```
f_count = args

// make a connection to the database server
conn = new Mongo();

// set the default database
db = conn.getDB("a1");
// duplicate the tweets collection and update the created_at type
// the new collection name is tweets_v2
// aggregation pipe line is used to avoid transferring the entire
// collection to the client side
db.tweets.aggregate(
[
    {
       $project: {
            id: 1,
          user_id: 1,
            retweet_id: 1,
            replyto_id: 1,
            hash_tags:1,
          user_mentions: 1,
            created_at: {
                $toDate: "$created_at"
           },

           text:1
       }
    },
    {
       $out: 'tweets_v2',
    },
 ]);

db.users.aggregate(
[
    {$out: "users_v2"}
]);

//index
    db.tweets_v2.createIndex({id:1})
    db.tweets_v2.createIndex({replyto_id:1})
    db.tweets_v2.createIndex({retweet_id:1})
    db.users_v2.createIndex({id:1})
    db.users_v2.createIndex({location:1})
    db.users_v2.createIndex({description:1})
// //optionally timing the execution
var start = new Date()
//q1:find the number of general tweets with at least one reply and one retweet in
the data set.
cursor  = db.tweets_v2.aggregate(
    [
        {$lookup:
            {
                from:"tweets_v2",
            localField: "id",
            foreignField: "replyto_id",
            as: "Reply_general"

            }
        },
        {$lookup:
            {
                from:"tweets_v2",
            localField: "id",
            foreignField: "retweet_id",
            as: "Retweet_general"

            }
        },
        {$project:
            {_id:1,
              id:1,
              retweet_count:1,
              retweet_id:1,
              replyto_id:1,
              Reply_general:1,
              Retweet_general:1,
              size_of_rep:{$size: "$Reply_general"},
              size_of_retweet_arr: {$size: "$Retweet_general"}
            }
        },
        {$match:
            {$and : [
                        {retweet_id:{$exists:false}},
                        {replyto_id:{$exists:false}},
                        {"size_of_rep":{$gt:0}},
                        {"size_of_retweet_arr":{$gt:0}}
                    ]
             }
        },
        {$group:
            {_id:0, count:{$sum:1}}
        },
        {$project:
            {_id:0, "Number of tweets":"$count"}
        }
    ]
)


print("Q1 ====================")
// display the result

while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}
// Query 2: Find the reply tweet that has the most retweets in the data set.
cursor  = db.tweets_v2.aggregate(
    [
        {
            $match: {retweet_id: {$exists:true}}
        },
        {
            $lookup: {
                        from: "tweets_v2",
                        localField: "retweet_id",
                        foreignField: "id",
                        as: "original_tweet"
```

```
                     }
        },
        {
            $match: {
                        $expr: {$gt: [{$size: "$original_tweet"},0 ]}
                    }
        },
        {
            $unwind: "$original_tweet"

        },
        {
            $match: {
                        "original_tweet.replyto_id": {$exists: true}
                    }
        },
        {
            $group: {
                        _id: "$original_tweet.id", retweet_count: {$sum : 1}
                    }
        },
        {
            $sort:{retweet_count:-1}
        },
        {
            $limit: 1
        },
        {$project: { id:"$_id",_id:0, "retweet_count":"$retweet_count"}}

    ]
)


print("Q2 ===================")
while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}

// Query 3: Find the top 5 hashtags appearing as the FIRST hashtag in a general or reply tweet, ignoring the case of the hashtag.
cursor  = db.tweets_v2.aggregate(
    [
        {
            $match: {
                        $or :[
                                {
                                   replyto_id: {$exists:true}
                                },
                                {
                                   $and : [
                                            {replyto_id:{$exists:false}},
                                            {retweet_id:{$exists:false}}
                                          ]
                                }
                             ]
                    }
        },

        {
```

```
    $unwind : "$hash_tags"
},
{
     $project:
            {
                 text:1,
                 hashtags:{$toLower : "$hash_tags.text"}
            }
},

{   $group: {
        "_id": "$_id",
        "first_tag": { "$first": "$hashtags" }
    }
},
{   $group:{
        _id:"$first_tag",
        hashtags_count:{$sum:1}
            }
},
{   $sort:{
        hashtags_count:-1,
        _id:1
           }
},
{   $limit: 5
},
{       $project:
            {
```

```
                        _id:0, tag: "$_id", count:"$hashtags_count"

                     }

        }





    ]
)


print("Q3 ===================")
while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}



// Query 4:Find the top 5 users with the most followers count

cursor  = db.tweets_v2.aggregate(
    [
        {$match :
            {
            $and: [
                    {hash_tags:{$exists:true}},
                    {user_mentions:{$exists:true}}
                 ]
            }
        },
        {$unwind : "$user_mentions" },
        {
            $lookup:
            {
                from: "users_v2",
                localField: "user_mentions.id",
                foreignField: "id",
                as: "userData"
            }
        },
        {$unwind : "$userData"},
        {$match: {'hash_tags.text':f_count}},
        {$match: {'userData.id': {$exists:true}}},
        { $project:{
          _id:1,
          id:"$userData.id",
          name:"$userData.name",
          location:"$userData.location",
          follower_count:"$userData.followers_count"
          }
        },
        {$group: {
        _id:"$id",
        id: {$first:"$id"},
        name: {$first:"$name"},
        location: {$first:"$location"},
        follower_count: {$first:"$follower_count"}
        }},
        {
          $sort:{follower_count:-1}
        },
        {
          $limit : 5
        },
        {$project: {
            _id : 0,
            id:1,
            name:1,
            location:1,
            follower_count:1
            }

         }







    ]
)


print("Q4 ===================")
print("input tag: " + f_count)
while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}
// Query 5: Find the number of general tweets published by users with neither
location nor description information.
cursor  = db.tweets_v2.aggregate(
    [
        {$match :
            {
            $and: [
                    {retweet_id:{$exists:false}},
                    {replyto_id:{$exists:false}}
                 ]
            }
        },
        {
            $lookup:
            {
                from: "users_v2",
                localField: "user_id",
                foreignField: "id",
                as: "userData2"
            }
        },
        {$unwind: "$userData2"},
        {
            $match :
            {
                $and: [
                    {'userData2.location':{$eq:""}},
                    {'userData2.description':{$eq:""}}
                 ]
            }
        },
        {$group : {_id: null,tweet_count:{$sum : 1}}},
        {$project: {_id:0,tweet_count:1}}

    ]
)


print("Q5 ====================")
while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}

// Query 6: Find the general tweet that receives most retweets in the first hour
after it is published.
cursor  = db.tweets_v2.aggregate(
    [
       {$lookup:
            {
                from: "tweets_v2",
                localField: "id",
                foreignField: "retweet_id",
                as: "retweet"
            }
        },
        {$unwind: "$retweet"},
        {$match: {
                $and: [ {retweet_id:{$exists:false}},
                        {replyto_id:{$exists:false}}
                       ]
                }
        },
        {
             $project: {
                        id:1,
                        create_time: {$toDate:"$created_at"},
                        retweet_time: {$toDate:"$retweet.created_at"}
                        }
        },

        {    $project: {
                        id:1,
                        create_time: 1,
                        retweet_time: 1,
                        duration : {$divide:[{$subtract:
["$retweet_time","$create_time"]},3600000]}
            }
         },
         {   $match :{
                        duration:{$lte:1}
                     }
         },
         {   $group :{_id:"$id",id: {$first:"$id"},  retweet_count:{$sum:1}}

         },
         {   $project:{_id:0,id:1,retweet_count:1}},
         { $sort:{retweet_count:-1}},
         { $limit:1}


    ]
)


print("Q6 ====================")
while ( cursor.hasNext() ) {
    printjson( cursor.next() );
}
var end = new Date()
print("Execution time: " + (end - start) + "ms")
// drop the newly created collection
db.tweets_v2.drop()
db.users_v2.drop()
```