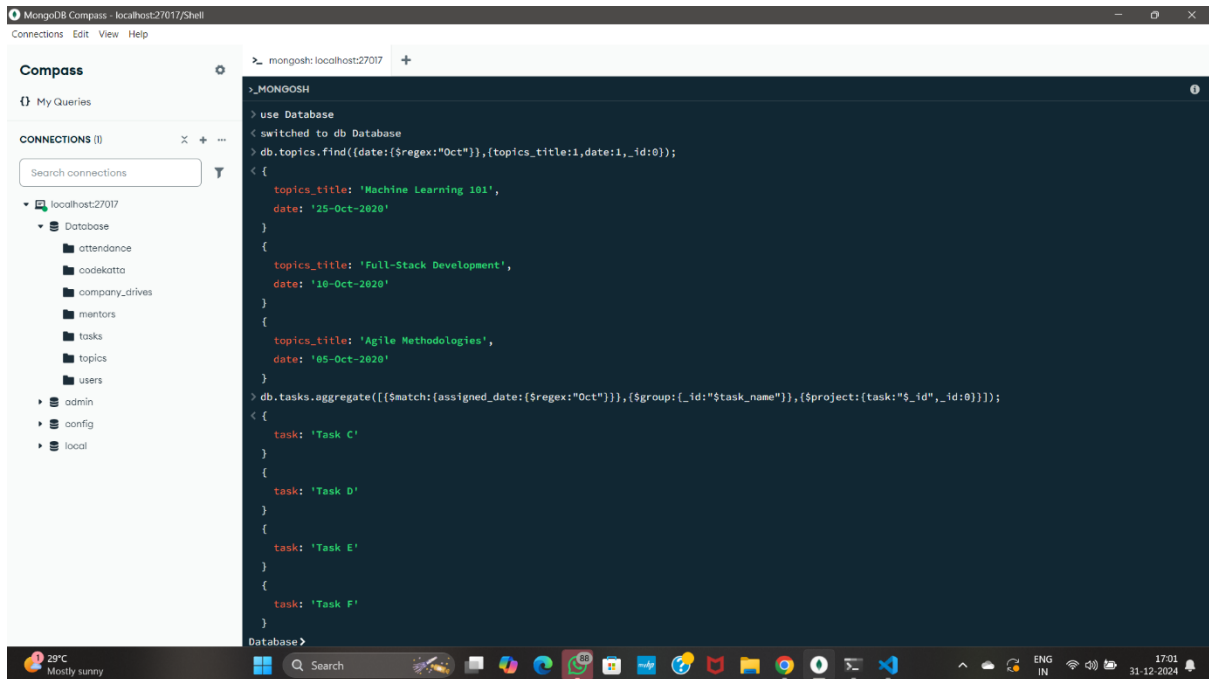


MongoDB – Database Design

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

- `db.topics.find({date:{$regex:"Oct"}},{topics_title:1,date:1,_id:0});`
- `db.tasks.aggregate([{$match:{assigned_date:{$regex:"Oct"}}},{ $group:{_id:"$task_name"}},{ $project:{task:"$ _id",_id:0}}]);`

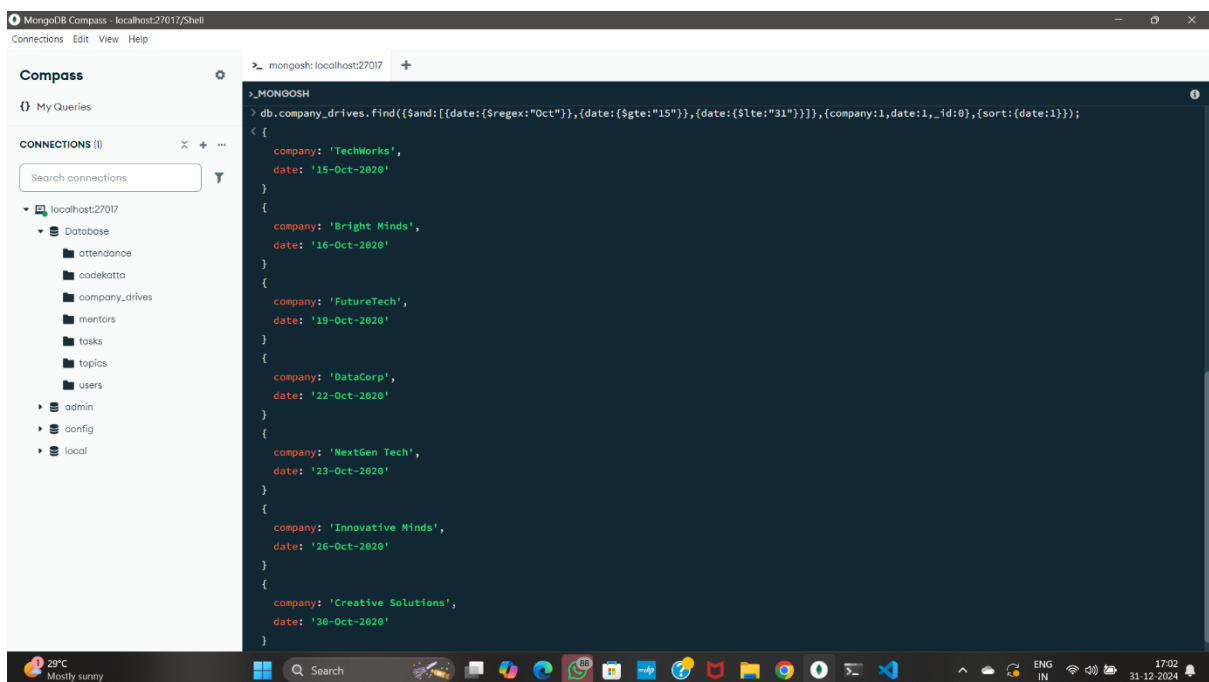


The screenshot shows the MongoDB Compass interface. On the left, the 'CONNECTIONS' panel lists the 'localhost:27017' connection with a tree view of databases including 'attendance', 'codekatta', 'company_drives', 'mentors', 'tasks', 'topics', 'users', 'admin', 'config', and 'local'. The main panel displays a MongoDB shell session with the following commands and results:

```
> use Database
< switched to db Database
> db.topics.find({date:{$regex:"Oct"}},{topics_title:1,date:1,_id:0});
< [
  {
    topics_title: 'Machine Learning 101',
    date: '25-Oct-2020'
  },
  {
    topics_title: 'Full-Stack Development',
    date: '10-Oct-2020'
  },
  {
    topics_title: 'Agile Methodologies',
    date: '05-Oct-2020'
  }
]
> db.tasks.aggregate([{$match:{assigned_date:{$regex:"Oct"}}},{ $group:{_id:"$task_name"}},{ $project:{task:"$ _id",_id:0}}]);
< [
  {
    task: 'Task C'
  },
  {
    task: 'Task D'
  },
  {
    task: 'Task E'
  },
  {
    task: 'Task F'
  }
]
```

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

- `db.company_drives.find({$and:[{date:{$regex:"Oct"}},{date:{$gte:"15"}},{date:{$lte:"31"}}]},{company:1,date:1,_id:0},{sort:{date:1}});`



The screenshot shows the MongoDB Compass interface. On the left, the 'CONNECTIONS' panel lists the 'localhost:27017' connection with a tree view of databases including 'attendance', 'codekatta', 'company_drives', 'mentors', 'tasks', 'topics', 'users', 'admin', 'config', and 'local'. The main panel displays a MongoDB shell session with the following command and results:

```
> db.company_drives.find({$and:[{date:{$regex:"Oct"}},{date:{$gte:"15"}},{date:{$lte:"31"}}]},{company:1,date:1,_id:0},{sort:{date:1}});
< [
  {
    company: 'TechWorks',
    date: '15-Oct-2020'
  },
  {
    company: 'Bright Minds',
    date: '16-Oct-2020'
  },
  {
    company: 'FutureTech',
    date: '19-Oct-2020'
  },
  {
    company: 'DataCorp',
    date: '22-Oct-2020'
  },
  {
    company: 'NextGen Tech',
    date: '23-Oct-2020'
  },
  {
    company: 'Innovative Minds',
    date: '26-Oct-2020'
  },
  {
    company: 'Creative Solutions',
    date: '30-Oct-2020'
  }
]
```

- Find all the company drives and students who are appeared for the placement.
 - `db.company_drives.find({}, {company:1,students_appeared:1,_id:0}).toArray();`

```

> db.company_drives.find({}, {company:1,students_appeared:1,_id:0}).toArray();
[
  {
    company: 'TechCorp',
    students_appeared: [ 'John Doe', 'Jane Smith', 'Michael Brown' ]
  },
  {
    company: 'Innovate Ltd.',
    students_appeared: [ 'Emily White', 'Chris Green', 'Laura Wilson' ]
  },
  {
    company: 'GlobalTech',
    students_appeared: [ 'David Taylor', 'Sophia Martinez', 'James Anderson' ]
  },
  {
    company: 'NextGen Solutions',
    students_appeared: [ 'Olivia Thomas', 'Daniel Harris', 'Mia Clark' ]
  },
  {
    company: 'TechWorks',
    students_appeared: [ 'Ethan Rodriguez', 'Ava Lewis', 'Noah Walker' ]
  },
  {
    company: 'FutureTech',
    students_appeared: [ 'Lily Young', 'Benjamin King', 'Isabella Wright' ]
  },
  {
    company: 'DataCorp',
    students_appeared: [ 'Charlotte Baker', 'John Doe', 'Jane Smith' ]
  }
]

```

- Find the number of problems solved by the user in codekata
 - `db.codekata.aggregate([{$match:{isCompleted:true}},{$group:{_id:"$userId",problemsSolved:{$sum:1}}},{ $project:{userId:"$ _id",problemsSolved:1,_id:0}}]);`

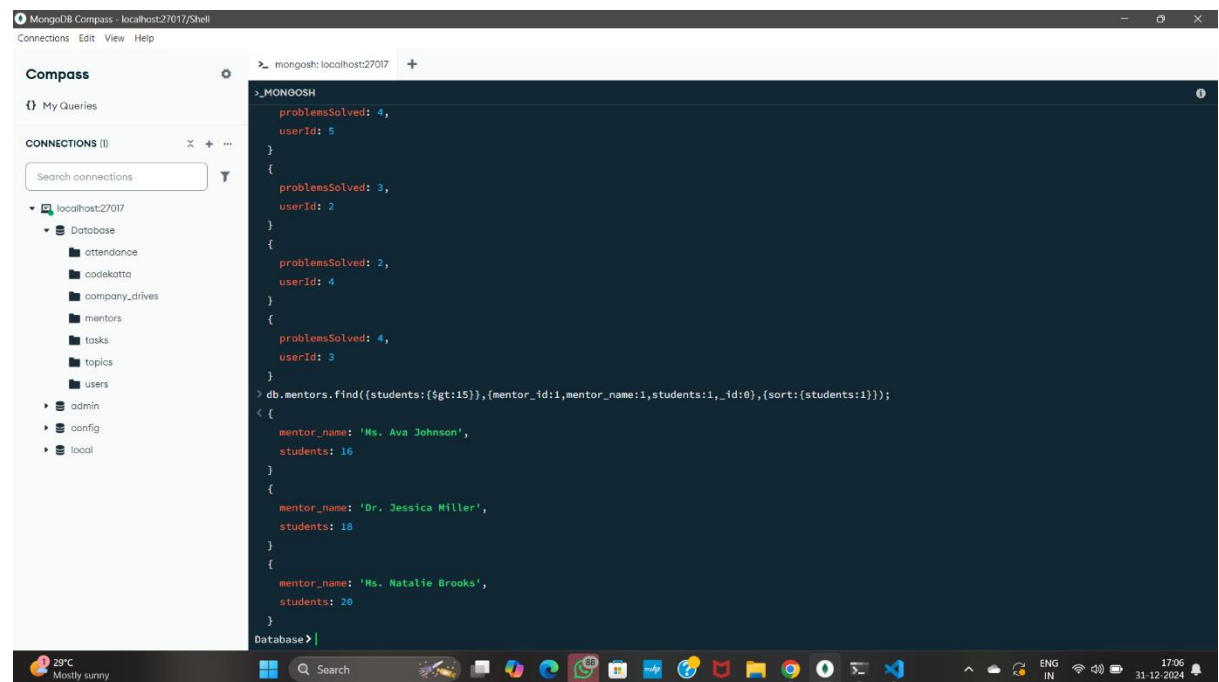
```

> db.codekata.aggregate([{$match:{isCompleted:true}},{$group:{_id:"$userId",problemsSolved:{$sum:1}}},{ $project:{userId:"$ _id",problemsSolved:1,_id:0}}]);
[
  {
    problemsSolved: 3,
    userId: 1
  },
  {
    problemsSolved: 4,
    userId: 5
  },
  {
    problemsSolved: 3,
    userId: 2
  },
  {
    problemsSolved: 2,
    userId: 4
  },
  {
    problemsSolved: 4,
    userId: 3
  }
]

```

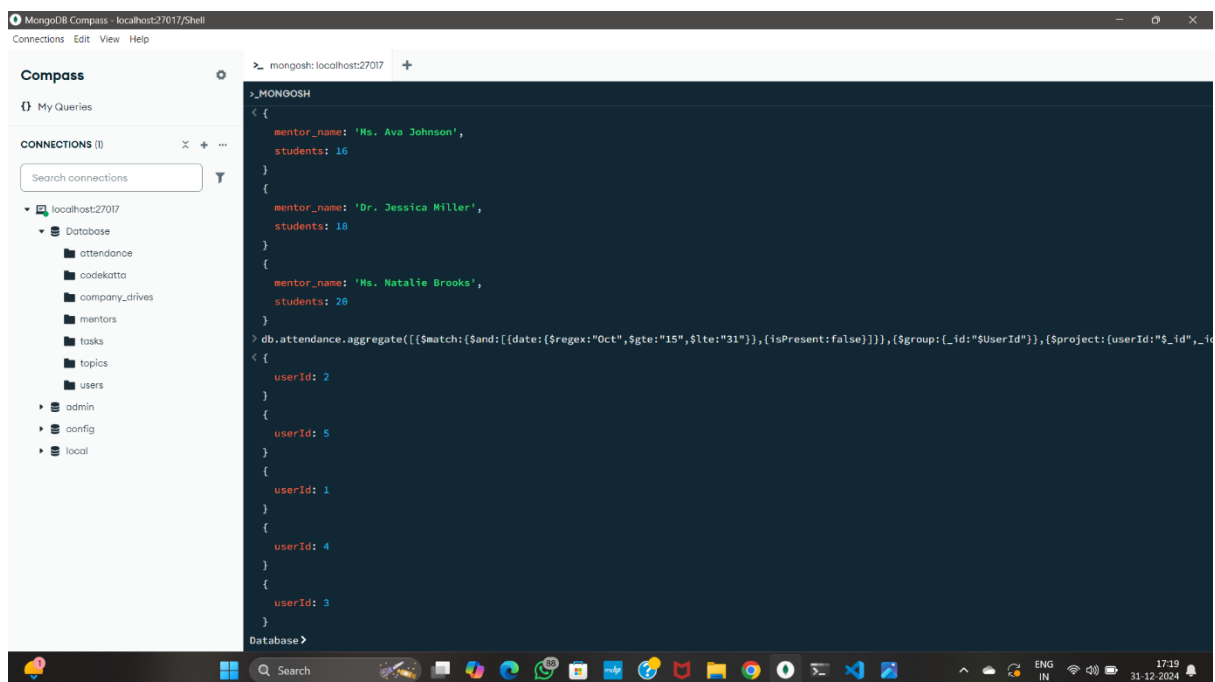
5. Find all the mentors with who has the mentee's count more than 15

- `db.mentors.find({students:{>15}},{mentor_id:1,mentor_name:1,students:1,_id:0},{sort:{students:1}});`

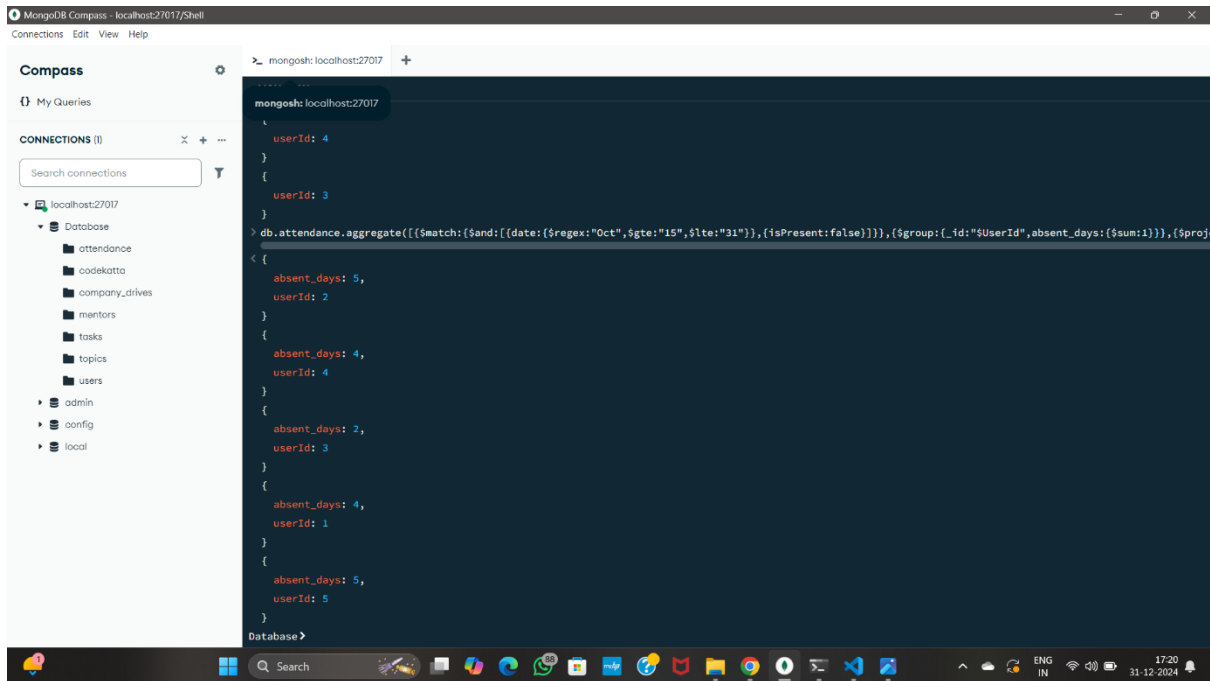


6. Find the number of users who are absent and task is not submitted between 15 oct-2020 and 31-oct-2020

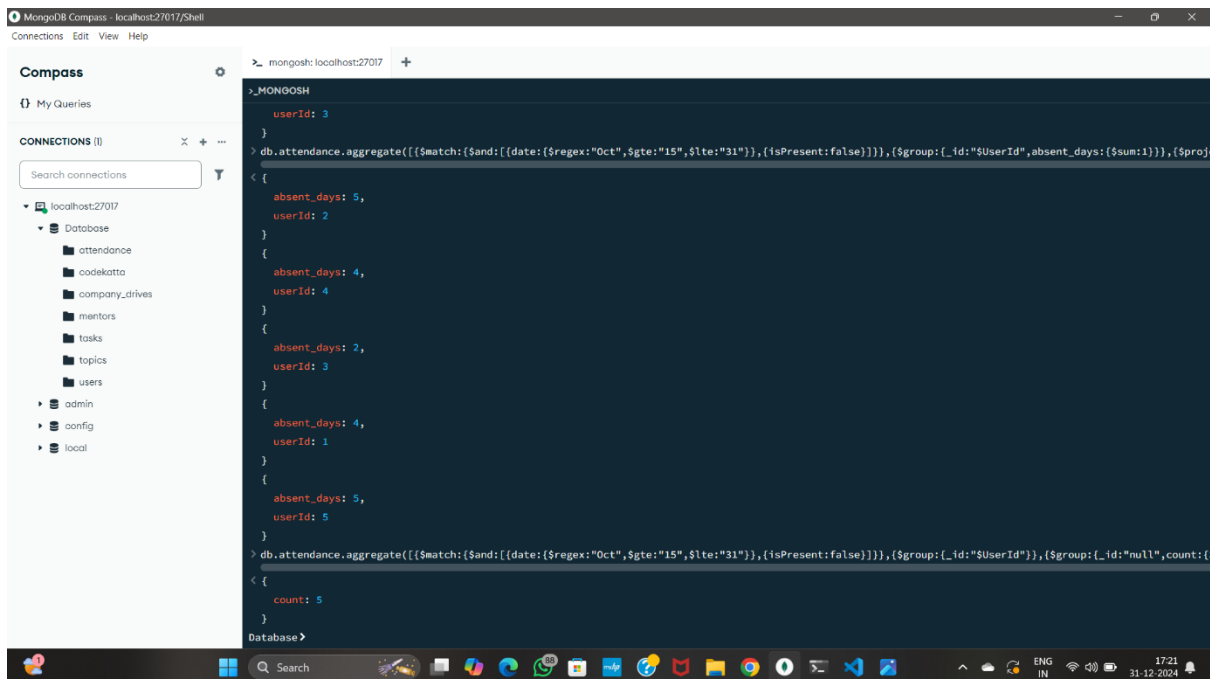
- To find list of users absent between between 15 oct-2020 and 31-oct-2020
 - ✓ `db.attendance.aggregate([{$match:{$and:{{date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isPresent:false}}}},{$group:{_id:"$UserId"}},{project:{userId:"$ _id",_id:0}}]);`



- To find absenteeism for each user between 15 oct-2020 and 31-oct-2020
 - ✓ `db.attendance.aggregate([{$match:{$and:[{date:{$regex:"Oct",$gte:"15"},$lte:"31"}]},isPresent:false}}],{$group:{$_id:"$UserId",absent_days:{$sum:1}}},{$project:{userId:"$_id",absent_days:1,_id:0}}]);`



- To find number of users absent between 15 oct-2020 and 31-oct-2020 - without repetition Ex: considering UserId only
 - ✓ `db.attendance.aggregate([{$match:{$and:[{date:{$regex:"Oct",$gte:"15"},$lte:"31"}]},isPresent:false}}],{$group:{$_id:"$UserId"}},{$group:{$_id:"null",count:{$sum:1}}},{$project:{count:1,_id:0}}]);`



- To find number of users absent between 15 oct-2020 and 31-oct-2020 - with repitition
Ex: UserID-1 is absent for 4 days + UserID-2 is absent for 5 days + UserID-3 is absent for 2 days + ...

✓ `db.attendance.find({$and:[{date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isPresent:false}]}).count();`

The screenshot shows the MongoDB Compass interface. On the left, the 'Database' dropdown is set to 'attendance'. The main panel displays the results of a query in the 'Database' tab. The query is:

```
db.attendance.aggregate([{$match:{$and:[{date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isPresent:false}]}],{$group:{_id:"$UserID",absent_days:{$sum:1}}},{$project:{absent_days:1,_id:0}}])
```

The results show 5 documents, each representing a user and their total absent days:

absent_days	userId
5	2
4	4
2	3
4	1
5	5

Below the results, the count of documents is shown as 5.

- To find list of user who did not submit task between 15 oct-2020 and 31-oct-2020
✓ `db.tasks.aggregate([{$match:{$and:[{due_date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isSubmitted:false}]}],{$group:{_id:"$UserID"}},{$project:{userId:"$_id",_id:0}}]);`

The screenshot shows the MongoDB Compass interface. On the left, the 'Database' dropdown is set to 'tasks'. The main panel displays the results of a query in the 'Database' tab. The query is:

```
db.tasks.aggregate([{$match:{$and:[{due_date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isSubmitted:false}]}],{$group:{_id:"$UserID"}},{$project:{userId:"$_id",_id:0}}])
```

The results show 5 documents, each representing a user and their total absent days:

absent_days	userId
4	4
2	3
4	1
5	5
5	2

Below the results, the count of documents is shown as 5.

- To find not submitted task list for every user between 15 oct-2020 and 31-oct-2020
 - ✓ `db.tasks.aggregate([{$match:{$and:[{due_date:{$regex:"Oct",$gte:"15",$lte:"31"}},{IsSubmitted:false}]}],{$group:{$_id:"$UserID",notSubmittedTask:{$sum:1}}},{$project:{userId:"$_id",notSubmittedTask:1,_id:0}}]);`

```

> db.attendance.aggregate([{$match:{$and:[{date:{$regex:"Oct",$gte:"15",$lte:"31"}},{isPresent:false}]}],{$group:{$_id:"$UserID"}},{$group:{$_id:"null",count:{$sum:1}}},{$project:{count:1,_id:0}}]);
{
  count: 5
}

```

- To find number of user who did not submit task between 15 oct-2020 and 31-oct-2020
 - ✓ `db.tasks.aggregate([{$match:{$and:[{due_date:{$regex:"Oct",$gte:"15",$lte:"31"}},{IsSubmitted:false}]}],{$group:{$_id:"$UserID"}},{$group:{$_id:"null",count:{$sum:1}}},{$project:{count:1,_id:0}}]);`

```

> db.tasks.aggregate([{$match:{$and:[{due_date:{$regex:"Oct",$gte:"15",$lte:"31"}},{IsSubmitted:false}]}],{$group:{$_id:"$UserID"}},{$group:{$_id:"null",count:{$sum:1}}},{$project:{count:1,_id:0}}]);
{
  count: 1
}

```