

CW5 mongodb

by Mei Tan

Submission date: 11-Apr-2020 04:38PM (UTC+0100)

Submission ID: 123958408

File name: 98979_Mei_Tan_CW5_mongodb_1211817_1949274170.pdf (142.21K)

Word count: 793

Character count: 4068

MongoDB with R packages

[Code ▾](#)

By Le Ping Tan (M00724895)

MongoDB with R packages was installed into the virtual environment and runned to execute below tasks:

Task 1

The content of students.json was extracted and imported into collection called students by firstly applied the required libraries and json file of project was imported into database of CW5 with collection named students as shown below:

[Hide](#)

```
library(mongolite)

students_scores <- mongo(collection = "students", db = "CW5")
students_data <- students_scores$import(file("students.json"))
```

```
incomplete final line found on 'students.json'
```

Next to check the count of data imported into collection - students.

[Hide](#)

```
students_scores$count()
```

```
[1] 200
```

Then simplify the collection into dataframe by according to id.

[Hide](#)

```
Students <- students_scores$find(sort = '{"_id":1}')
```

Task 2

Aggregate framework function was used by initial "project" function to splitting data. Then follow by "unwind" command to split sub features from the selected feature, which followed by "match" command to query students who achieved score of exam above 60. "group" query was used to compile the output from "match" command above. Lastly, "sort" was executed to end the aggregation framework by displaying the output in ascending order based on the exam scores.

[Hide](#)

```

examscore_abv_60 <- students_scores$aggregate(['{"$project":{"_id":"$_id", "name":"$name", "score":"$scores"}}',
  {"$unwind":"$score"},
  {"$match":{"score.type":"exam", "score.score":{"$gte":60}}},
  {"$group":{"_id":"$_id", "score_exam":{"$sum":"score.score"}}},
  {"$sort":{"score_exam":-1}}'])
examscore_abv_60

```

	_id <int>	score_exam <dbl>
1	1	60.06045
2	188	60.31473
3	79	61.20158
4	122	61.47627
5	18	62.12870
6	118	62.20458
7	105	62.28389
8	129	62.61424
9	111	62.74310
10	108	63.75595
1-10 of 80 rows		Previous 1 2 3 4 5 6 ... 8 Next

Task 3

Query to seek the last five students by showing their minimum and maximum scores was executed by aggregation framework as well. Follow with "project" and unwind function to split the features. However, "group" command executed based on students' Id for minimum and maximum scores. Lastly, "sort" parameter to allow in descending order and "limit" to restrict only 5 outputs.

[Hide](#)

```

score_last5 <- students_scores$aggregate(['{"$project":{"_id":"$_id", "name":"$name", "score":"$scores.score"}}',
  {"$unwind":"$score"},
  {"$group":{"_id":"$_id", "min_score":{"$min":"score"}, "max_score":{"$max":"score"}}},
  {"$sort":{"_id":-1}},
  {"$limit":5}'])
score_last5

```

	_id <int>	min_score <dbl>	max_score <dbl>
1	199	28.86824	82.11743

	_id <int>	min_score <dbl>	max_score <dbl>
2	198	11.90757	64.85650
3	197	31.16288	79.15856
4	196	33.63300	78.79257
5	195	11.75008	90.77751
5 rows			

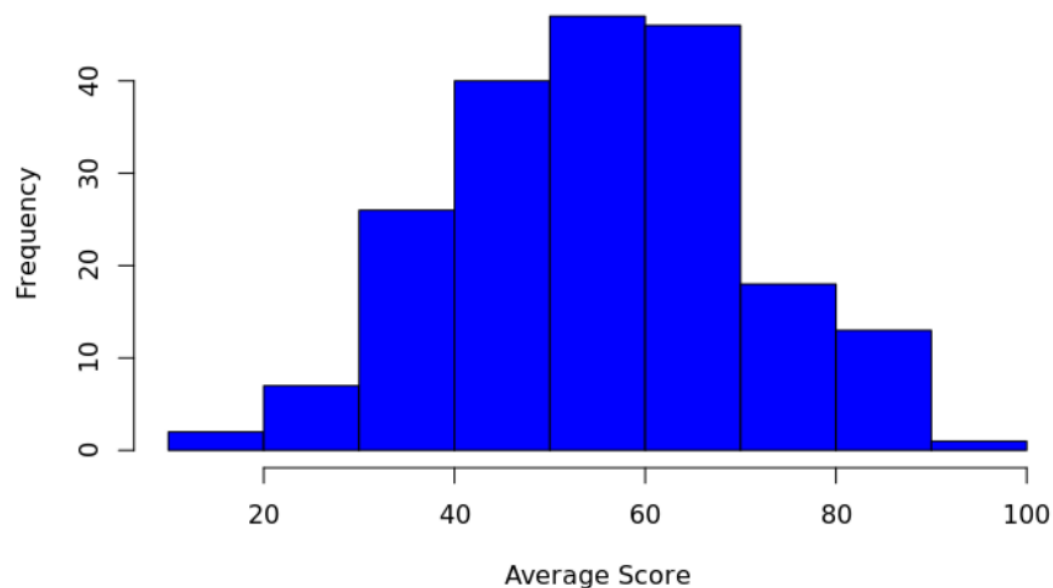
Task 4

To query the the average score for student was carried out by using aggregation framework as well. Initiated by "project", "unwind" parameters then followed by "group" based on the average score and sorted in ascending order. Lastly to visualised the distribution of students' average scores in the histogram as below:

Hide

```
1 score_avg <- students_scores$aggregate(['{"$project":{"_id":"$ _id", "name":"$name",
"score":"$scores"}}',
1      {"$unwind":"$score"},
      {"$group":{"_id":"$ _id", "Avg_score":{"$avg":"$score.score"}}},
      {"$sort":{"_id":1}}']')
hist(score_avg$Avg_score, main = "Histogram of Average Score", xlab = "Average Score", col = "blue")
```

Histogram of Average Score



Task 5

Binding of two datasets, original dataset of Students and score_avg which structured in task 4 was executed

to formed a new datasets. This new data is in dataframe structure with specific features of id, name and average score which later will be converted into csv file. Lastly it was extraced and displayed as shown below:

[Hide](#)

```
final_score <- cbind(Students, score_avg)
final_score <- data.frame(id = final_score$'_id', name = final_score$name, average_score = final_score$Avg_score)

final_score_csv <- write.csv(final_score, file = "finalscore.csv", row.names = FALSE)

final_score_db <- read.csv(file = "finalscore.csv")
final_score_db
```

id	name	average_score
<int>	<fctr>	<dbl>
0	aimee Zank	16.37332
1	Aurelia Menendez	61.53990
2	Corliss Zuk	46.53869
3	Bao Ziglar	46.23571
4	Zachary Langlais	67.79879
5	Wilburn Spiess	44.67290
6	Jenette Flanders	49.07345
7	Salena Olmos	76.46531
8	Daphne Zheng	37.57226
9	Sanda Ryba	73.36109

1-10 of 200 rows

Previous 1 2 3 4 5 6 ... 20 Next

ORIGINALITY REPORT

20%

SIMILARITY INDEX

0%

INTERNET SOURCES

0%

PUBLICATIONS

20%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Middlesex University

Student Paper

20%

Exclude quotes Off

Exclude bibliography Off

Exclude matches Off