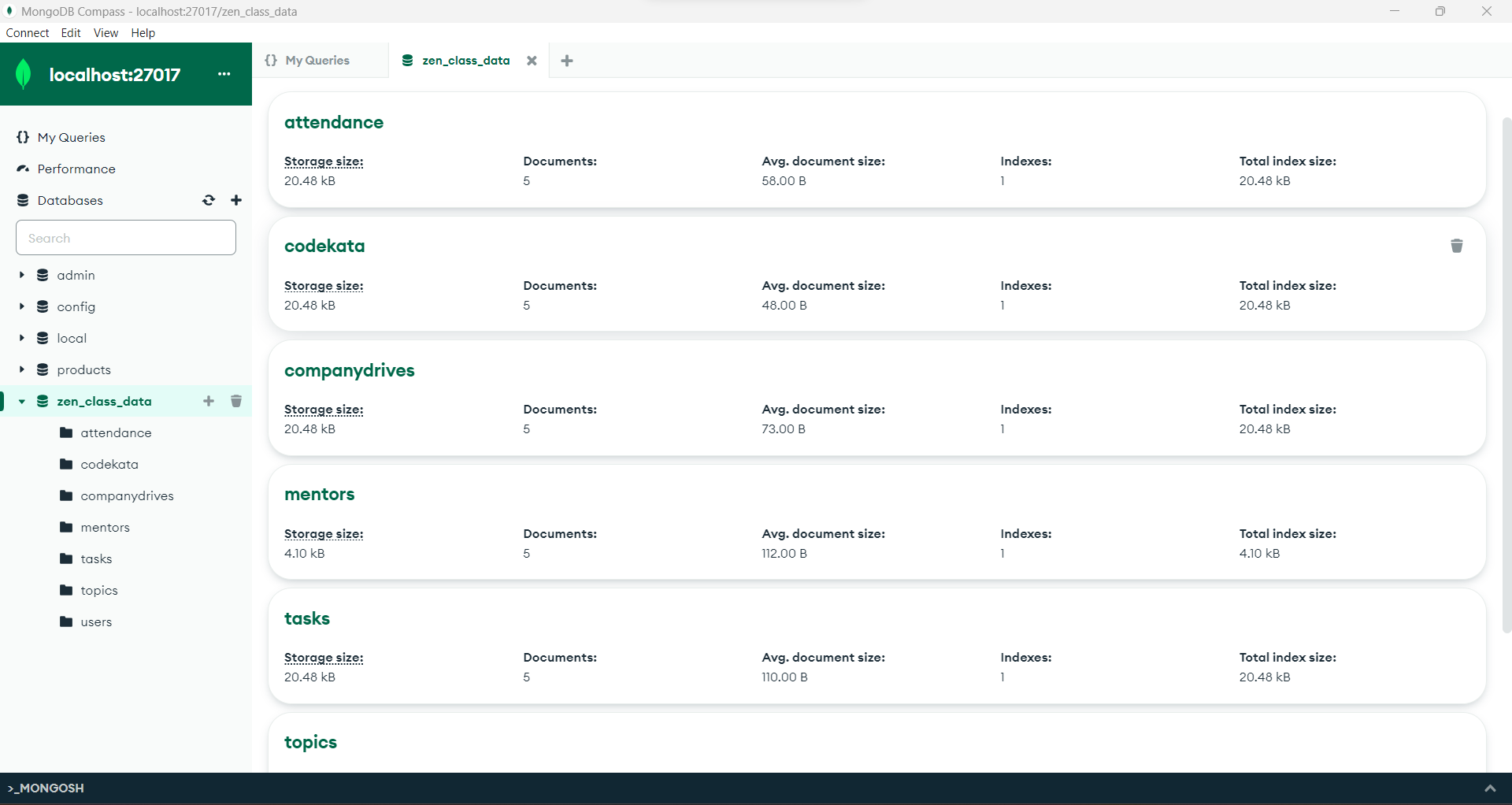
**MongoDB Task**

**Design database for Zen class programme:**



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

Solution:

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"

}

}

])

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

**Solution :**

db.companydrives.find({

$or: [

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

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

]

})

1. 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"

}

}

])

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

Solution:

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

}

}

])

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

Solution:

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,

mentee\_count: 1

}

}

])

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

Solution:

db.attendance.aggregate([

{

$lookup: {

from: "topics",

localField: "topicid",

foreignField: "topicid",

as: "topics"

}

},

{

$lookup: {

from: "tasks",

localField: "topicid",

foreignField: "topicid",

as: "tasks"

}

},

{

$match: {

attended: false,

"tasks.submitted": false,

$and: [

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

{ "topics.topic\_date": { $lte: new Date("31-oct-2020") } },

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

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

]

}

},

{$count: "No\_of\_students\_absent"}

])