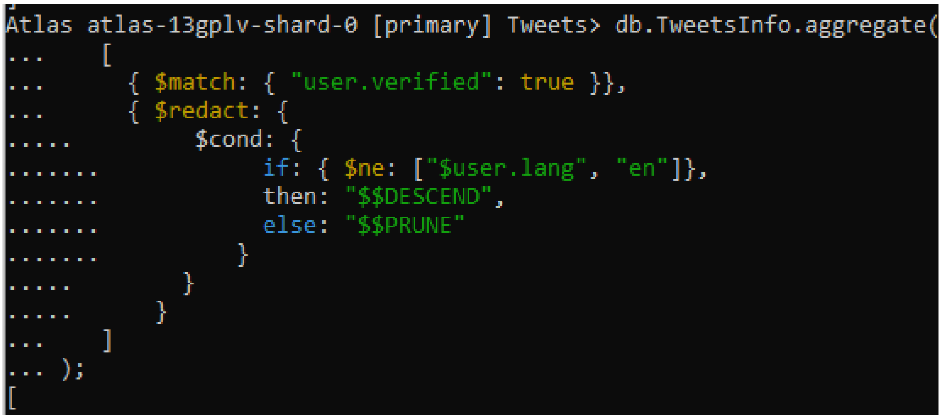
```
[
  {
    user: { location: 'Baltimore, MD' },
    daySubstring: '25 ',
    monthSubstring: 'Dec ',
    yearSubstring: '2008',
    nameandscreenname: 'Jack Barakat, Screen Name: JackAllTimeLow'
  },
  {
    user: { location: 'New York, NY' },
    daySubstring: '08 ',
    monthSubstring: 'Jun ',
    yearSubstring: '2009',
    nameandscreenname: 'HuffPost Green, Screen Name: HuffPostGreen'
  },
  {
    user: { location: 'NYC' },
    daySubstring: '04 ',
    monthSubstring: 'May ',
    yearSubstring: '2007',
    nameandscreenname: 'Gary Vaynerchuk, Screen Name: garyvee'
  },
```

Task no	10
Scenario Description	The database consists of tweets made between 18:11 and 20:03 on 02/09/2010. Using this small sample of tweets, we can estimate how many tweets are made within any arbitrary time interval. For example, we can divide the window up into 10 minute intervals and record how many tweets were made in each and use this to model a sampling distribution of the number of tweets.
Value Proposition	It is possible to apply this to larger datasets, or even a stream of data, and use this information to predict or gain insight into Twitter's userbase's behaviors; this could prove useful for stakeholders and advertisers. (e.g. Promoting certain ads during peak periods)
Query screenshot	

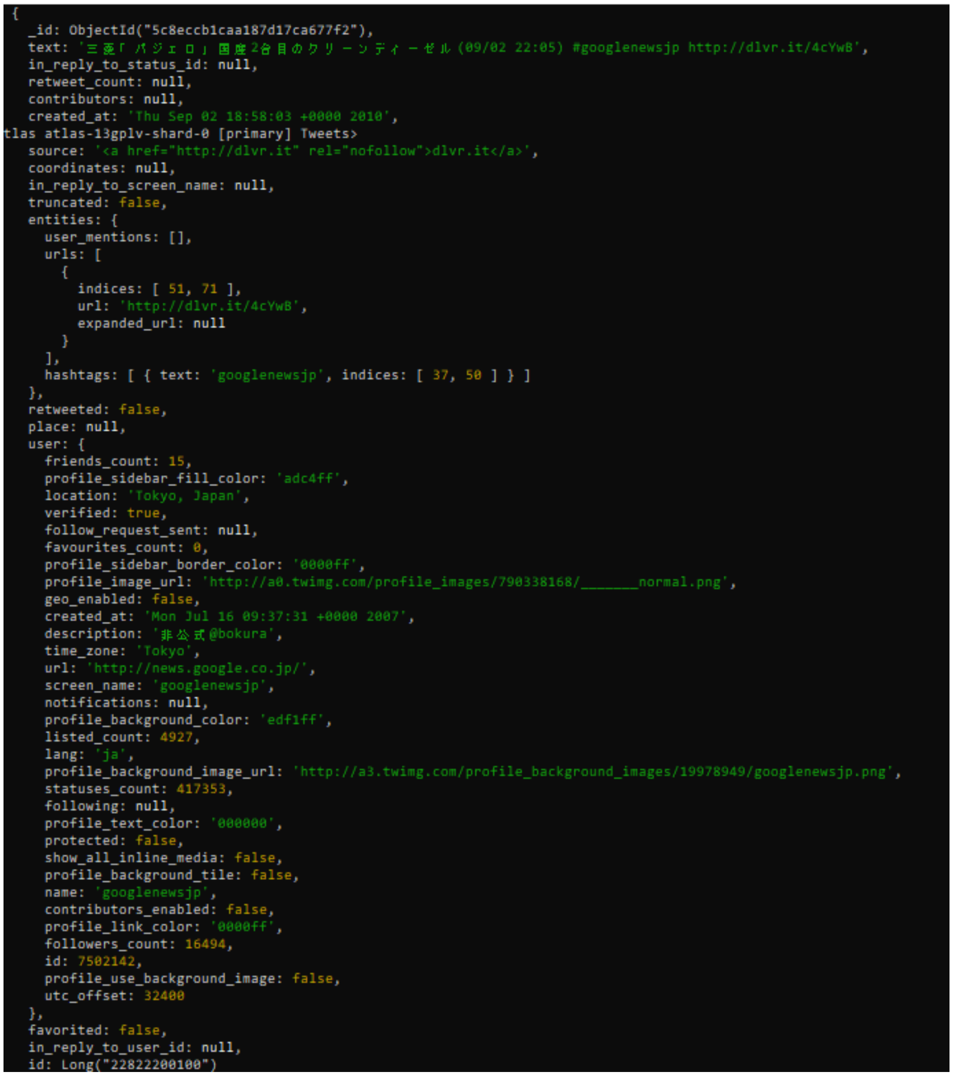
```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{$project:{"minutes":{$m
inute:{$dateFromString:{dateString:"$created_at"}}},"hours":{$hour:{$dateFromString:{dateStrin
g:"$created_at"}}}}},{$group:{_id:{"hour":"$hours","interval":{$subtract:["$minutes",{$mod:["$
minutes",10]}]}},"numTweets":{$sum:1}}},{$sort:{"_id.hour":1,"_id.interval":1}}])
```

Output screenshot	

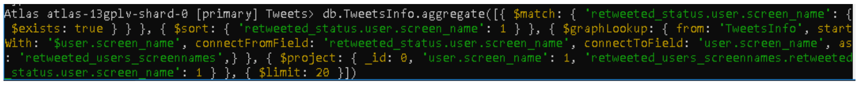
```
[
  { _id: { hour: 18, interval: 10 }, numTweets: 2013 },
  { _id: { hour: 18, interval: 20 }, numTweets: 2172 },
  { _id: { hour: 18, interval: 30 }, numTweets: 2302 },
  { _id: { hour: 18, interval: 40 }, numTweets: 2208 },
  { _id: { hour: 18, interval: 50 }, numTweets: 2153 },
  { _id: { hour: 19, interval: 0 }, numTweets: 2292 },
  { _id: { hour: 19, interval: 10 }, numTweets: 2143 },
  { _id: { hour: 19, interval: 20 }, numTweets: 2127 },
  { _id: { hour: 19, interval: 30 }, numTweets: 2130 },
  { _id: { hour: 19, interval: 40 }, numTweets: 2202 },
  { _id: { hour: 19, interval: 50 }, numTweets: 2198 },
  { _id: { hour: 20, interval: 0 }, numTweets: 892 }
]
```

Task no	11
Scenario Description	Stakeholders may be interested to expand their audience to target non-English speakers. Therefore, we need to get accounts that are verified and use language that is not english.
Value Proposition	Stakeholders can use the data to get information of verified accounts with non-English speaking community and collaborate with them to expand their market.
Query screenshot	

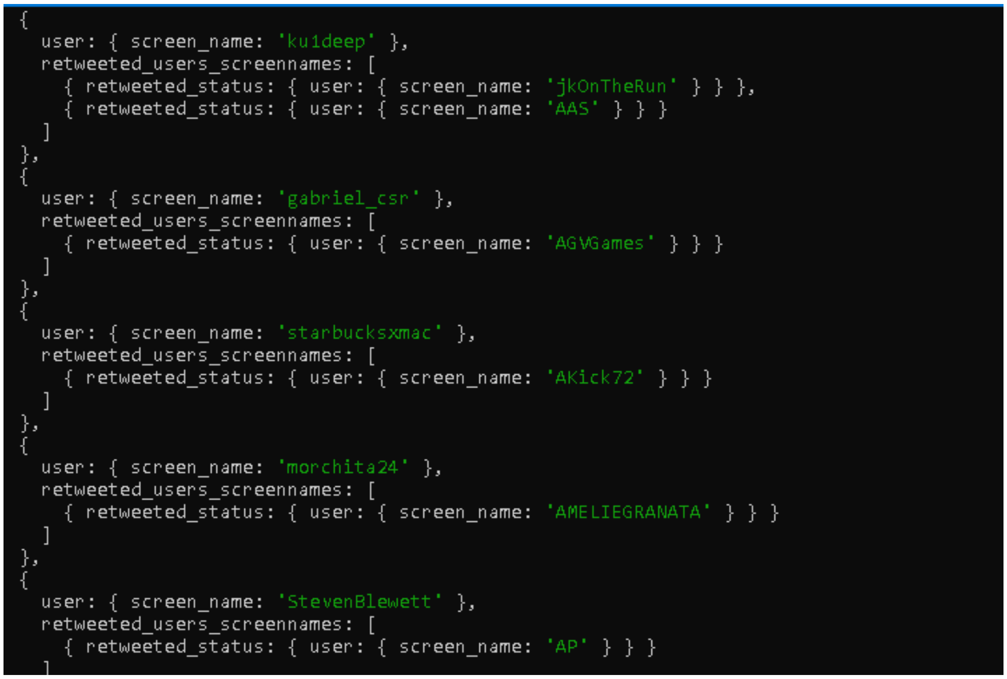
```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate(
...     [
...       { $match: { "user.verified": true }},
...       { $redact: {
.....         $cond: {
.......         if: { $ne: ["$user.lang", "en"]},
.......         then: "$$DESCEND",
.......         else: "$$PRUNE"
.......       }
.....       }
.....     }
...     ]
... );
[
```

Output screenshot	

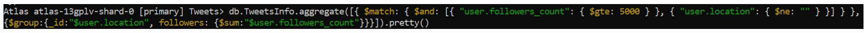
```
{
  _id: ObjectId("5c8eccb1caa187d17ca677f2"),
  text: '三菱「パジェロ」国産2台目のクリーンディーゼル (09/02 22:05) #googlenewsjp http://dlvr.it/4cYwB',
  in_reply_to_status_id: null,
  retweet_count: null,
  contributors: null,
  created_at: 'Thu Sep 02 18:58:03 +0000 2010',
tlas atlas-13gplv-shard-0 [primary] Tweets>
  source: '<a href="http://dlvr.it" rel="nofollow">dlvr.it</a>',
  coordinates: null,
  in_reply_to_screen_name: null,
  truncated: false,
  entities: {
    user_mentions: [],
    urls: [
      {
        indices: [ 51, 71 ],
        url: 'http://dlvr.it/4cYwB',
        expanded_url: null
      }
    ],
    hashtags: [ { text: 'googlenewsjp', indices: [ 37, 50 ] } ]
  },
  retweeted: false,
  place: null,
  user: {
    friends_count: 15,
    profile_sidebar_fill_color: 'adc4ff',
    location: 'Tokyo, Japan',
    verified: true,
    follow_request_sent: null,
    favourites_count: 0,
    profile_sidebar_border_color: '0000ff',
    profile_image_url: 'http://a0.twimg.com/profile_images/790338168/_______normal.png',
    geo_enabled: false,
    created_at: 'Mon Jul 16 09:37:31 +0000 2007',
    description: '非公式@bokura',
    time_zone: 'Tokyo',
    url: 'http://news.google.co.jp/',
    screen_name: 'googlenewsjp',
    notifications: null,
    profile_background_color: 'edf1ff',
    listed_count: 4927,
    lang: 'ja',
    profile_background_image_url: 'http://a3.twimg.com/profile_background_images/19978949/googlenewsjp.png',
    statuses_count: 417353,
    following: null,
    profile_text_color: '000000',
    protected: false,
    show_all_inline_media: false,
    profile_background_tile: false,
    name: 'googlenewsjp',
    contributors_enabled: false,
    profile_link_color: '0000ff',
    followers_count: 16494,
    id: 7502142,
    profile_use_background_image: false,
    utc_offset: 32400
  },
  favorited: false,
  in_reply_to_user_id: null,
  id: Long("22822200100")
```

Task no	12
Scenario Description	I wanted to see users that have retweeted on a tweet, their screen name, and try to match it to an existing screenname in the collection.
Value Proposition	Find and recursively search for screennames that match the retweeted_status.user.screen_name to user.screen_name. Place these users in an array called "retweeted_users_screennames".
Query screenshot	

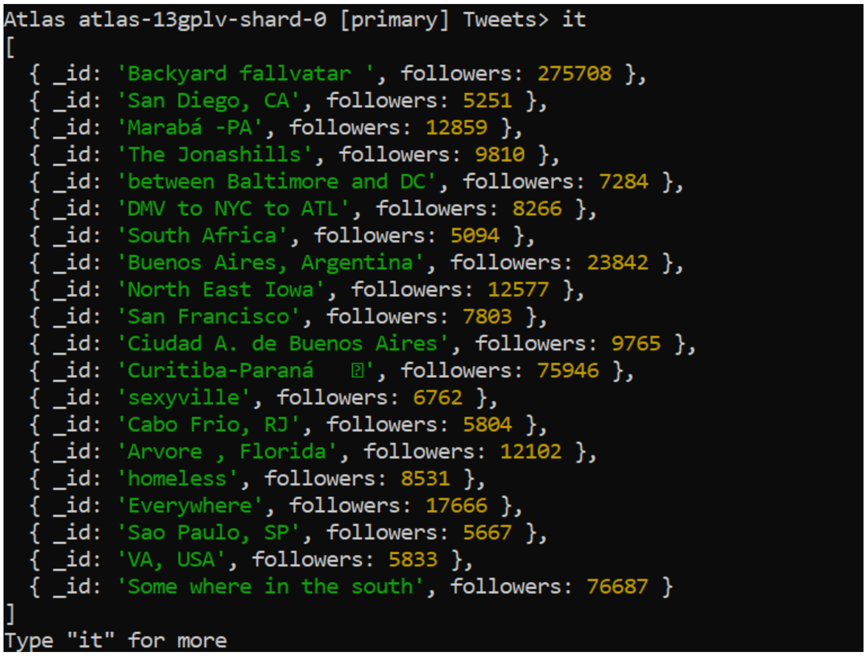
```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{ $match: { 'retweeted_status.user.screen_name': {
 $exists: true } } }, { $sort: { 'retweeted_status.user.screen_name': 1 } }, { $graphLookup: { from: 'TweetsInfo', start
With: '$user.screen_name', connectFromField: 'retweeted_status.user.screen_name', connectToField: 'user.screen_name', as
: 'retweeted_users_screennames',} }, { $project: { _id: 0, 'user.screen_name': 1, 'retweeted_users_screennames.retweeted
_status.user.screen_name': 1 } }, { $limit: 20 }])
```

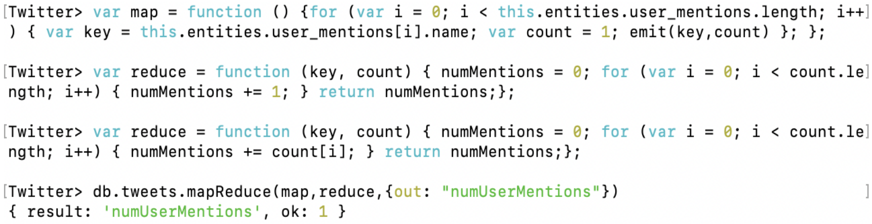
Output screenshot	

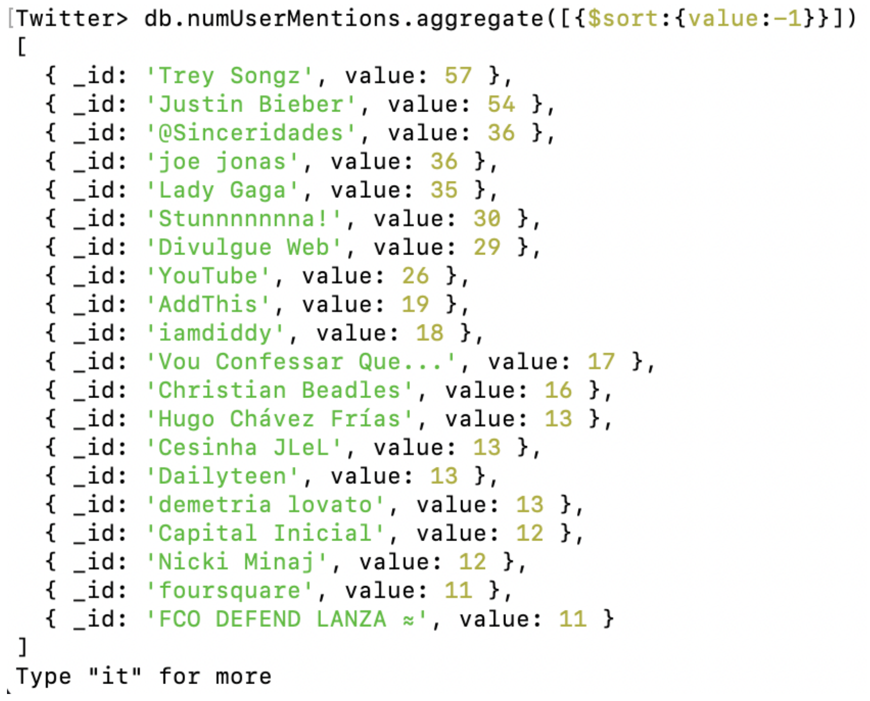
```
{
  user: { screen_name: 'ku1deep' },
  retweeted_users_screennames: [
    { retweeted_status: { user: { screen_name: 'jkOnTheRun' } } },
    { retweeted_status: { user: { screen_name: 'AAS' } } }
  ]
},
{
  user: { screen_name: 'gabriel_csr' },
  retweeted_users_screennames: [
    { retweeted_status: { user: { screen_name: 'AGVGames' } } }
  ]
},
{
  user: { screen_name: 'starbucksxmac' },
  retweeted_users_screennames: [
    { retweeted_status: { user: { screen_name: 'AKick72' } } }
  ]
},
{
  user: { screen_name: 'morchita24' },
  retweeted_users_screennames: [
    { retweeted_status: { user: { screen_name: 'AMELIEGRANATA' } } }
  ]
},
{
  user: { screen_name: 'StevenBlewett' },
  retweeted_users_screennames: [
    { retweeted_status: { user: { screen_name: 'AP' } } }
  ]
```

Task no	13
Scenario Description	As a data analyst, I want to view data from users that have a large following, or friend count, as well as their location, which cannot be null or empty. The purpose of this view would be to identify which countries' citizens interact with Twitter, and the amount they interact with it. For example, if there are more users from USA with a higher following than users from Japan, it can be safe to assume that the citizens residing in USA would utilize Twitter more frequently than Japan. This may prove useful to stakeholders, looking for possible target audiences.
Value Proposition	The followers, grouped by location.
Query screenshot	

```
Atlas atlas-13gplv-shard-0 [primary] Tweets> db.TweetsInfo.aggregate([{ $match: { $and: [{ "user.followers_count": { $gte: 5000 } }, { "user.location": { $ne: "" } }] } },
{$group:{_id:"$user.location", followers: {$sum:"$user.followers_count"}}}]).pretty()
```

Output screenshot	
	Atlas atlas-13gplv-shard-0 [primary] Tweets> it
	[
	  { _id: 'Backyard fallvatar ', followers: 275708 },
	  { _id: 'San Diego, CA', followers: 5251 },
	  { _id: 'Marabá -PA', followers: 12859 },
	  { _id: 'The Jonashills', followers: 9810 },
	  { _id: 'between Baltimore and DC', followers: 7284 },
	  { _id: 'DMV to NYC to ATL', followers: 8266 },
	  { _id: 'South Africa', followers: 5094 },
	  { _id: 'Buenos Aires, Argentina', followers: 23842 },
	  { _id: 'North East Iowa', followers: 12577 },
	  { _id: 'San Francisco', followers: 7803 },
	  { _id: 'Ciudad A. de Buenos Aires', followers: 9765 },
	  { _id: 'Curitiba-Paraná ⍰', followers: 75946 },
	  { _id: 'sexyville', followers: 6762 },
	  { _id: 'Cabo Frio, RJ', followers: 5804 },
	  { _id: 'Arvore , Florida', followers: 12102 },
	  { _id: 'homeless', followers: 8531 },
	  { _id: 'Everywhere', followers: 17666 },
	  { _id: 'Sao Paulo, SP', followers: 5667 },
	  { _id: 'VA, USA', followers: 5833 },
	  { _id: 'Some where in the south', followers: 76687 }
	]
	Type "it" for more

Task no	14
Scenario Description	It would be useful to know how many times users have been previously mentioned by others. We can use a map-reduce function to do this. Then, using the new summarized data, we are able to provide more insight using aggregation - for example, who are the most mentioned users within this sample?
Value Proposition	This query would be useful to businesses or marketers, as they would be able to track how well the engagement and activity of their products with the twitter user base is doing.
Query screenshot	 [Twitter> var map = function () {for (var i = 0; i < this.entities.user_mentions.length; i++]) { var key = this.entities.user_mentions[i].name; var count = 1; emit(key,count) }; }; [Twitter> var reduce = function (key, count) { numMentions = 0; for (var i = 0; i < count.le] ngth; i++) { numMentions += 1; } return numMentions;}; [Twitter> var reduce = function (key, count) { numMentions = 0; for (var i = 0; i < count.le] ngth; i++) { numMentions += count[i]; } return numMentions;}; [Twitter> db.tweets.mapReduce(map,reduce,{out: "numUserMentions"})] { result: 'numUserMentions', ok: 1 }

Output screenshot	

```
[Twitter> db.numUserMentions.aggregate([{$sort:{value:-1}}])
[
  { _id: 'Trey Songz', value: 57 },
  { _id: 'Justin Bieber', value: 54 },
  { _id: '@Sinceridades', value: 36 },
  { _id: 'joe jonas', value: 36 },
  { _id: 'Lady Gaga', value: 35 },
  { _id: 'Stunnnnnnna!', value: 30 },
  { _id: 'Divulgue Web', value: 29 },
  { _id: 'YouTube', value: 26 },
  { _id: 'AddThis', value: 19 },
  { _id: 'iamdiddy', value: 18 },
  { _id: 'Vou Confessar Que...', value: 17 },
  { _id: 'Christian Beadles', value: 16 },
  { _id: 'Hugo Chávez Frías', value: 13 },
  { _id: 'Cesinha JLeL', value: 13 },
  { _id: 'Dailyteen', value: 13 },
  { _id: 'demetria lovato', value: 13 },
  { _id: 'Capital Inicial', value: 12 },
  { _id: 'Nicki Minaj', value: 12 },
  { _id: 'foursquare', value: 11 },
  { _id: 'FCO DEFEND LANZA ≈', value: 11 }
]
Type "it" for more
```

Task 15*: Write a short report summarising your findings of Tasks 2-14. Based on the summary, provide at least 5 recommendations for consideration to the key stakeholders. Your findings and recommendations should be understandable by the stakeholders.

Nicholas:
After evaluation of the "tweets" dataset, it was discovered that there were particular findings in the database that may prove useful to the key stakeholders of this project. Specifically, there can be five recommendations for consideration for this dataset.

In summary, the dataset consisted of many fields and values for the users, but not actually the tweets itself. As shown in the findings, a majority of documents did not relate at all to the tweet, but rather the user who tweeted the tweet. As a result, numerical data on tweets and the like, such as retweets, are impossible to analyse. However, user data is documented nicely, such as followers, friends, the location they reside in, and the language the user speaks.

The first recommendation is to realise that some fields in the dataset, such as "retweet_count" and "retweeted" , were null across all documents. This made it difficult to analyse the tweet itself, as there were limited fields that possessed values, and should be taken into consideration.

The second recommendation is to realise that some values in the fields of object, "user", such as "location", and "lang", are not mapped to a list of real-life locations, but rather something the user enters in themselves. This was made apparent in *Task 13,* wherein a large portion of values in the "location" field, were fictitious.

The third recommendation is to notice the date range of which this dataset was produced. As shown in *Task 5,* the dates displayed in the console prompt are placed approximately within the late 2000s and early 2010s. This may impact the key stakeholders that wish to market to users on twitter, based off of these tweets, as the time difference is too great to effectively analyse.

The fourth recommendation is to notice the difference between followers and friends in *Task 5.* As observed, two out of the three displayed results show a large ratio difference between friends and followers, with the friend count being substantially lower than the follower count. There are exceptions and outliers to this difference, however, as evidenced with the latter third result, as well as *Task 3,* showcasing both large follower counts and friends counts.

The final recommendation advised is to gravitate towards the results found in *Task 9* and *Task 13.* As discussed, the desired outcome was for both was to find a link between location and follower count. It was discovered that users from USA outperformed many other countries in terms of follower count.

Irfan:

In summary, the database consists of numerous tweets made by users from around the world. A tweet can reference another users and task 14 shows some of the most frequently mentioned names. Task 5 shows that 2000 new tweets are being created every 5 seconds which implies that the database only covers a small fraction of the total number of tweets, but we can use the available data to predict possible trends. In the database, each user has account details which contains useful information such as the followers count, friends count, language and country. This information can be used to find the distribution of accounts according to their followers count. Task 7 shows that accounts are unevenly distributed where majority of users have lesser than 50000 followers and this is supported by the finding of task 2 which shows that the average followers count is 7000. Accounts with high followers count are typically verified as shown in task 6 and tasks 11 and 13 show that the majority of the most followed accounts are from America and they can speak english. In addition, we use the user's friends counts such as in task 3, to get account information of users with many social connections. The rest of the findings help to efficiently show relevant information that is beneficial to the stakeholders. Therefore, we would like to propose 5 recommendations to the key stakeholders using the findings.

Firstly, we recommend the stakeholders to consider the rate at which new tweets are being created. The finding of task 5 shows that every 5 seconds, 2000 new tweets are posted which implies that the platform has many active users and the business is successful.

Secondly, stakeholders should consider followers count as accounts with higher followers count generally have more influence. Ths means that their tweets are able to reach larger group of audience. The result from tasks 7 and 2 show that only a small percentage of users has more than 100000 followers, which means that it is difficult to get followers. This is beneficial for advertisers since they can use the information to appropriately pay users to promote their products based on their number of followers.

Another number that stakeholders should consider is the user's friends count. Unlike followers count, friends count represents the number of accounts that the user follows. User usually follows another user due to matching interests. As shown in the result of task 3, users with high friends count typically have high online presence, leading to more social connections and greater influence. Therefore, it useful for advertisers looking to target a specific community.

Additionally, stakeholders should consider the user's language. The platform consists of users from all around the world and they speak different languages. The finding from task 11 shows that there are accounts with high followers count that do not use english as their language, which means that the platform is also popular for non-English speakers. Stakeholders can collaborate with those accounts to help expand their market to non-English speaking community.

Lastly, stakeholders should consider the number of times a user is mentioned in another tweet. The more tweets mentioning a specific user, the higher the popularity of the mentioned user. A tweet usually mentions another user due to an event that occurs involving the user. With the information, we can determine whether it is negative or positive. If it is positive, stakeholders can take the opportunity to benefit from the user.

Andrew:

Our team was given a dataset composed of all tweets between 18:11:23 and 20:03:53 on 02/09/2010 to evaluate, with the aim of presenting any insights or findings that are of use to the project's stakeholders. The findings from the queries presented in this report provide significant insight into the trends and influence of twitters and how the stakeholders could expand their brand by utilizing the influence of social media. Specifically, our findings could be used to make recommendations on who to endorse or sponsor, where they could best tap into foreign audiences, and when to promote their product.

From the dataset, we found examples such as Justin Bieber's and Youtube's twitter accounts. However, we also found that the number of users with a large number of followers seemed to fall off as the follower count increased – Within the dataset, we observed that there were 24806 users with 0-50,000 followers, 9 users with 50,000-100,000 followers, and only one user with 100,000-500,00 followers.
As such, I would be able to recommend that Twitter is definitely a viable platform to begin promoting the stakeholder's company on.

We firstly assumed that users with a larger following base would have a larger spread of influence rather than smaller accounts. From our findings, it seems that this is correct, as these users tended to have a significant number of interactions with other users via having more user mentions and number of tweets. Additionally, it was found that users with more followers tended to tweet more. Users will follow others if they find their account interesting or entertaining, which would be beneficial to a user's influence on others, as Twitter users would see more tweets from the person they have an interest in. Due to this, I would be able to recommend that the stakeholders should consider endorsing users with a larger following. However, they should also seriously consider the number of mentions a user receives, how often the user tweets, and how many friends they have, as these factors definitely impact a user's influence on Twitter.

However, the number of friends a user has, has little to no correlation with the number of followers, meaning that two followers with the same amount of followers might not have the same amount of influence. The user "J Money" had 19,545 followers at the time, but only 142 friends, whereas the user "Savage Kayven" had 19,725 followers and 18,882 friends. From this, I can recommend that the stakeholders should not arbitrarily choose a user with a large number of followers, but should also take time to research more about the individual's Twitter habits.

The stakeholders might also want to begin advertising their brand internationally. From our findings, more prominent western countries such as the United States obviously seemed to contain users with larger followings; for example, the combined follower count of users from Arvore, Florida was over 12,000 and close to 8,000 in San Francisco. However, other countries from South America seemed to have an even larger presence, where users from areas such as Buenos Aires, Argentina had a combined follower count of 23,842. This finding implies that the stakeholders should definitely consider expanding their area of advertising into countries such as Argentina and Brazil.

Finally, our query from task 10 broke down the number of all tweets sent during the time the dataset was collected and grouped the number of tweets into 12 intervals of 10 minutes. The number of tweets sent seemed to fluctuate around 2000-2300 tweets every 10 minutes. While no evident trend could be observed, this was due to a small sample of 12 intervals. However, this seems to be a promising avenue of research, and I recommend the stakeholders to apply this query to larger datasets, or even a data stream of tweets.

Minjae Lee:

We received a file called Tweets to do these tasks. The file Tweets was received in json format, which contains the contents of Tweets written by everyone in the world between 18:11:23 and 20:03:53 on 02/09/2010.After these tasks, I thought I should give you five recommendations.

Before I make a recommendation, this dataset has a lot of useful and useful information, but there are also a lot of data that is completely useless for some stakeholders. And the important thing is that this data is from people who tweeted, there was no data on tweets. Among the fields, there were many fields that were NULL and many fields that were FALSE values. As one of them, all the validated fields in the user field were FALSE.

So the first thing I would like to recommend is that, before we start working on the project, the stakeholders who are going to write this data should know that all the users in this TWEETS data are not verified.It depends on which project you're working on, but if you need the user's verification, this data will be completely useless.

The second recommendation is that the LOCATION of some users written in this data is not accurate. As a result of looking at all of these data, the user's LOCATION is not accurate. Some users wrote what kind of device they wrote and even had coordinates, but not others. For example, the user of ObjectId ("5c8eccb0caa187d17ca626da") reads "Tofu Town :" on the user's LOCATION. Because of this, we can see that the LOCATION of this data is not accurate.

As a third recommendation, I thought it was difficult to find information with only this pure data. So if you look at Task8, user.name field and text field are combined, and the reason is simple. This is because I thought that unless some large companies were working on projects with data, it might not be necessary much other than these two fields. And on the data alone, it takes too long to find which TEXT a user enters, and if you know the name of the user and know what TEXT the user writes, it's only a matter of time before you find other fields you want, so I recommend you combine user.name and text fields like Task8.

Fourthly, this data called TWEETS is about users who wrote TWEET between 18:11:23 and 20:03:53 on 02/09/2010. That means that this data called TWEETS is 12 years old. Therefore, users and companies who will use this data must know and use it before using it that it contains records 12 years ago.

Finally, the recommendation is that there are many fields called IDs in this data that may be confusing, so you should carefully look at what IDs mean.

Write the name and student number of each of your group members in a separate column. For each person, indicate the extent to which your team agrees with the statement on the left (**SA - Strongly disagree; D =disagree; A=agree; SA=strongly agree**).

Evaluation Criteria	**Group member:** Irfan Rashad n10968741	**Group member:** Nicholas Faleao n11288027	**Group member:** Andrew Anderson n11072351	**Group member:** Minjae Lee n11198885
Attends group meetings and contributes meaningfully to group discussions.	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD
Completes assigned tasks on time.	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	SA/A/[✓ D]/SD
Prepares high-quality work.	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	SA/[✓ A]/D/SD
Demonstrates a cooperative and supportive attitude.	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD
Contributes significantly to the success of the project.	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD	[✓ SA]/A/D/SD
Based on these considerations, state a peer mark that each team member should receive out of 10.	10/10	10/10	10/10	7/10