	School of Computer Science & IT JAIN (DEEMED TO BE UNIVERSITY) Knowledge Campus, Bangalore
Department of Master of Computer Applications – Academic Year 2020 – 2021	

Department of Master of Computer Science
NoSQL Databases Lab (18MCAR306L)

Experiment 8, 9 & 10

Submitted By: **Submitted To:**

Yerraballi Suresh Kumar Reddy   Dr. Gangothri
19MCAR0080
Semester: 3rd MCA

Semester: 2nd MCA 'B'
Jain University,
9th Block Jayanagar,
Bangalore- 560069

EXPERIMENT 08

Aim

Anand Corp is a leading corporate training provider. A lot of prestigious organizations send their employees to Anand Corp for training on different skills. As a distinct training provider, Anand Corp has decided to share analysis report with their clients. This report will help their clients know the employees who have completed training and evaluation exam, what are their strengths, and what are the areas where employees need improvement. This is going to be a unique selling feature for the Anand Corp. As Anand Corp is already doing great business and they give training to a large number of people every month, they have huge amount of data to deal with. They have hired you as an expert and want your help to solve this problem.

Attributes of data – Id: id of the person who was trained, Name: name of the person who was trained, Evaluation: evaluation term, Score: score achieved by the person for the specific term.

A person can undergo multiple evaluations. Each evaluation will have a unique result score. You can see the sample data below.

Sample Data

{ "_id": 0, "name": "Andy", "results": [{ "evaluation": "term1", "score": 1.463179736705023 },{ "evaluation": "term2", "score": 11.78273309957772 }, { "evaluation": "term3", "score":6.676176060654615 }]}

PQR Corp has assigned the following tasks to you to analyze the results:

Find count and percentage of employees who failed in term 1, the passing score being 37.

Procedure

After inserting the specified data and other similar entries, here's how to find count and percentage of
employees who failed in term 1, **db.anandcorp.aggregate([{ $unwind:"$results"}, { $match:{ "results.evaluation": "term1"} },{ $group: { _id: null, totalCount: { $sum:1}, failCount: { $sum:{ "$cond": [{ "$lt": ["$results.score", 37] }, 1, 0] } } } }, { $project: { _id: 0, totalCount: 1,failCount: 1, failPercent: { $multiply: [{ $divide: ["$failCount", "$totalCount"] }, 100] } } }]);**

Input:

```
> use exp8
switched to db exp8
> db.anandcorp.insert"_id": 0, "name": "Andy", "results": [{ "evaluation": "term1", "score": 1.463179736705023 },
... { "evaluation": "term2", "score": 11.78273309957772 }, { "evaluation": "term3", "score":
... 6.676176060654615 }]}
uncaught exception: SyntaxError: unexpected token: string literal :
@(shell):1:19
>
> db.anandcorp.insert({"_id": 0, "name": "Andy", "results": [{ "evaluation": "term1", "score": 1.463179736705023 },{ "evaluation": "term2", "score": 11.78273309957772
}, { "evaluation": "term3", "score":6.676176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 1, "name": "suresh", "results": [{ "evaluation": "term1", "score": 19.463179736705023 },{ "evaluation": "term2", "score": 1.7827330995777
2 },{ "evaluation": "term3", "score":3.676176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 2, "name": "harish", "results": [{ "evaluation": "term1", "score": 12.463179736705023 },{ "evaluation": "term2", "score": 13.782733099957
72 }, { "evaluation": "term3", "score":16.126176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 3, "name": "ganesh", "results": [{ "evaluation": "term1", "score": 2.463179736705023 },{ "evaluation": "term2", "score": 4.78273309957772
 },{ "evaluation": "term3", "score":1.126176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 4, "name": "tharun", "results": [{ "evaluation": "term1", "score": 12.463179736705023 },{ "evaluation": "term2", "score": 14.782733099957
72 }, { "evaluation": "term3", "score":9.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 5, "name": "sai", "results": [{ "evaluation": "term1", "score": 17.463179736705023 },{ "evaluation": "term2", "score": 16.78273309957772
}, { "evaluation": "term3", "score":15.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.insert({"_id": 6, "name": "sandeep", "results": [{ "evaluation": "term1", "score":17.463179736705023 },{ "evaluation": "term2", "score": 6.7827330995777
2 }, { "evaluation": "term3", "score":5.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
> db.anandcorp.aggregate( [ { $unwind:"$results"}, { $match:{ "results.evaluation": "term1" },{ $group: { _id: null, totalCount: { $sum:1}, failCount: { $sum:{ "$cond
": [ { "$lt": [ "$results.score", 37 ] }, 1, 0 ] } } } }, { $project: { _id: 0, totalCount: 1,failCount: 1, failPercent: { $multiply: [ { $divide: [ "$failCount", "$to
talCount" ] }, 100 ] } } } ] )
{ "totalCount" : 7, "failCount" : 7, "failPercent" : 100 }
> db.anandcorp.insert({"_id": 7, "name": "kan", "results": [{ "evaluation": "term1", "score":17.463179736705023 },{ "evaluation": "term2", "score": 36.78273309957772 }
, { "evaluation": "term3", "score":45.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
```

```
> db.anandcorp.insert({"_id": 8, "name": "mia", "results": [{ "evaluation": "term1", "score":47.463179736705023 },{ "evaluation": "term2", "score": 36.78273309957772 }
, { "evaluation": "term3", "score":45.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
```

Output:

```
> db.anandcorp.aggregate( [ { $unwind:"$results"}, { $match:{ "results.evaluation": "term1"} },{ $group: { _id: null, totalCount: { $sum:1}, failCount: { $sum:{ "$cond
": [ { "$lt": [ "$results.score", 37 ] }, 1, 0 ] } } } }, { $project: { _id: 0, totalCount: 1,failCount: 1, failPercent: { $multiply: [ { $divide: [ "$failCount", "$to
talCount" ] }, 50 ] } } } ] )
{ "totalCount" : 9, "failCount" : 8, "failPercent" : 44.44444444444444 }
> db.anandcorp.insert({"_id": 9, "name": "maya", "results": [{ "evaluation": "term1", "score":54.463179736705023 },{ "evaluation": "term2", "score": 36.78273309957772
}, { "evaluation": "term3", "score":45.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
```

```
> db.anandcorp.insert({"_id":10, "name": "raju", "results": [{ "evaluation": "term1", "score":35.463179736705023 },{ "evaluation": "term2", "score": 36.78273309957772
}, { "evaluation": "term3", "score":45.626176060654615 }]})
WriteResult({ "nInserted" : 1 })
```

Output:

```
> db.anandcorp.aggregate( [ { $unwind:"$results"}, { $match:{ "results.evaluation": "term1"} },{ $group: { _id: null, totalCount: { $sum:1}, failCount: { $sum:{ "$cond
": [ { "$lt": [ "$results.score", 37 ] }, 1, 0 ] } } } }, { $project: { _id: 0, totalCount: 1,failCount: 1, failPercent: { $multiply: [ { $divide: [ "$failCount", "$to
talCount" ] }, 100 ] } } } ] )
{ "totalCount" : 11, "failCount" : 9, "failPercent" : 81.81818181818183 }
```

EXPERIMENT 09

Aim

Execute below sets of problem by taking reference of Experimkent Number 08 and find out:

- Find employees who failed in aggregate (term1 + term2 + term3).
- Find the average score of trainees for term1.

Procedure

After inserting the specified data and other similar entries, here's how to perform the given commands:

1. To find employees who failed in aggregate

'**db.AnandCorp.aggregate([{"$project":{"overall_result":{"$arrayElemAt":[{"$filter" :{"input":"$results","as": "res", "cond": { "$eq": ["$$res", "evaluation"] } } }, 0] } } }, {"$project":{ "overall_fail": { "$lt": ["$overall_result.score", 37] } } }, { "$group": { "_id": null,"total": { "$sum": 1 }, "overall_fail_count": { "$sum": {"$cond": ["overall_fail", 1,0] } } } }]);'**.

2. To find the average score of trainees for term 1,

'**db.AnandCorp.aggregate([{ $unwind:"$results" }, { $match: { "results.evaluation": "term1" } }, { $group: { _id:"$results.evaluation", average_score: { $avg: "$results.score" } } }]);'**.

1 Output:

```
> db.anandcorp.aggregate( [{ "$project": { "overall_result": { "$arrayElemAt": [ { "$filter": { "input": "$results","as": "res", "cond": { "$eq": [ "$$res", "evaluatio
n" ] } } }, 0] } } }, {"$project":{ "overall_fail": { "$lt": [ "$overall_result.score", 37 ] } } }, { "$group": { "_id": null,"total": { "$sum": 1 }, "overall_fail_cou
nt": { "$sum": { "$cond": [ "overall_fail", 1,0 ] } } } } ] )
{ "_id" : null, "total" : 11, "overall_fail_count" : 11 }
```

2 Output:

```
> db.anandcorp.aggregate( [ { $unwind:"$results" }, { $match: { "results.evaluation": "term1" } }, { $group: { _id:"$results.evaluation", average_score: { $avg: "$resu
lts.score" } } } ] )
{ "_id" : "term1", "average_score" : 26.963179736705026 }
```

EXPERIMENT 10

Aim

Execute below sets of problem by taking reference of Experiment Number 08 and find out:

☐ Find the average score of trainees for aggregate (term1 + term2 + term3).
☐ Find number of employees who failed in all the three (term1 + term2 + term3).
☐ Find the number of employees who failed in any of the three (term1 + term2 + term3).

Procedure

After inserting the specified data and other similar entries, here's how to perform the given commands:

1. To find the average score of trainees for aggregate is

'db.AnandCorp.aggregate([{ $unwind:"$results" }, { $group: { _id: null, avgScore: { $avg: "$results.score" } } }]);'.

2. To find the numbers of employees who failed in all three evaluations,

'db.AnandCorp.aggregate([{ $unwind: "$results" }, { $match: { "results.score": { $lt: 37 } } }, { $group: { _id: "$name", numFails: { $sum: 1 } } }, { $match: { numFails: { $eq:3 } } }, { $group: { _id: null, totalNumEmpFailed: { $sum: 1 } } }]);'.

3. The numbers of employees who failed in any of the three evaluations,

'db.AnandCorp.aggregate([{ $unwind: "$results" }, { $match: { "results.score": { $lt: 37 } } }, { $group: { _id: "$name", numFails: { $sum: 1 } } }, { $match: { numFails: { $gt: 0 } } }, { $group: { _id: null, totalNumEmpFailed: { $sum: 1 } } }]);'.

1 Output:

```
> db.anandcorp.aggregate( [ { $unwind:"$results" }, { $group: { _id: null, avgScore: { $avg: "$results.score" } } } ] )
{ "_id" : null, "avgScore" : 24.87680741009023 }
```

2 Output:

```
> db.anandcorp.aggregate( [ { $unwind: "$results" }, { $match: { "results.score": { $lt:37 } } }, { $group: { _id: "$name", numFails: { $sum: 1 } } }, { $match: { numF
ails: { $eq:3 } } }, { $group: { _id: null, totalNumEmpFailed: { $sum: 1 } } } ] )
{ "_id" : null, "totalNumEmpFailed" : 7 }
```

3 Output:

```
> db.anandcorp.aggregate( [ { $unwind: "$results" }, { $match: { "results.score": { $lt:37 } } }, { $group: { _id: "$name", numFails: { $sum: 1 } } }, { $match: { numF
ails: { $gt:0 } } }, { $group: { _id: null, totalNumEmpFailed: { $sum: 1 } } } ] )
{ "_id" : null, "totalNumEmpFailed" : 11 }
```