**MongoDB Task**

1. Find all the topics and tasks which are thought in the month of October

db.topics.find({

date: {

$gte: new ISODate("01-10-2020"), // Start of October

$lt: new ISODate("30-10-2020") // Start of November

}

}, {

topic: 6,

});

db.tasks.find({

date: {

$gte: new ISODate("01-10-2020"), // Start of October

$lt: new ISODate("30-10-2020") // Start of November

}

}, {

task: 6,

});

2.Find all the company drives which appeared between 15 oct-2020 and 31-oct-2020

db.company\_drives.find({

date: {

$gte: new ISODate("2020-10-15T00:00:00Z"),

$lt: new ISODate("2020-11-01T00:00:00Z")

}

}, {

drive\_name: 6,

description: 1,

\_id: 0 });

3.Find all the company drives and students who are appeared for the placement

db.company\_drives.find(

{},

{ drive\_name: 1, date: 1 } // Only include drive\_name and date in the output

).pretty();

db.students.find(

{},

{ name: 1, email: 1, drives\_attended: 1 } // Include name, email, and drives\_attended in the output

).pretty();

[

{ "drive\_name": "Company A", "date": "2023-10-20T00:00:00Z" },

{ "drive\_name": "Company B", "date": "2023-10-22T00:00:00Z" }

]

[

{ "name": "John Doe", "email": "john.doe@example.com", "drives\_attended": [ObjectId("..."), ObjectId("...")] },

{ "name": "Jane Smith", "email": "jane.smith@example.com", "drives\_attended": [ObjectId("...")] }

]

4. Find the number of problems solved by the user in codekata

const userId = "user1234"; // Replace with the user ID you're interested in

const userSolvedCount = db.user\_solutions.aggregate([

{ $match: { user\_id: userId } }, // Find the specific user

{ $project: { number\_of\_problems\_solved: { $size: "$problems\_solved" } } } // Count the problems solved

]);

userSolvedCount.forEach(user => printjson(user));

{ "number\_of\_problems\_solved": }

5.Find all the mentors with who has the mentee's count more than 15

db.mentors.aggregate([

{

$project: {

mentor\_id: 1,

mentor\_name: 1,

mentee\_count: { $size: "$mentees" } // Count the number of mentees

}

},

{

$match: {

mentee\_count: { $gt: 15 } // Filter for mentees count greater than 15

}

}

]).pretty();

6. Find the number of users who are absent and task is not submitted between 15 oct-2020 and 31-oct-2020

db.user\_status.aggregate([

{

$match: {

is\_absent: true, // Filter for users who are absent

attendance\_date: {

$gte: ISODate("2020-10-15T00:00:00Z"), // Starting date

$lte: ISODate("2020-10-31T23:59:59Z") // Ending date

}

}

},

{

$group: {

\_id: null, // Grouping by null to get a total count

absent\_count: { $sum: 1 } // Count the absent users

}

}

]).pretty();

[

{

"\_id": null,

"absent\_count": 5 // Example number of absent users

}

]

db.user\_status.aggregate([

{

$match: {

task\_submitted: false, // Filter for users who did not submit tasks

attendance\_date: {

$gte: ISODate("2020-10-15T00:00:00Z"), // Starting date

$lte: ISODate("2020-10-31T23:59:59Z") // Ending date

}

}

},

{

$group: {

\_id: null, // Grouping by null to get a total count

not\_submitted\_count: { $sum: 1 } // Count users who did not submit tasks

}

}

]).pretty();

[

{

"\_id": null,

"not\_submitted\_count": 8 // Example number of users who did not submit tasks

}

]