

# IMAT3104: Database Management and Programming Assignment

LAMPROS KARSERAS  
P2424629

# Table of Contents

---

Q1:.....	3
Q2:.....	4
Q3:.....	5
Q4:.....	6
Q5:.....	7
Q6:.....	8
Q7:.....	9
Q8:.....	10
Q9:.....	11
E1: .....	12
D1:.....	14

## Q1:

*Find the description and price of the product entitled “Kid's Astronaut Hat”. No other product details are required.*

```
db.getCollection("P2424629_products").find(  
  { "lk_title": "Kid's Astronaut Hat" },  
  { "lk_description": 1, "lk_price": 1, "_id": 0 }  
)
```

Robo 3T - 1.4

File View Options Window Help

The screenshot shows the Robo 3T 1.4 application interface. On the left is a tree view of the database structure. The main area on the right displays a MongoDB query and its result.

**Tree View:**

- IMAT3104 (5)
  - System
  - config
  - csj
    - Collections (4)
      - Copy\_of\_P2424629\_products
      - Copy\_of\_P2424629\_reviews
      - P2424629\_products (selected)
        - Indexes (1)
          - \_id\_
        - P2424629\_reviews
      - Functions
      - Users
    - pd

**Query Editor:**

```
* db.getCollection("P2424629_...")
```

**Query:**

```
1 db.getCollection("P2424629_products").find(  
2   { "lk_title": "Kid's Astronaut Hat" },  
3   { "lk_description": 1, "lk_price": 1, "_id": 0 }  
4 )
```

**Result:**

```
P2424629_products 0.001 sec.  
1 /* 1 */  
2 {  
3   "lk_price" : 19.95,  
4   "lk_description" : "What kid hasn't dreamed of traveling  
5 }
```

Figure 1

## Q2:

*Find all details of products that are priced at either \$7.99 or \$14.99.*

```
db.getCollection("P2424629_products").find(  
  { $or: [ { "lk_price": 7.99 }, { "lk_price": 14.99 } ] }  
)
```

Robo 3T - 1.4

File View Options Window Help

New release available. Find out [what's new in Robo 3T](#) - [Download here](#).

IMAT3104 (5)

- System
- config
- csj
  - Collections (4)
    - Copy\_of\_P2424629\_products
    - Copy\_of\_P2424629\_reviews
    - P2424629\_products
      - Indexes (1)
        - \_id\_
    - P2424629\_reviews
  - Functions
  - Users
  - pd

\* db.getCollection("P2424629\_products").find(  
 { \$or: [ { "lk\_price": 7.99 }, { "lk\_price": 14.99 } ] }  
)

P2424629\_products 0.002 sec.

Key	Value
(1) ObjectId("5d0b6d1cd7367de7f58b4980") { 7 fields }	
_id	ObjectId("5d0b6d1cd7367de7f58b4980")
lk_asin	7842955867
lk_title	Blessed By Pope Benedetto XVI Padre Pio Rose Scent
lk_price	7.99
lk_imUrl	http://ecx.images-amazon.com/images/I/41EVtNYK
lk_salesRank	{ 1 field }
lk_categories	[ 2 elements ]
(2) ObjectId("5d0b6d1cd7367de7f58b497f") { 8 fields }	
_id	ObjectId("5d0b6d1cd7367de7f58b497f")
lk_asin	7842953562
lk_title	Blessed By Pope Benedetto XVI Our Lady of Charity I
lk_price	14.99
lk_imUrl	http://ecx.images-amazon.com/images/I/41C3QDH
lk_related	{ 1 field }
lk_salesRank	{ 1 field }
lk_categories	[ 2 elements ]
(3) ObjectId("5d0b6d1cd7367de7f58b49ea") { 10 fields }	
_id	ObjectId("5d0b6d1cd7367de7f58b49ea")
lk_asin	B00000755K
lk_related	{ 3 fields }
lk_title	Girls Will Be Girls
lk_price	14.99
lk_salesRank	{ 1 field }
lk_imUrl	http://ecx.images-amazon.com/images/I/51B5jGH2
lk_brand	
lk_categories	[ 3 elements ]
lk_description	Shot in High Definition. Yes it's true, girls will be girls
(4) ObjectId("5d0b6d1cd7367de7f58b4a80") { 7 fields }	
_id	ObjectId("5d0b6d1cd7367de7f58b4a80")
lk_asin	B00005KJXQ
lk_title	Rubie's Howling Ghost Costume
lk_price	7.99
lk_imUrl	http://ecx.images-amazon.com/images/I/31n-113D
lk_brand	Rubie's Costume Co
lk_categories	[ 1 element ]

Figure 2

Q3:

*Count the number of products that have a price but do not have a description.  
Clearly show the number you receive as your answer.*

```
db.getCollection("P2424629_products").find(  
  { "lk_price": { $ne: null }, "lk_description": null }  
) .count()
```

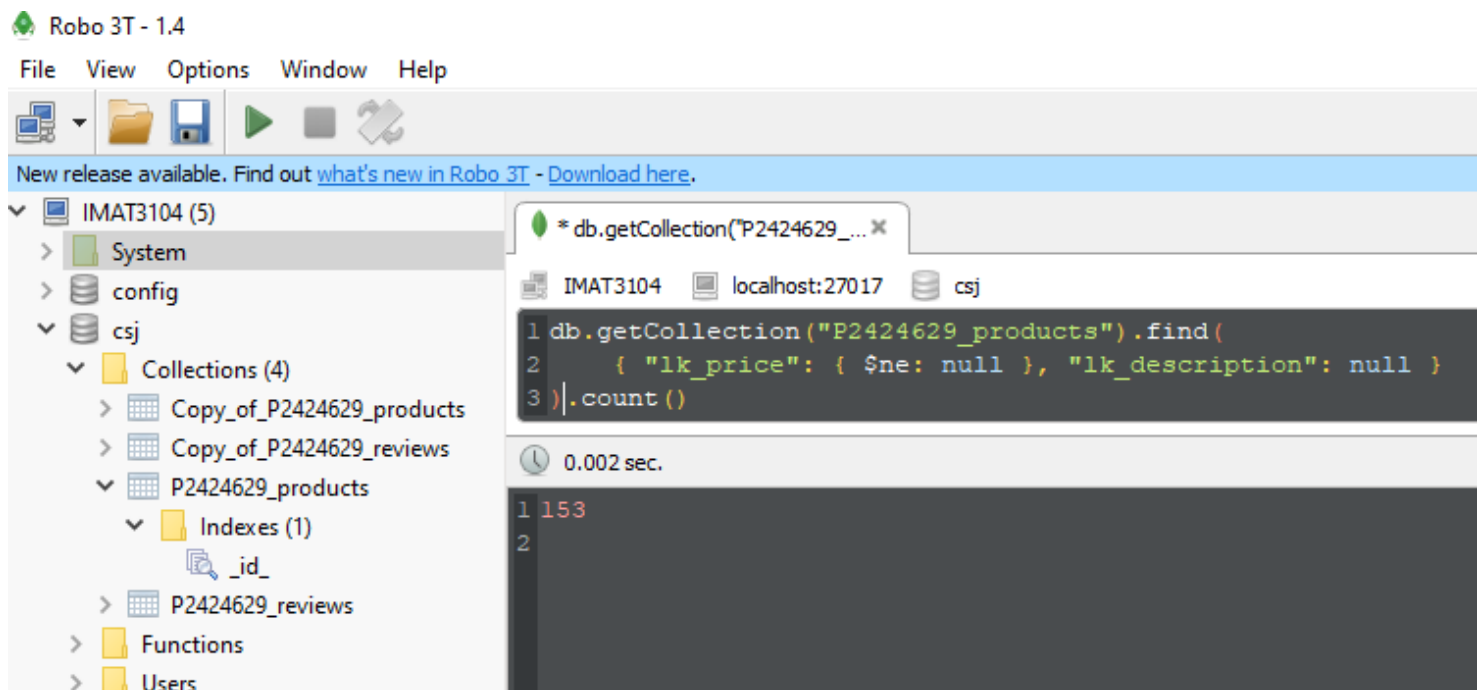


Figure 3

## Q4:

List the product IDs of products that were reviewed before the 15<sup>th</sup> June 2000. All calculations should be performed by MongoDB. Only list the product IDs and do not show any duplicates.

```
var filterTime = ISODate("2000-06-15").getTime() / 1000

db.P2424629_reviews.aggregate([

  { $match: { lk_unixReviewTime: { $lt: filterTime } } },
  { $group: { _id: "$lk_asin" } }

])
```

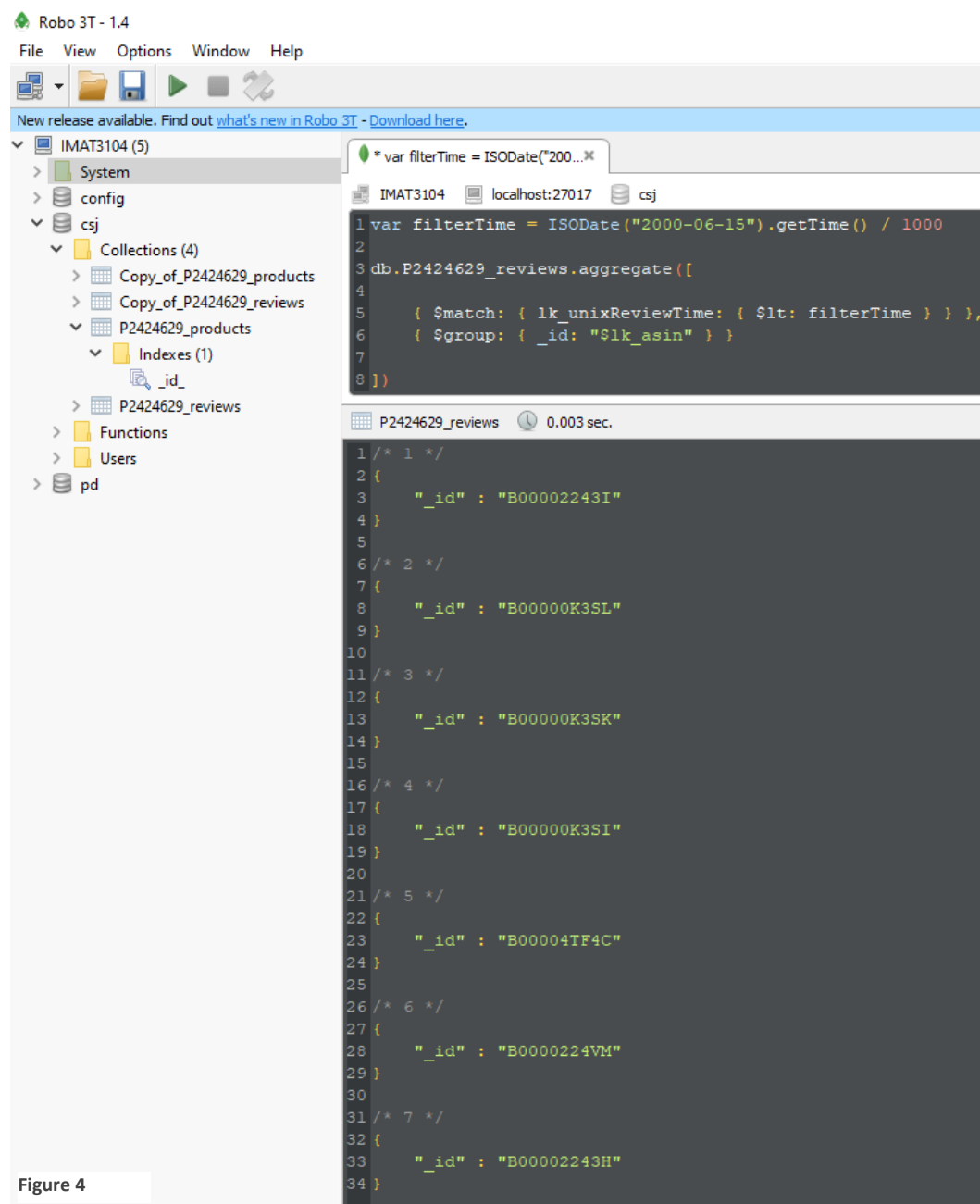


Figure 4

## Q5:

List all the reviewers who have written at least 3 reviews. Show the reviewer ID, reviewer name and the number of reviews for each one. List the names of reviewers in alphabetical order. [Hint: it is possible to solve using aggregate pipeline method].

```
db.getCollection("P2424629_reviews").aggregate([
  {
    "$group": {
      "_id": "$lk_reviewerID",
      "NumberOfReviews": { "$sum": 1 },
      "ReviewerName": { "$first": "$lk_reviewerName" }
    }
  },
  { "$match": { "NumberOfReviews": { "$gte": 3 } } },
  { "$sort": { "ReviewerName": 1 } },
  {
    "$project": {
      "_id": 0,
      "ReviewerID": "$_id",
      "ReviewerName": "$ReviewerName",
      "NumberOfReviews": "$NumberOfReviews"
    }
  }
])
```

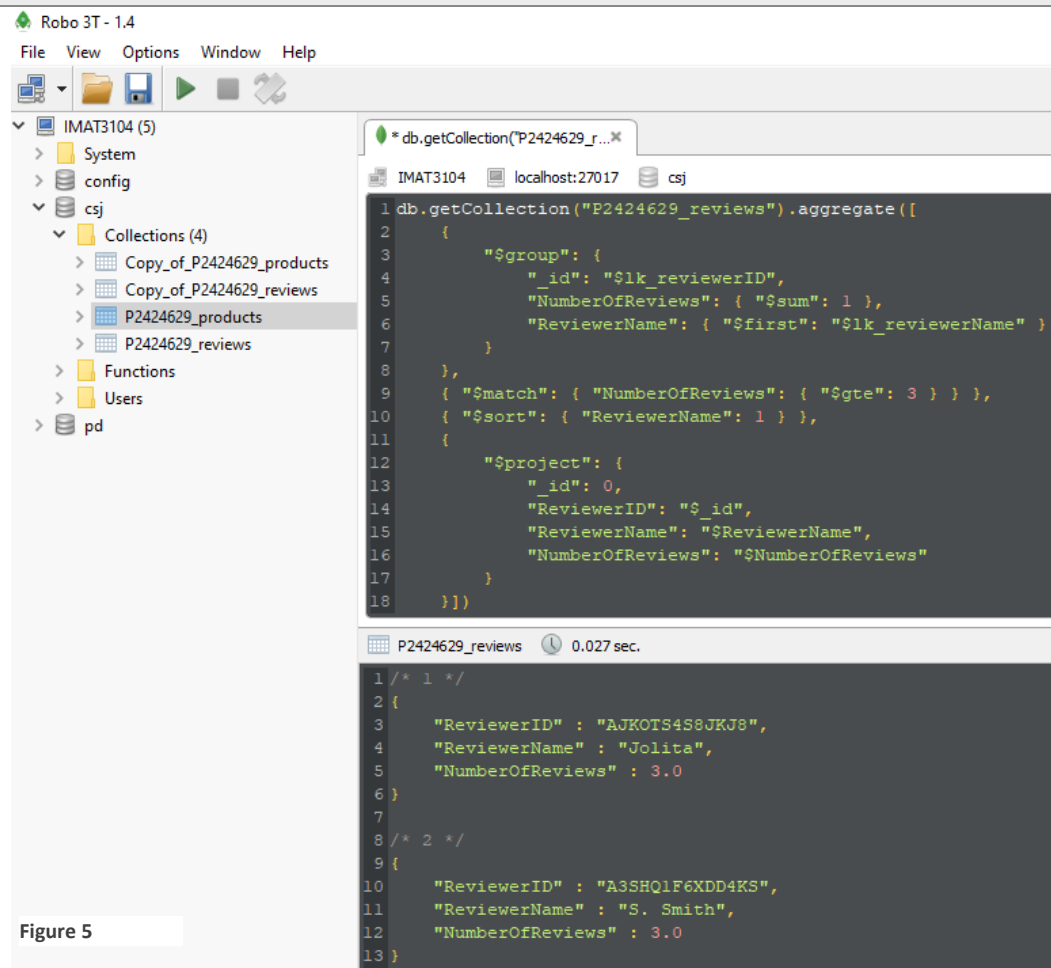


Figure 5

## Q6:

Count the number of related products that were also viewed with the product "Mens Timex". Only give the number of related products as your answer. [Hint: it is possible to solve using aggregate pipeline method].

```
db.getCollection("P2424629_products").aggregate([
  {
    $match: { lk_title: "Mens Timex" }
  },
  {
    $project: {
      _id: 0,
      NumberOfRelatedViewed: { $size: "$lk_related.also_viewed" }
    }
  }
])
```

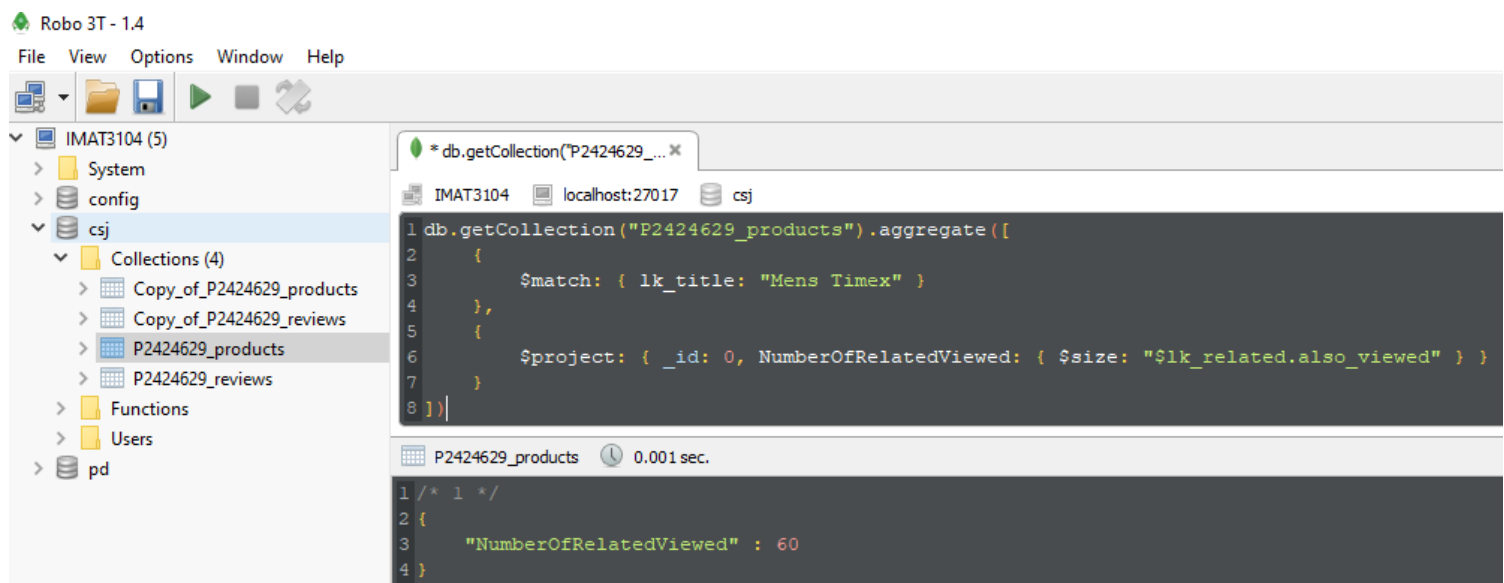


Figure 6



## Q7:

List the product ID and title of all products in the “T-Shirts” category. No other product details are required. [Hint: it is possible to solve using aggregate pipeline method].

```
db.getCollection("P2424629_products").aggregate([
  { "$unwind": { "path": "$lk_categories" } },
  { "$match": { "lk_categories": "T-Shirts" } },
  {
    "$project": {
      "_id": 0,
      "ProductID": "$lk_asin",
      "ProductTitle": "$lk_title"
    }
  }
])
```

Robo 3T - 1.4

File View Options Window Help

IMAT3104 (5)

- System
- config
- csj
  - Collections (4)
    - Copy\_of\_P2424629\_products
    - Copy\_of\_P2424629\_reviews
    - P2424629\_products
    - P2424629\_reviews
  - Functions
  - Users
  - pd

\* db.getCollection("P2424629\_... x

IMAT3104 localhost:27017 csj

```
1 db.getCollection("P2424629_products").aggregate([
2   { "$unwind": { "path": "$lk_categories" } },
3   { "$match": { "lk_categories": "T-Shirts" } },
4   {
5     "$project": {
6       "_id": 0,
7       "ProductID": "$lk_asin",
8       "ProductTitle": "$lk_title"
9     }
10  }
11 ])
```

P2424629\_products 0.003 sec.

	ProductID	ProductTitle
1	5555014368	Top Gear Official Merchandise - Captain Slow T-Shirt
2	5555014090	Official Top Gear 'My Dad is The Stig' T-Shirt
3	555501239X	Top Gear Official Merchandise - Stig Navy
4	6041616796	Muay Thai Kick Boxing T-shirt Thai Tattoo Twin Tigers (XL)
5	6041006041	Sevendayz Men's The Weeknd Xo Long Sleeve Baseball T-Shirt
6	9789841655	Tommy Bahama Embroidered Port Authority Silk Camp Shirt (Color: Black, Size L)
7	9820303354	Tommy Bahama Embroidered Legendary Lures Silk Camp Shirt (Color: Oceanaire Blue, Size XL)
8	B000072SJ2	Loosegear Short Sleeve

Figure 7

Q8:

Find the highest rating by the reviewer identified as "A2B3TVWTZ7609A" by using **map reduce** programming.

```
var map = function () {
    emit(this.lk_reviewerID, this.lk_overall)
}

var reduce = function (key, values) {
    var max = values[0]
    values.forEach(function (value) {
        if (value > max) {
            max = value
        }
    })
    return max
}

db.getCollection("P2424629_reviews").mapReduce(map, reduce, {
    query: { lk_reviewerID: "A2B3TVWTZ7609A" },
    out: { inline: 1 }
})
```

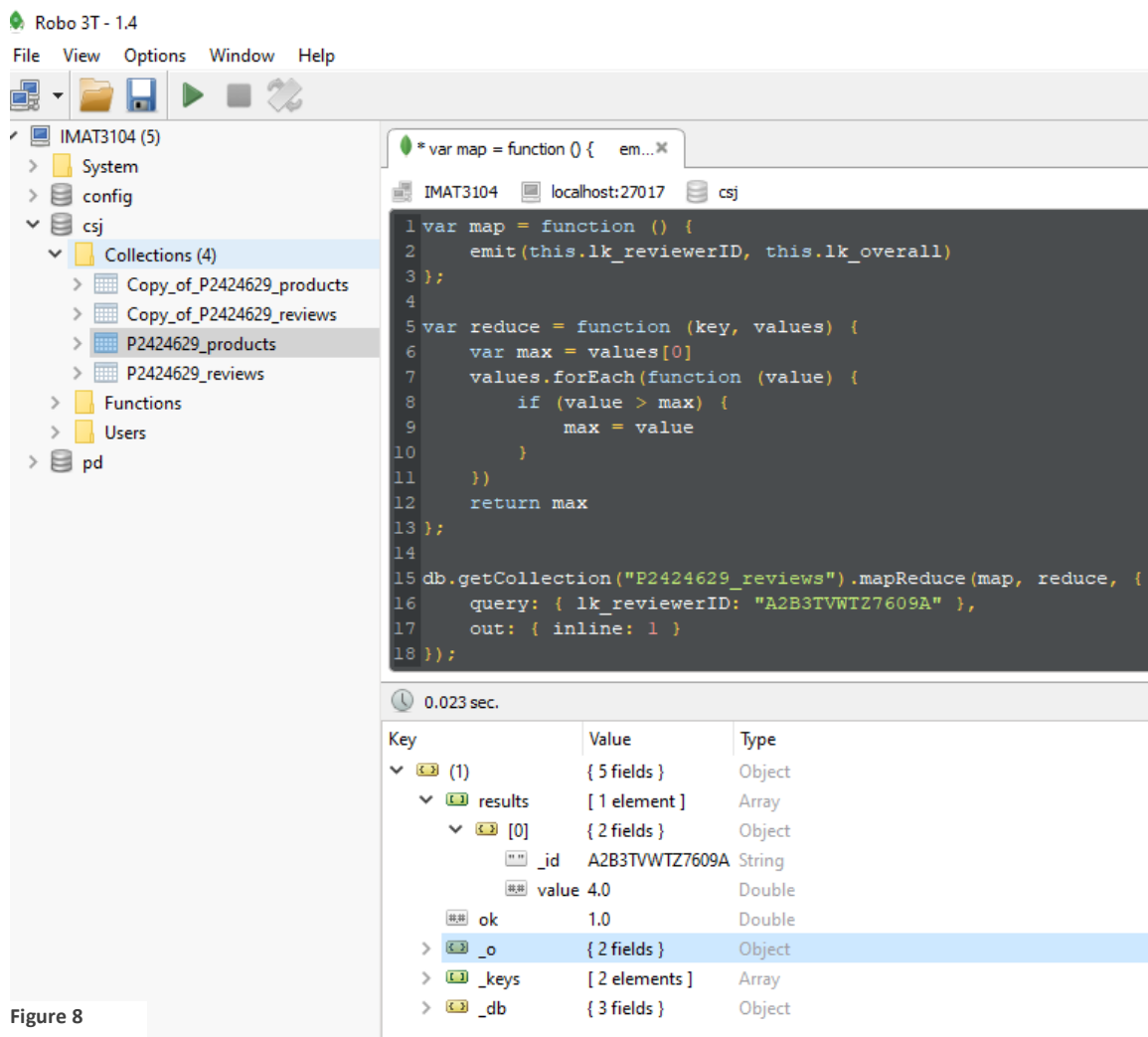


Figure 8

## Q9:

Using both collections, find the title and description of products reviewed by the reviewer identified by "A2C65Q9IDNCSR". No other product details are required.

```
db.getCollection("P2424629_reviews").aggregate([
  { $match: { lk_reviewerID: "A2C65Q9IDNCSR" } },
  {
    $lookup: {
      from: "P2424629_products",
      localField: "lk_asin",
      foreignField: "lk_asin",
      as: "product_data"
    }
  },
  { $unwind: "$product_data" },
  {
    $project: {
      _id: 0,
      Title: "$product_data.lk_title",
      Description: "$product_data.lk_description",
    }
  }
])
```

Robo 3T - 1.4

File View Options Window Help

The screenshot shows the Robo 3T interface. On the left, a tree view shows the database structure: IMAT3104 (5) containing System, config, csj, and pd. Under csj, there are four collections: Copy\_of\_P2424629\_products, Copy\_of\_P2424629\_reviews, P2424629\_products, and P2424629\_reviews. The main window displays a MongoDB aggregation query in a text editor. The query is the same as the one shown in the code block above. Below the editor, the results of the query are shown in a table view. The table has two columns: Title and Description. The first result is for the product 'Buzz Lightyear Deluxe - Size: Child S(4-6)' with the description 'Kids can get ready to blast off \to infinity and beyond'. The second result is for the product 'Elope Kid's Magician Hat With Rabbit, Black/White, One Size' with the description 'Presto, change-o! Children can transform themselves into'.

```
1 db.getCollection("P2424629_reviews").aggregate([
2   { $match: { lk_reviewerID: "A2C65Q9IDNCSR" } },
3   {
4     $lookup: {
5       from: "P2424629_products",
6       localField: "lk_asin",
7       foreignField: "lk_asin",
8       as: "product_data"
9     }
10  },
11  { $unwind: "$product_data" },
12  {
13    $project: {
14      _id: 0,
15      Title: "$product_data.lk_title",
16      Description: "$product_data.lk_description",
17    }
18  }
19 ]])
```

Title	Description
"Buzz Lightyear Deluxe - Size: Child S(4-6)",	"Kids can get ready to blast off \to infinity and beyond"
"Elope Kid's Magician Hat With Rabbit, Black/White, One Size",	"Presto, change-o! Children can transform themselves into"

Figure 9

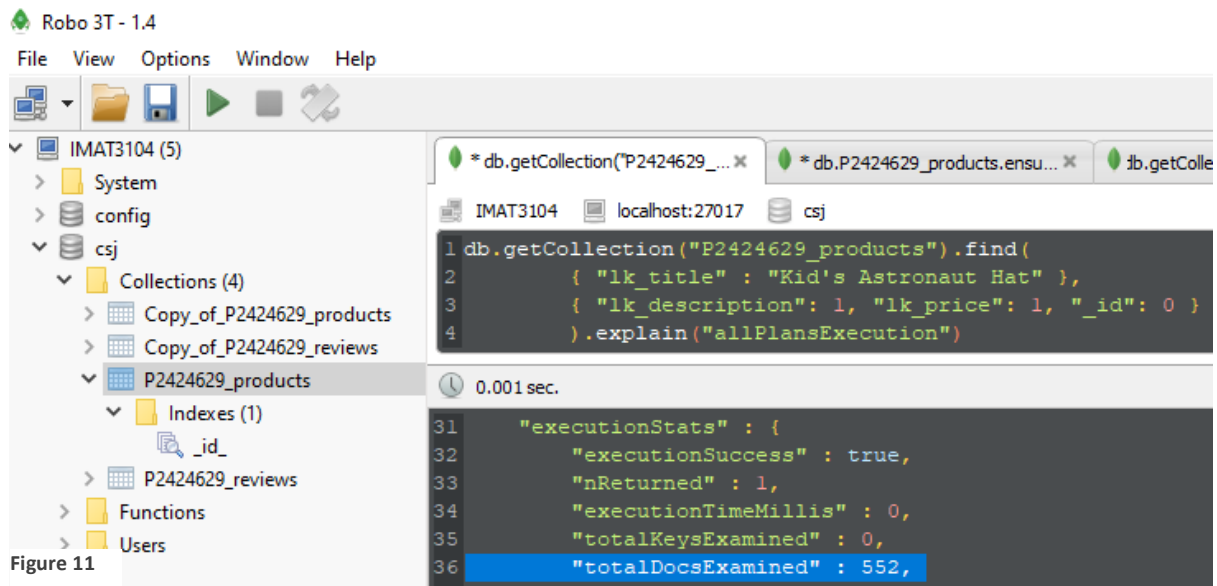
*Implement one index that would improve the querying of the database based on one or more of the queries (Q1-Q9). Identify the chosen query or queries and explain and justify your choice of index. Present, explain and compare execution plans to support your choice and summarise your findings*

My rational for choosing “lk\_title” field as the index for products is, the benefits in query speeds for queries Q1 and Q6 and for future proof database design.

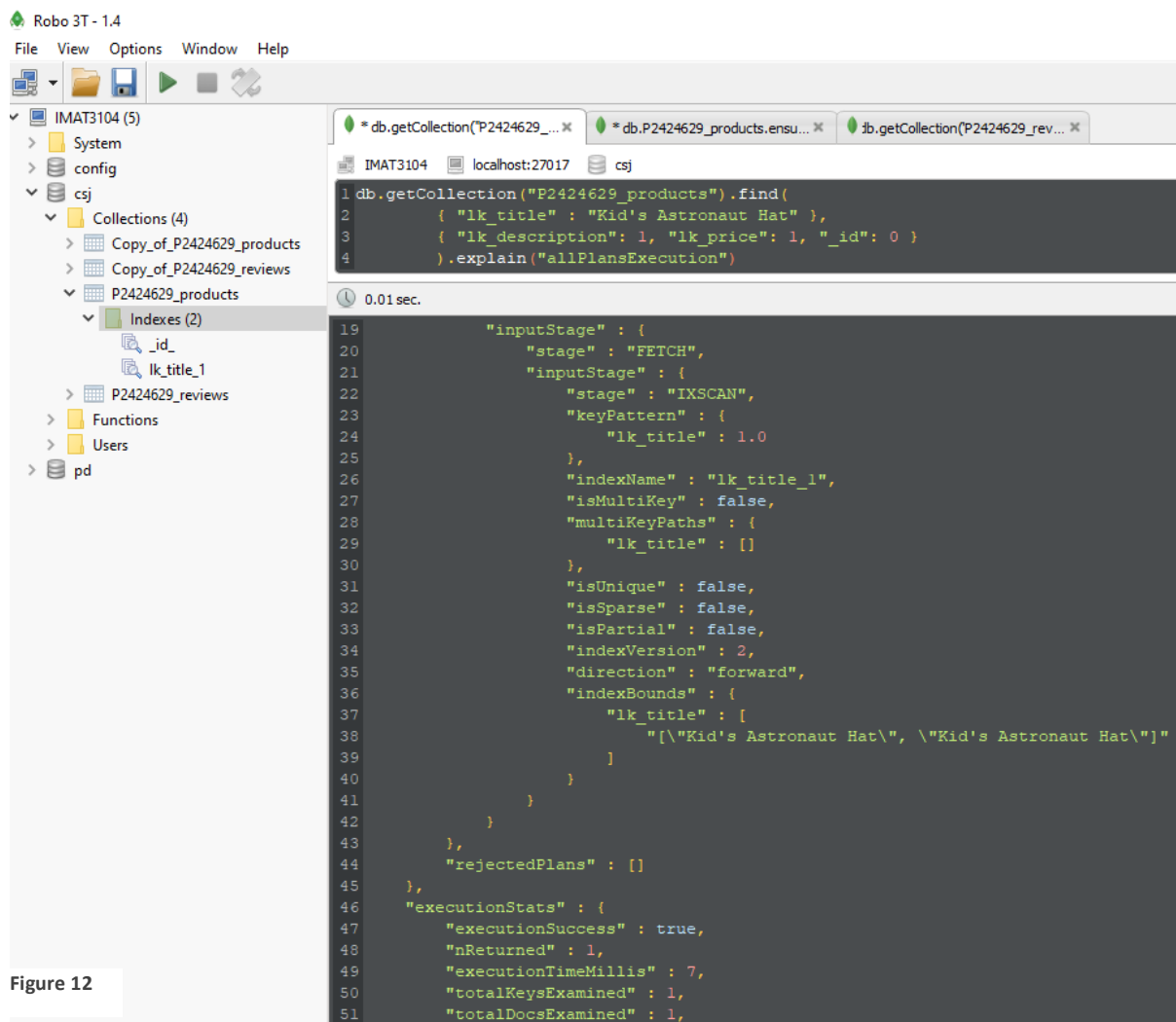
Another reason for choosing the “lk\_title” field is the very high **cardinality**. Since every product is unique, the indexing of “lk\_title” field will yield significant gains in speed.

For example, analysing the queries Q1 and Q6 with `explain("allPlansExecution")` shows the benefits of indexing the "lk\_title" field. As shown in Figure10, without indexing there is an InputStage of full collection Scan, which means the query "walked" through all the 552 documents to return 1, as shown in Figure11.





With indexing the “lk\_title” field, the query just immediately returns a single document, without examining all the collection, as show in Figure12. The input Stage is changed to FETCH and IXSCAN, which means it looks just in indexed entries, and examined just a single document.



D1:

---

*Write code in MongoDB to automatically embed the details of reviews from the reviews collection with their corresponding product in the product collection.*

*Therefore, the products collection will contain both the product data and review data in a single collection of products.*

```
var reviews_cursor = db.Copy_of_P2424629_reviews.find({}, { _id: 0 })

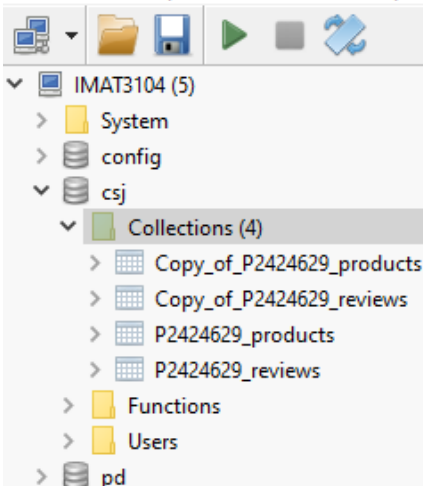
while (reviews_cursor.hasNext()) {
  var nextReviewDoc = reviews_cursor.next()
  var review_lk_asin = nextReviewDoc.lk_asin
  var review_lk_reviewerID = nextReviewDoc.lk_reviewerID
  var review_lk_reviewerName = nextReviewDoc.lk_reviewerName
  var review_lk_helpful = nextReviewDoc.lk_helpful
  var review_lk_reviewText = nextReviewDoc.lk_reviewText
  var review_lk_overall = nextReviewDoc.lk_overall
  var review_lk_summary = nextReviewDoc.lk_summary
  var review_lk_unixReviewTime = nextReviewDoc.lk_unixReviewTime
  var review_lk_reviewTime = nextReviewDoc.lk_reviewTime

  db.Copy_of_P2424629_products.update(
    { lk_asin: review_lk_asin },
    {
      $push:
        {
          lk_reviewerID: review_lk_reviewerID,
          lk_reviewerName: review_lk_reviewerName,
          lk_helpful: review_lk_helpful,
          lk_reviewText: review_lk_reviewText,
          lk_overall: review_lk_overall,
          lk_summary: review_lk_summary,
          lk_unixReviewTime: review_lk_unixReviewTime,
          lk_reviewTime: review_lk_reviewTime
        }
    })
}

// Query Testing with all the Review fields + lk_title
db.Copy_of_P2424629_products.find(
  { lk_asin: "B00005JHK9" },
  {
    _id: 0, lk_title: 1, lk_reviewerID: 1, lk_reviewerName: 1,
    lk_helpful: 1, lk_reviewText: 1, lk_overall: 1,
    lk_summary: 1, lk_unixReviewTime: 1, lk_reviewTime: 1
  })
```

Robo 3T - 1.4

File View Options Window Help



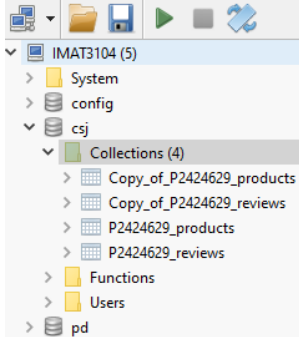
\* var reviews\_cursor = db.Cop...

IMAT3104 localhost:27017 csj

```
1 var reviews_cursor = db.Copy_of_P2424629_reviews.find({}, { _id: 0 })
2 while (reviews_cursor.hasNext()) {
3   var nextReviewDoc = reviews_cursor.next()
4   var review_lk_asin = nextReviewDoc.lk_asin
5   var review_lk_reviewerID = nextReviewDoc.lk_reviewerID
6   var review_lk_reviewerName = nextReviewDoc.lk_reviewerName
7   var review_lk_helpful = nextReviewDoc.lk_helpful
8   var review_lk_reviewText = nextReviewDoc.lk_reviewText
9   var review_lk_overall = nextReviewDoc.lk_overall
10  var review_lk_summary = nextReviewDoc.lk_summary
11  var review_lk_unixReviewTime = nextReviewDoc.lk_unixReviewTime
12  var review_lk_reviewTime = nextReviewDoc.lk_reviewTime
13
14  db.Copy_of_P2424629_products.update(
15    { lk_asin: review_lk_asin },
16    {
17      $push:
18    }
```

16 sec.

```
1 Updated 1 existing record(s) in 2ms
2 Updated 1 existing record(s) in 1ms
3 Updated 1 existing record(s) in 1ms
4 Updated 1 existing record(s) in 1ms
5 Updated 1 existing record(s) in 1ms
6 Updated 1 existing record(s) in 1ms
7 Updated 1 existing record(s) in 1ms
8 Updated 1 existing record(s) in 0ms
9 Updated 1 existing record(s) in 4ms
10 Updated 1 existing record(s) in 1ms
11 Updated 1 existing record(s) in 1ms
12 Updated 1 existing record(s) in 0ms
13 Updated 1 existing record(s) in 1ms
14 Updated 1 existing record(s) in 1ms
```



```

1 var reviews_cursor = db.Copy_of_P2424629_reviews.find({}, { _id: 0 })
2 while (reviews_cursor.hasNext()) {
3   var nextReviewDoc = reviews_cursor.next()
4   var review_lk_asin = nextReviewDoc.lk_asin
5   var review_lk_reviewerID = nextReviewDoc.lk_reviewerID
6   var review_lk_reviewerName = nextReviewDoc.lk_reviewerName
7   var review_lk_helpful = nextReviewDoc.lk_helpful
8   var review_lk_reviewText = nextReviewDoc.lk_reviewText
9   var review_lk_overall = nextReviewDoc.lk_overall
10  var review_lk_summary = nextReviewDoc.lk_summary
11  var review_lk_unixReviewTime = nextReviewDoc.lk_unixReviewTime
12  var review_lk_reviewTime = nextReviewDoc.lk_reviewTime
13
14  db.Copy_of_P2424629_products.update(
15    { lk_asin: review_lk_asin },
16    {
17      $push:
18        {
19          lk_reviewerID: review_lk_reviewerID,
20          lk_reviewerName: review_lk_reviewerName,
21          lk_helpful: review_lk_helpful,
22          lk_reviewText: review_lk_reviewText,
23          lk_overall: review_lk_overall,
24          lk_summary: review_lk_summary,
25          lk_unixReviewTime: review_lk_unixReviewTime,
26          lk_reviewTime: review_lk_reviewTime
27        }
28    })
29 }
30 // Query Testing
31 db.Copy_of_P2424629_products.find(
32   { lk_asin: "B000005JHK9" },
33   {
34     _id: 0, lk_title: 1, lk_reviewerID: 1, lk_reviewerName: 1,
35     lk_helpful: 1, lk_reviewText: 1, lk_overall: 1,
36     lk_summary: 1, lk_unixReviewTime: 1, lk_reviewTime: 1
37   })

```

```

robo3t
Copy_of_P2424629_products 0 sec.
1 /* 1 */
2 {
3   "lk_title" : "Gladiator Costume: Boy's Size 7-8",
4   "lk_helpful" : [
5     [
6       3.0,
7       4.0
8     ],
9     [
10      12.0,
11      13.0
12    ]
13  ],
14  "lk_overall" : [
15    1.0,
16    5.0
17  ],
18  "lk_reviewText" : [
19    "Looks cool but the plastic is very thin on the chest shield, leg and arm guards. The face plate ve",
20    "Overall, this is a neat costume for the money and my son thinks it's pretty cool. I think there a",
21  ],
22  "lk_reviewTime" : [
23    "10 1, 2003",
24    "10 3, 2002"
25  ],
26  "lk_reviewerID" : [
27    "A2Y0II6CM888FO",
28    "A2PIV63VGD2BFX"
29  ],
30  "lk_reviewerName" : [
31    "BAB \"BAB\"",
32    "\"dauph08\""
33  ],
34  "lk_summary" : [
35    "Gladiator Costume",
36    "Not bad for the money"
37  ],
38  "lk_unixReviewTime" : [
39    1064966400.0,
40    1033603200.0
41  ]
42 }

```