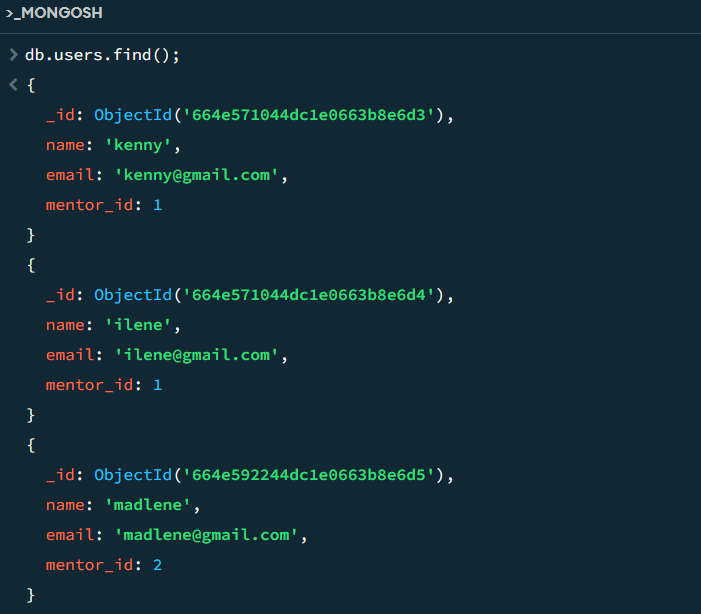
MongoDB Task-2

Zen class programme:

users:

db.createCollection(“users);





codekata:

db.createCollection(“codekata”);

db.codekata.insertMany([

{"\_id": ObjectId("664e571044dc1e0663b8e6d3"),"problems\_solved": 10, "date": ISODate("2020-10-20")},

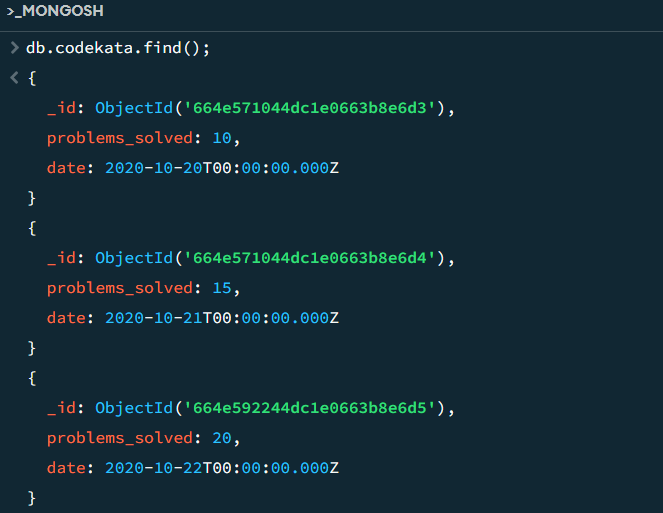
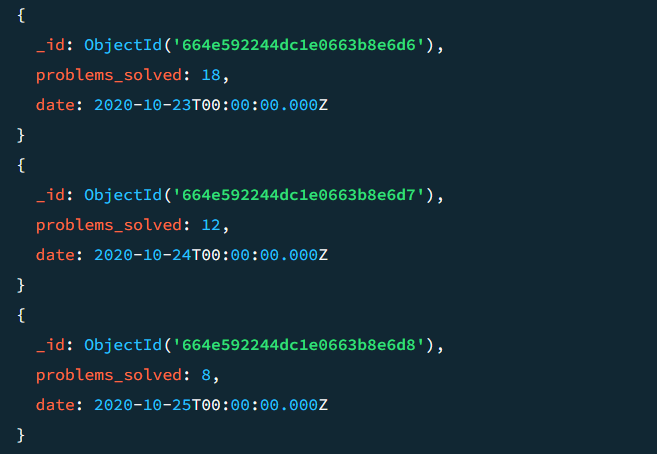
{"\_id": ObjectId("664e571044dc1e0663b8e6d4"),"problems\_solved": 15, "date": ISODate("2020-10-21")},

{"\_id": ObjectId("664e592244dc1e0663b8e6d5"),"problems\_solved": 20, "date": ISODate("2020-10-22")},

{"\_id": ObjectId("664e592244dc1e0663b8e6d6"),"problems\_solved": 18, "date": ISODate("2020-10-23")},

{"\_id": ObjectId("664e592244dc1e0663b8e6d7"),"problems\_solved": 12, "date": ISODate("2020-10-24")},

{"\_id": ObjectId("664e592244dc1e0663b8e6d8"),"problems\_solved": 8, "date": ISODate("2020-10-25")}

]);

attendance:

db.createCollection(“attendance”);

db.attendance.insertMany([

{"\_id": ObjectId("664e571044dc1e0663b8e6d3"),"date": ISODate("2020-10-20"),"status":"present" },

{"\_id": ObjectId("664e571044dc1e0663b8e6d4"),"date": ISODate("2020-10-21"),"status":"absent" },

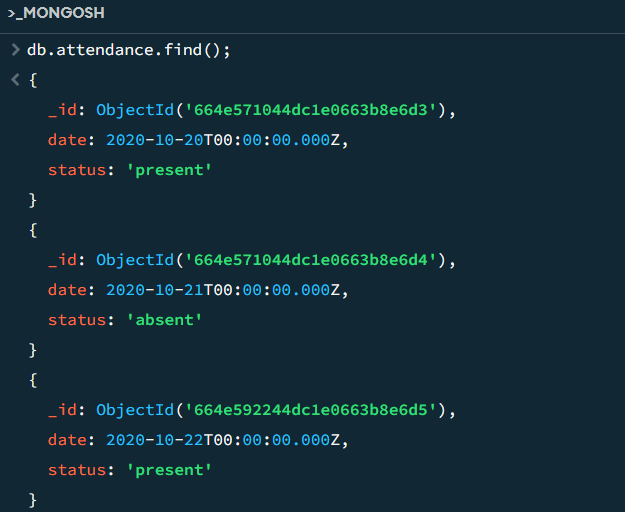
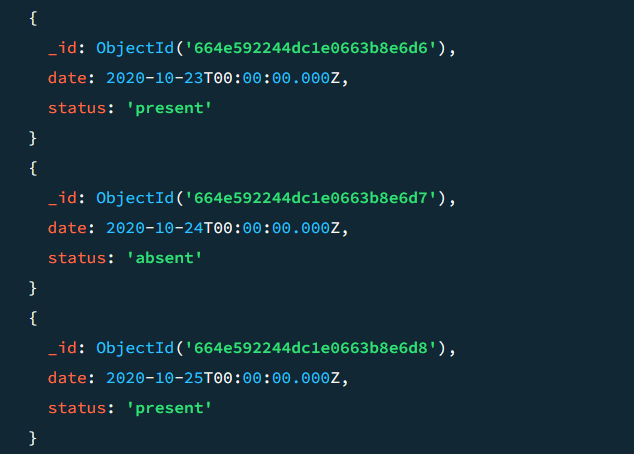
{"\_id": ObjectId("664e592244dc1e0663b8e6d5"),"date": ISODate("2020-10-22"),"status":"present" },

{"\_id": ObjectId("664e592244dc1e0663b8e6d6"),"date": ISODate("2020-10-23"),"status":"present" },

{"\_id": ObjectId("664e592244dc1e0663b8e6d7"),"date": ISODate("2020-10-24"),"status":"absent" },

{"\_id": ObjectId("664e592244dc1e0663b8e6d8"),"date": ISODate("2020-10-25"),"status":"present" }

]);

topics:

db.createCollection(“topics”);

db.topics.insertMany([

{ "title": "MongoDB", "date": ISODate("2020-10-01") },

{ "title": "Javascript", "date": ISODate("2020-10-10") },

{ "title": "React", "date": ISODate("2020-10-15") },

{ "title": "Fullstack Development", "date": ISODate("2020-10-20") },

{ "title": "SQL", "date": ISODate("2020-10-25") }

]);





tasks:

db.createCollection(“tasks”);

db.tasks.insertMany([

{ "title": "SQL", "date": ISODate("2020-10-25"), "submitted": true, "\_id": ObjectId("664e571044dc1e0663b8e6d3") },

{ "title": "Javascript", "date": ISODate("2020-10-10"), "submitted": false, "\_id": ObjectId("664e571044dc1e0663b8e6d4") },

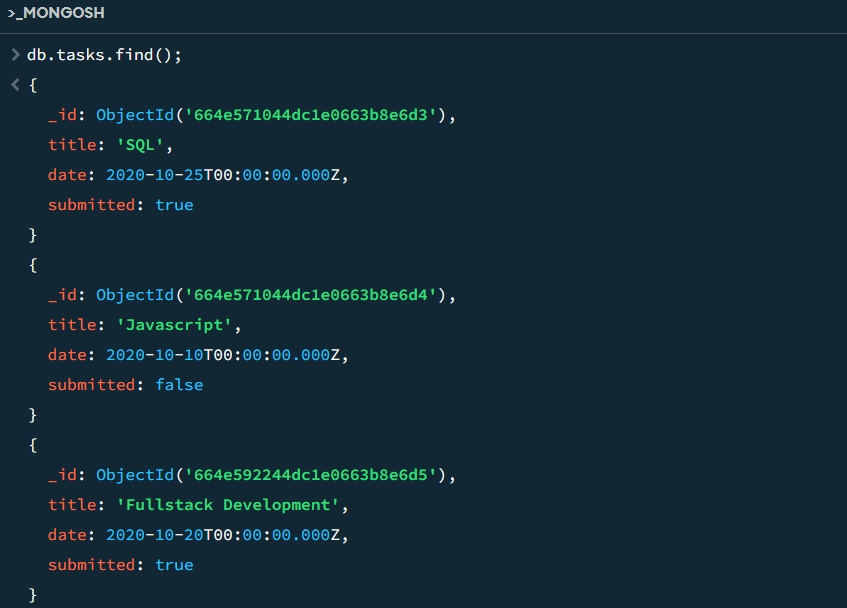
{ "title": "Fullstack Development", "date": ISODate("2020-10-20"), "submitted": true, "\_id": ObjectId("664e592244dc1e0663b8e6d5") },

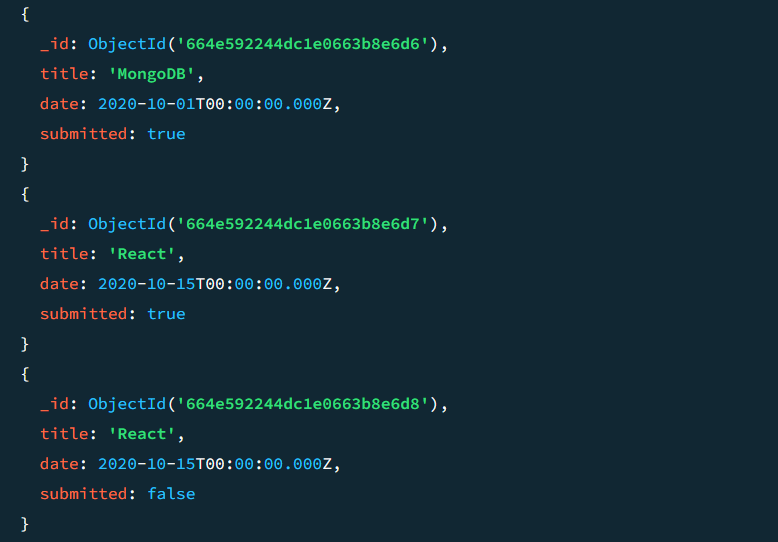
{ "title": "MongoDB", "date": ISODate("2020-10-01"), "submitted": true, "\_id": ObjectId("664e592244dc1e0663b8e6d6") },

{ "title": "React", "date": ISODate("2020-10-15"), "submitted": true, "\_id": ObjectId("664e592244dc1e0663b8e6d7") },

{ "title": "React", "date": ISODate("2020-10-15"), "submitted": false, "\_id": ObjectId("664e592244dc1e0663b8e6d8") }

]);





company\_drives:

db.createCollection(“company\_drives);

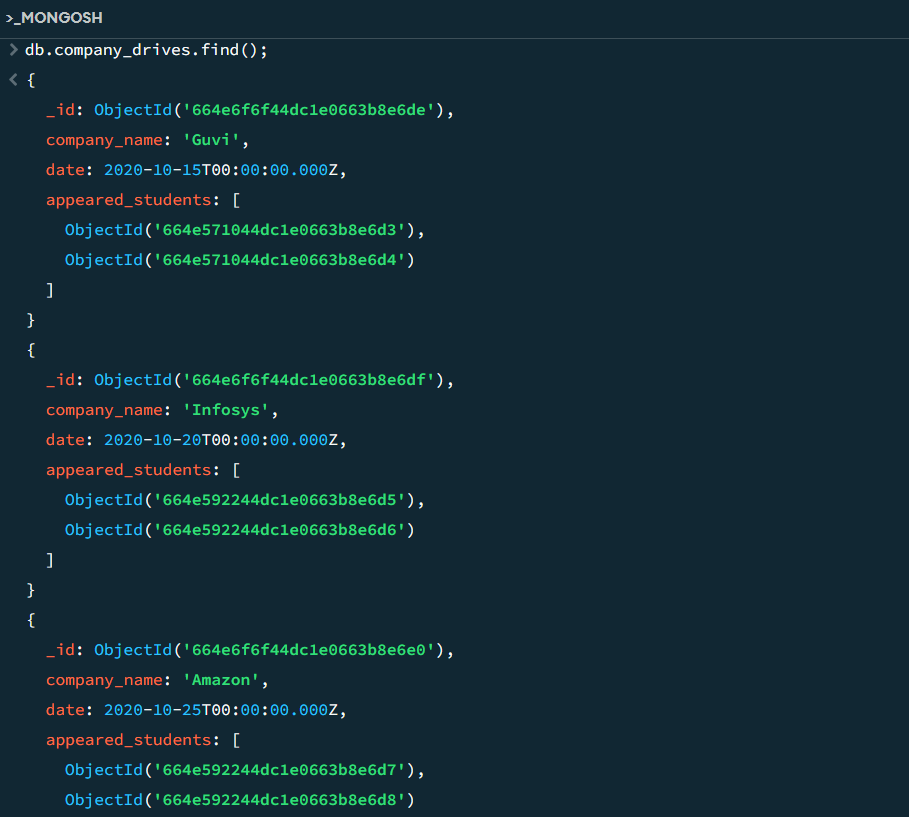
db.company\_drives.insertMany([

{ "company\_name":"Guvi", "date": ISODate("2020-10-15"), "appeared\_students":[ObjectId("664e571044dc1e0663b8e6d3"), ObjectId("664e571044dc1e0663b8e6d4")] },

{ "company\_name":"Infosys", "date": ISODate("2020-10-20"), "appeared\_students":[ObjectId("664e592244dc1e0663b8e6d5"), ObjectId("664e592244dc1e0663b8e6d6")] },

{ "company\_name":"Amazon", "date": ISODate("2020-10-25"), "appeared\_students":[ObjectId("664e592244dc1e0663b8e6d7"), ObjectId("664e592244dc1e0663b8e6d8")] }

]);



mentors:

db.createCollection(“mentors”);

db.mentors.insertMany([

{ "mentor\_id": 1,"mentor\_name": "clinton", "mentee\_count": 15 },

{ "mentor\_id": 2,"mentor\_name": "felene", "mentee\_count": 25 },

{ "mentor\_id": 3,"mentor\_name": "Ryan", "mentee\_count": 21 }

]);



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

db.topics.aggregate([

{

$match: {

date: {

$gte: ISODate("2020-10-01"),

$lte: ISODate("2020-10-31")

}

}

},

{

$lookup: {

from: "tasks",

let: { topicTitle: "$title", topicDate: "$date" },

pipeline: [

{

$match: {

$expr: {

$and: [

{ $eq: ["$title", "$$topicTitle"] },

{ $eq: ["$date", "$$topicDate"] }

]

}

}

},

{

$project: {

\_id: 1,

title: 1,

date: 1,

submitted: 1

}

}

],

as: "tasks"

}

},

{

$project: {

\_id: 0,

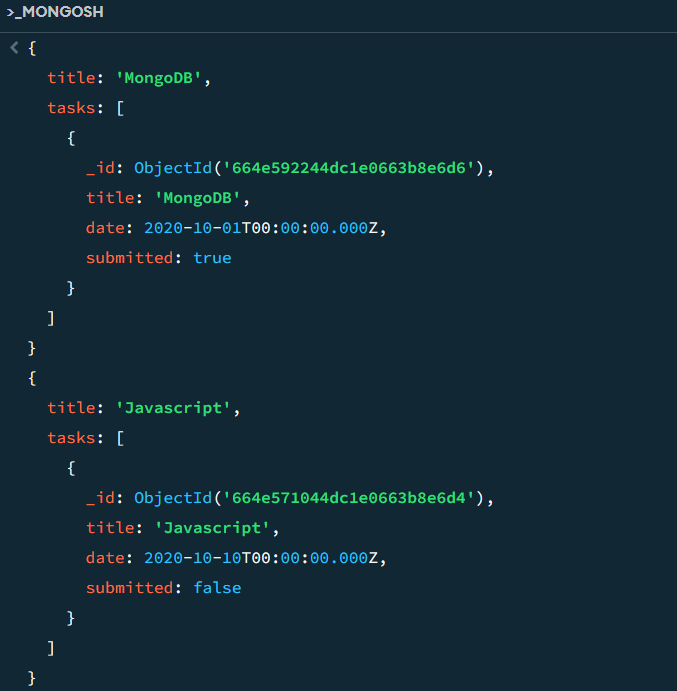
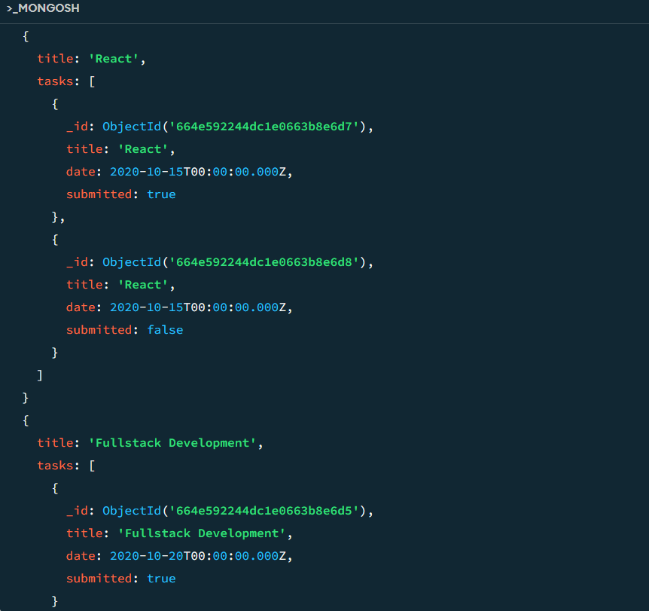
title: 1,

tasks: 1

}

}

]).pretty();



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

db.company\_drives.find({

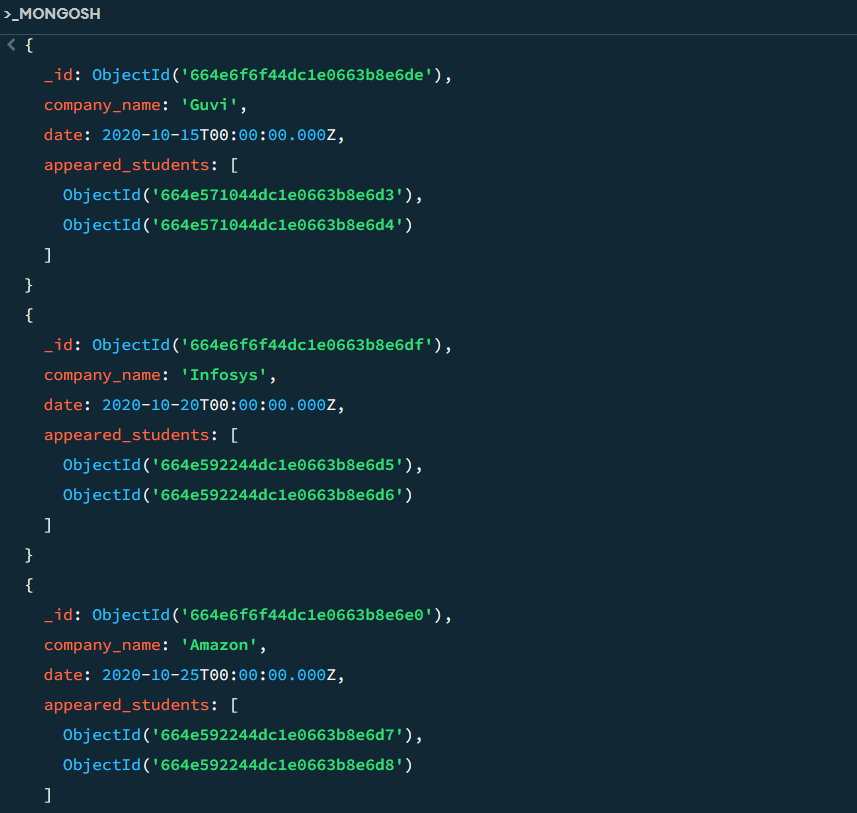
date: {

$gte: ISODate("2020-10-15"),

$lte: ISODate("2020-10-31")

}

}).pretty();



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

db.company\_drives.aggregate([

{

$lookup: {

from: "users",

localField: "appeared\_students",

foreignField: "\_id",

as: "appeared\_students\_info"

}

},

{

$project: {

company\_name: 1,

appeared\_students\_info: {

\_id: 1,

name: 1,

email: 1,

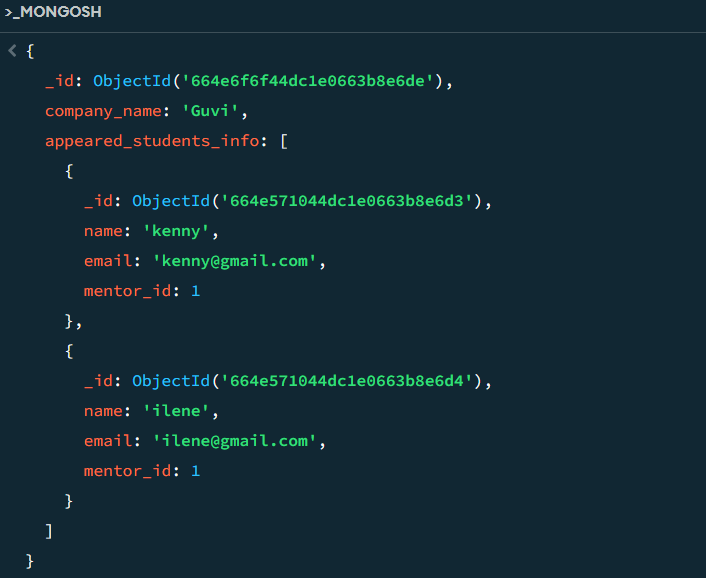
mentor\_id: 1

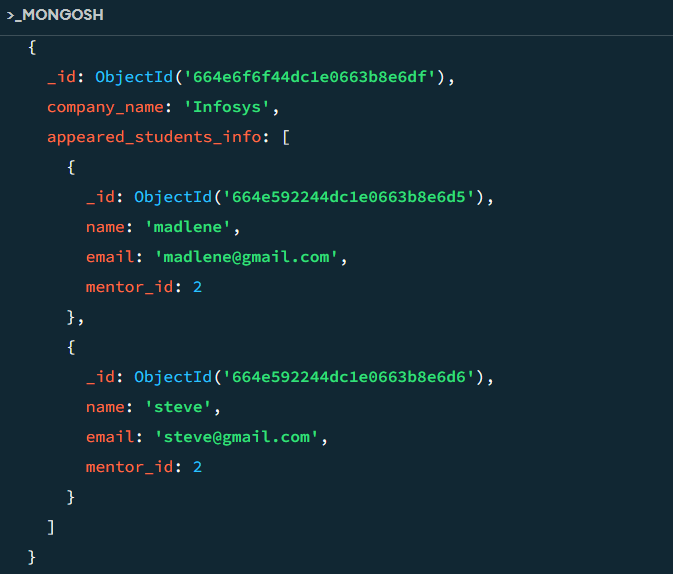
}

}

}

]).pretty();





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

db.codekata.aggregate([

{

$group: {

\_id: "$\_id",

total\_problems\_solved: { $sum: "$problems\_solved" }

}

}

]).pretty();



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

db.mentors.find({ mentee\_count: { $gt: 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.attendance.aggregate([

{

$match: {

date: {

$gte: ISODate("2020-10-15"),

$lte: ISODate("2020-10-31")

},

status: "absent"

}

},

{

$lookup: {

from: "tasks",

localField: "\_id",

foreignField: "\_id",

as: "tasks"

}

},

{

$unwind: "$tasks"

},

{

$match: {

"tasks.submitted": false

}

},

{

$group: {

\_id: null,

count: { $sum: 1 }

}

},

{

$project: {

\_id: 0,

num\_absent\_and\_not\_submitted: "$count"

}

}

]);

