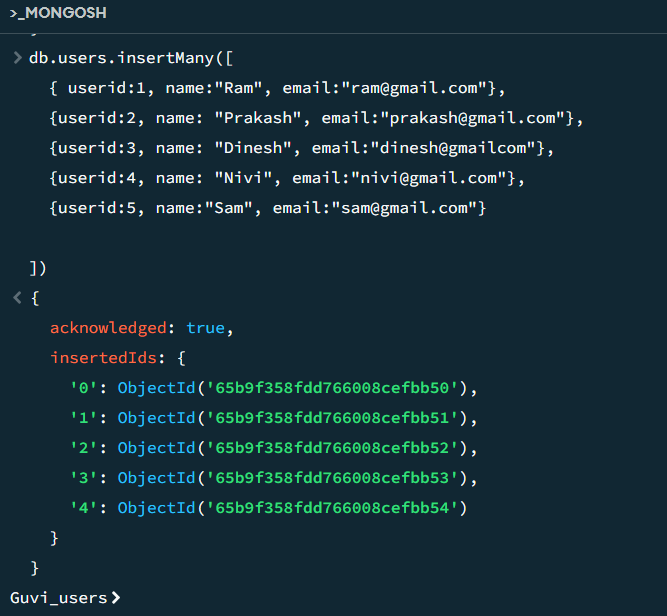
**MONGODB TASK 2**

Design Database for zen class Programmme

**Create database for Guvi\_users:**



db.users.insertMany([

{ userid:1, name:"Ram", email:"ram@gmail.com"},

{userid:2, name: "Prakash", email:"prakash@gmail.com"},

{userid:3, name: "Dinesh", email:"dinesh@gmailcom"},

{userid:4, name: "Nivi", email:"nivi@gmail.com"},

{userid:5, name:"Sam", email:"sam@gmail.com"}

])

**Create collection and insert data :”Codekata”**

db.codekata.insertMany([

{userid:1, problems:50},

{userid:2, problems:60},

{userid:3, problems:90},

{userid:4, problems:70},

{userid:5, problems:80}

])



**Create collection and insert data :”Attendance”**

db.attendance.insertMany([

{userid:1, topicid:2, attended: true},

{userid:2, topicid:1, attended:true},

{userid:3, topicid:5, attended:true},

{userid:4, topicid:3, attended:true},

{userid:5, topicid:4, attended:false}

])



**Create collection and insert data: “Topics”**

db.topics.insertMany([

{topicid:1, topic:"HTML", topic\_date: new Date("18-Oct-2020")},

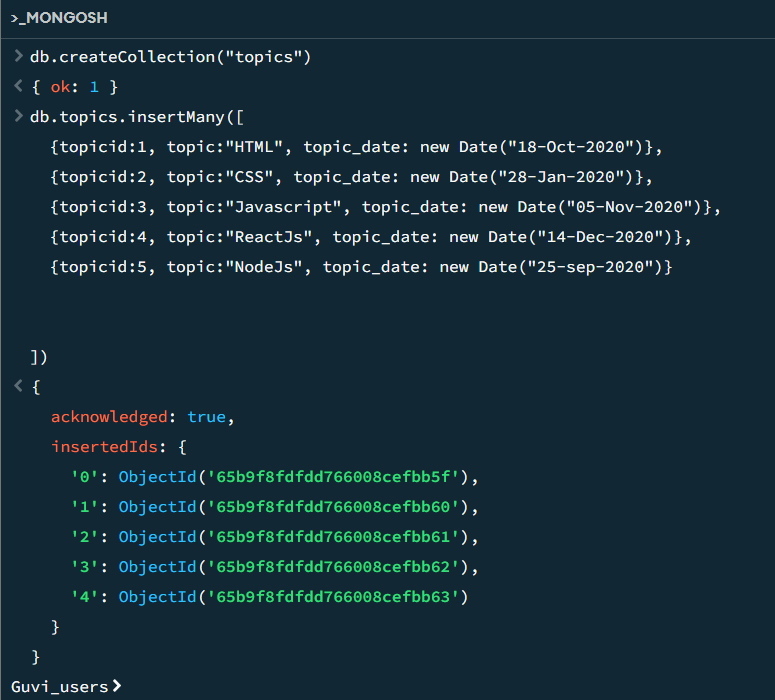
{topicid:2, topic:"CSS", topic\_date: new Date("28-Jan-2020")},

{topicid:3, topic:"Javascript", topic\_date: new Date("05-Nov-2020")},

{topicid:4, topic:"ReactJs", topic\_date: new Date("14-Dec-2020")},

{topicid:5, topic:"NodeJs", topic\_date: new Date("25-sep-2020")}

])



**Create collection and insert data: “Tasks”**

db.tasks.insertMany([

{taskid:1, topicid:1, userid:1, task:"HTML Task", due\_date: new Date("18-Oct-2020"), submitted: true},

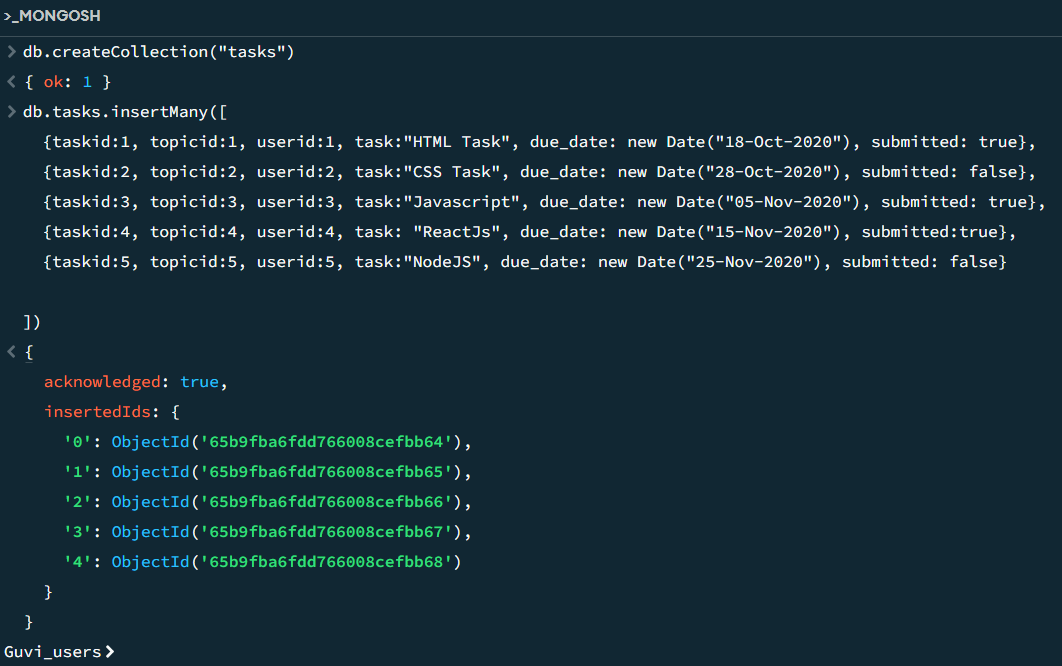
{taskid:2, topicid:2, userid:2, task:"CSS Task", due\_date: new Date("28-Oct-2020"), submitted: false},

{taskid:3, topicid:3, userid:3, task:"Javascript", due\_date: new Date("05-Nov-2020"), submitted: true},

{taskid:4, topicid:4, userid:4, task: "ReactJs", due\_date: new Date("15-Nov-2020"), submitted:true},

{taskid:5, topicid:5, userid:5, task:"NodeJS", due\_date: new Date("25-Nov-2020"), submitted: false}

])



**Create collection and insert data:”CompanyDrives”**

db.companydrives.insertMany([

{userid:1, drive\_date: new Date("20-Oct-2020"), company:"Apple"},

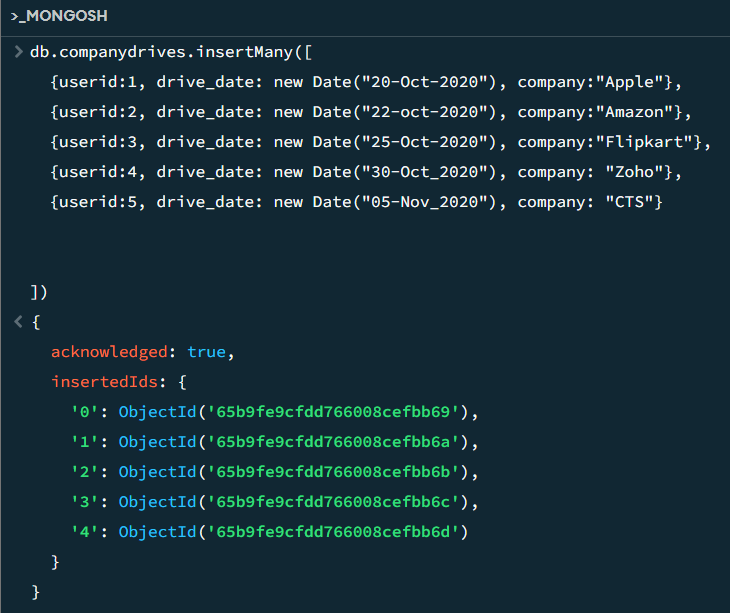
{userid:2, drive\_date: new Date("22-oct-2020"), company:"Amazon"},

{userid:3, drive\_date: new Date("25-Oct-2020"), company:"Flipkart"},

{userid:4, drive\_date: new Date("30-Oct\_2020"), company: "Zoho"},

{userid:5, drive\_date: new Date("05-Nov\_2020"), company: "CTS"}

])



**Create collections and insert data:”Mentors”**

db.mentors.insertMany([

{mentorid:1, mentorname: "Rupan", mentor\_email: "rupan@gmail.com", mentee\_count: 20},

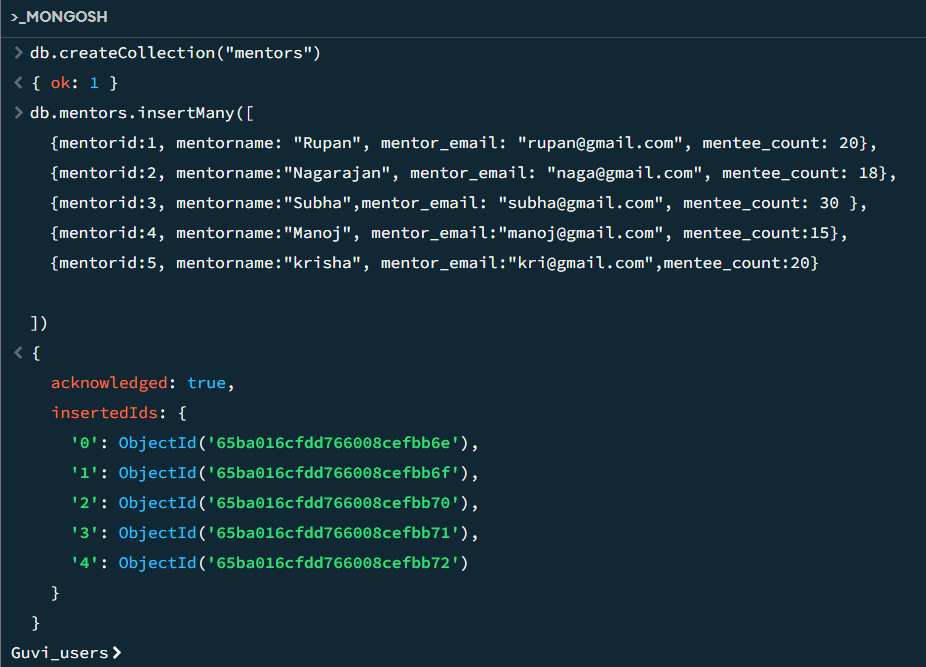
{mentorid:2, mentorname:"Nagarajan", mentor\_email: "naga@gmail.com", mentee\_count: 18},

{mentorid:3, mentorname:"Subha",mentor\_email: "subha@gmail.com", mentee\_count: 30 },

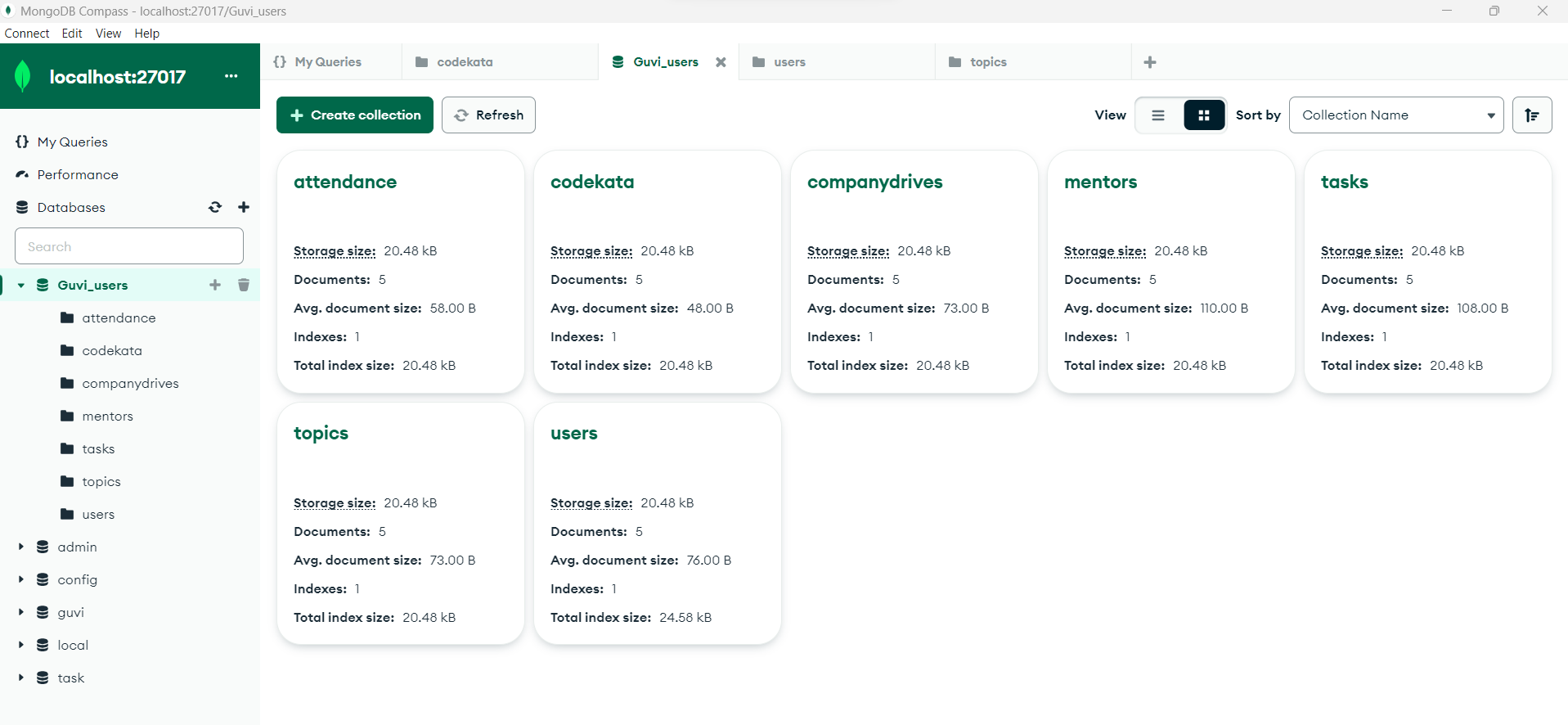
{mentorid:4, mentorname:"Manoj", mentor\_email:"manoj@gmail.com", mentee\_count:15},

{mentorid:5, mentorname:"krisha", mentor\_email:"kri@gmail.com",mentee\_count:20}

])



CREATED DATABASE



**Task Solutions:**

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

db.topics.aggregate([

{

$lookup: {

from: "tasks",

localField: "topicid",

foreignField: "topicid",

as: "taskinfo"

}

},

{

$match: {

$and: [

{ topic\_date: { $gte: new Date("2020-10-01"), $lt: new Date("2020-11-01") } },

{

$or: [

{ "taskinfo.due\_date": { $gte: new Date("2020-10-01"), $lt: new Date("2020-11-01") } },

{ "taskinfo.due\_date": { $exists: false } }

]

}

]

}

},

{

$project: {

\_id: 0,

topicid: 1,

topic: 1,

topic\_date: 1,

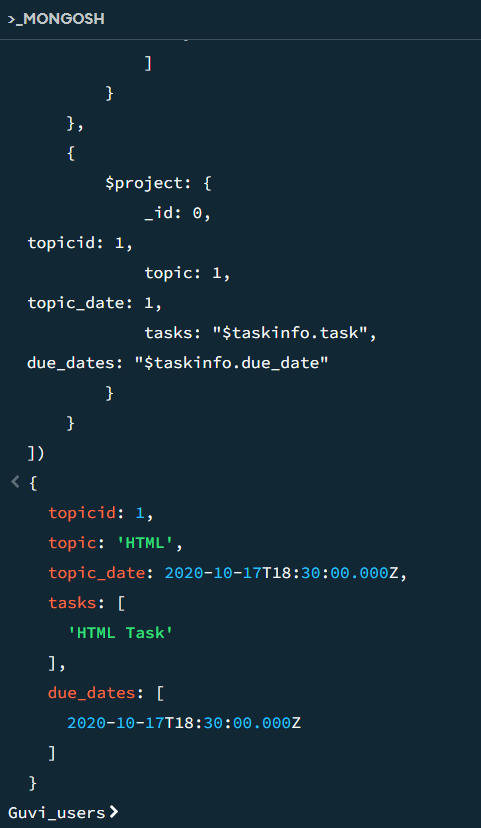
tasks: "$taskinfo.task",

due\_dates: "$taskinfo.due\_date"

}

}

])



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

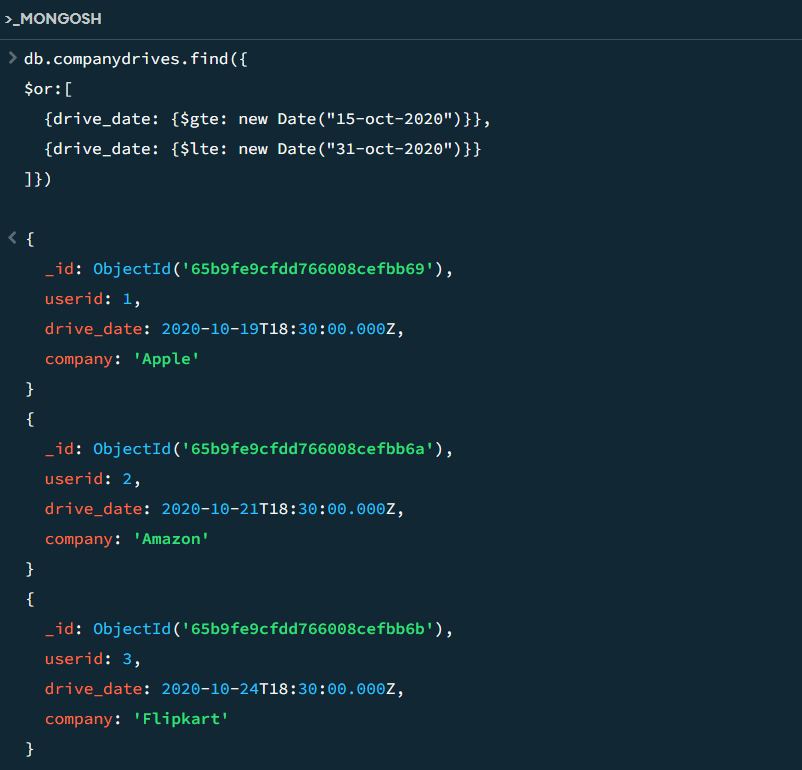
db.companydrives.find({

$or:[

{drive\_date: {$gte: new Date("15-oct-2020")}},

{drive\_date: {$lte: new Date("31-oct-2020")}}

]})



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

db.companydrives.aggregate([

{

$lookup:{

from:"users",

localField:"userid",

foreignField:"userid",

as: "userinfo"

}

},

{

$project:{

\_id:0,

company:1,

drive\_date:1,

students:"$userinfo"}}

])



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

db.codekata.aggregate([

{

$lookup:{

from:"users",

localField:"userid",

foreignField:"userid",

as:"userinfo"

}},

{

$group:{

\_id:{

userid:"$userid",

username:"$userinfo.name"

},

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

}},

{

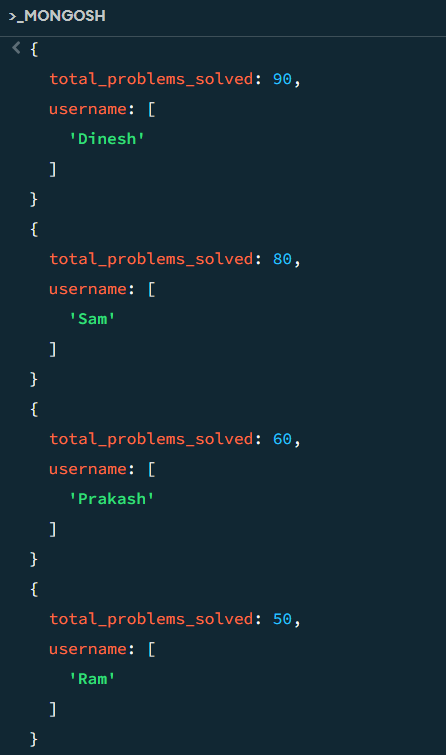
$project:{

\_id:0,

userid:"$\_id:userid",

username:"$\_id.username",

total\_problems\_solved:1

}}])  


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

db.users.aggregate([

{

$match:{mentorid:{$exists:true}}

},

{

$group:{

\_id:"$mentorid",

mentorname:{$first: "$mentorname"},

mentee\_count:{$sum:1}

}},

{

$match:{mentee\_count:{$gt:15}}

},

{

$project:{

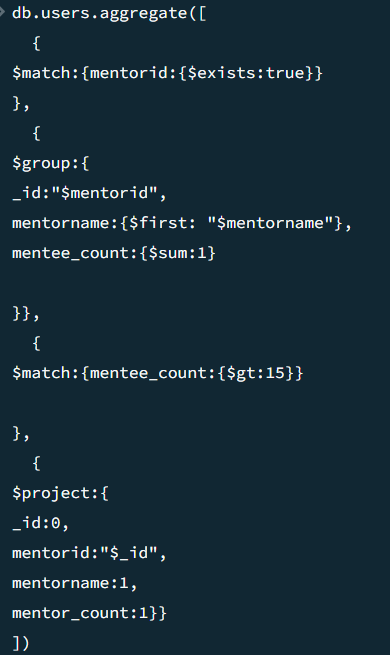
\_id:0,

mentorid:"$\_id",

mentorname:1,

mentor\_count:1}}

])



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

