

## MongoDB Task - 2

### Zen Class Database Collection Task:

Create database:

User Zen Class:

The screenshot displays the MongoDB Compass interface and the MongoDB Shell. In the Compass window, the 'zen\_class' database is selected, and the 'users' collection is visible. The collection details show a storage size of 4.10 kB, 0 documents, an average document size of 0 B, 1 index, and a total index size of 4.10 kB. The MongoDB Shell window shows the following commands and output:

```
> MONGOOSH
> use zen_class
< switched to db zen_class
> db.createCollection("users")
< { ok: 1 }
> db.users.insertMany([
  { userid: 1, name: "Santhosh", email: "santhosh@gmail.com" },
  { userid: 2, name: "Surya", email: "surya@gmail.com" },
  { userid: 3, name: "Shiyam", email: "shiyam@gmail.com" },
  { userid: 4, name: "Muhib", email: "muhib@gmail.com" },
  { userid: 5, name: "Surya Kumar", email: "suryakumar@gmail.com" }
])
```

---

Create collection and insert data – “USERS” :

db.users.insertMany([

{ userid: 1, name: "Santhosh", email: "santhosh@gmail.com" },

{ userid: 2, name: "Surya", email: "surya@gmail.com" },

{ userid: 3, name: "Shiyam", email: "shiyam@gmail.com" },

{ userid: 4, name: "Muhib", email: "muhib@gmail.com" },

{ userid: 5, name: "Surya Kumar", email: "suryakumar@gmail.com" } }

])

```

>_MONGOOSH
0 v
> use zen_class
< switched to db zen_class
> db.createCollection("users")
< { ok: 1 }
> db.users.insertMany([
  { userid: 1, name: "Santhosh", email: "sathosh@gmail.com" },
  { userid: 2, name: "Surya", email: "surya@gmail.com" },
  { userid: 3, name: "Shiyam", email: "shiyam@gmail.com" },
  { userid: 4, name: "Muhib", email: "muhib@gmail.com" },
  { userid: 5, name: "Surya Kumar", email: "suryakumar@gmail.com" }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65a9feaf4594f97ba3bd6a16'),
    '1': ObjectId('65a9feaf4594f97ba3bd6a17'),
    '2': ObjectId('65a9feaf4594f97ba3bd6a18'),
    '3': ObjectId('65a9feaf4594f97ba3bd6a19'),
    '4': ObjectId('65a9feaf4594f97ba3bd6a1a')
  }
}

```

## Create collection and insert data – “CODEKATA” :

```
db.createCollection("codekata");
```

```

db.codekata.insertMany([
  { userid: 1, problems: 50 },
  { userid: 2, problems: 60 },
  { userid: 3, problems: 90 },
  { userid: 4, problems: 51 },
  { userid: 5, problems: 61 }
])

```

```

> db.createCollection("codekata");
db.codekata.insertMany([
  { userid: 1, problems: 50 },
  { userid: 2, problems: 60 },
  { userid: 3, problems: 90 },
  { userid: 4, problems: 51 },
  { userid: 5, problems: 61 }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65aa00714594f97ba3bd6a1b'),
    '1': ObjectId('65aa00714594f97ba3bd6a1c'),
    '2': ObjectId('65aa00714594f97ba3bd6a1d'),
    '3': ObjectId('65aa00714594f97ba3bd6a1e'),
    '4': ObjectId('65aa00714594f97ba3bd6a1f')
  }
}

```

### Create collection and insert data – “ATTENDANCE” :

```
db.createCollection("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 }  
])
```



```
>_MONGOSH  
> db.createCollection("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 }  
)  
< {  
  acknowledged: true,  
  insertedIds: {  
    '0': ObjectId('65aa04184594f97ba3bd6a20'),  
    '1': ObjectId('65aa04184594f97ba3bd6a21'),  
    '2': ObjectId('65aa04184594f97ba3bd6a22'),  
    '3': ObjectId('65aa04184594f97ba3bd6a23'),  
    '4': ObjectId('65aa04184594f97ba3bd6a24')  
  }  
}  
zen_class>
```

### Create collection and insert data – “TOPICS” :

```
db.createCollection("topics");  
  
db.topics.insertMany([  
    { topicid: 1, topic: "HTML", topic_date: new Date("18-Oct-2023") },  
    { topicid: 2, topic: "CSS", topic_date: new Date("28-Oct-2023") },  
    { topicid: 3, topic: "JavaScript", topic_date: new Date("05-Nov-2023") },  
    { topicid: 4, topic: "ReactJS", topic_date: new Date("15-Nov-2023") },  
    { topicid: 5, topic: "NodeJS", topic_date: new Date("25-Nov-2023") }  
])
```

```
> MONGOSH
> db.createCollection("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("15-Nov-2020") },
  { topicid: 5, topic: "NodeJS", topic_date: new Date("25-Nov-2020") }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65aa0c38a0512bdd547212fd'),
    '1': ObjectId('65aa0c38a0512bdd547212fe'),
    '2': ObjectId('65aa0c38a0512bdd547212ff'),
    '3': ObjectId('65aa0c38a0512bdd54721300'),
    '4': ObjectId('65aa0c38a0512bdd54721301')
  }
}
zen_class>
```

---

## Create collection and insert data – “TASKS” :

```
db.createCollection("tasks");
```

```
db.tasks.insertMany([
```

```
  { taskid: 1, topicid: 1, userid: 1, task: "HTML Task", due_date: new
Date("18-Oct-2023"), submitted: true },
```

```
  { taskid: 2, topicid: 2, userid: 2, task: "CSS Task", due_date: new Date("28-
Oct-2023"), submitted: false },
```

```
  { taskid: 3, topicid: 3, userid: 3, task: "Javascript Task", due_date: new
Date("05-Nov-2023"), submitted: true },
```

```
  { taskid: 4, topicid: 4, userid: 4, task: "React Task", due_date: new Date("15-
Nov-2023"), submitted: true },
```

```
  { taskid: 5, topicid: 5, userid: 5, task: "NodeJS Task", due_date: new
Date("25-Nov-2023"), submitted: false }
```

```
])
```

```

>_MONGOSH
> db.createCollection("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 Task", due_date: new Date("05-Nov-2020"), submitted: true },
  { taskid: 4, topicid: 4, userid: 4, task: "React Task", due_date: new Date("15-Nov-2020"), submitted: true },
  { taskid: 5, topicid: 5, userid: 5, task: "NodeJS Task", due_date: new Date("25-Nov-2020"), submitted: false }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65aa0cffa0512bdd54721302'),
    '1': ObjectId('65aa0cffa0512bdd54721303'),
    '2': ObjectId('65aa0cffa0512bdd54721304'),
    '3': ObjectId('65aa0cffa0512bdd54721305'),
    '4': ObjectId('65aa0cffa0512bdd54721306')
  }
}
zen_class>

```

## Create collection and insert data – “COMPANY DRIVES” :

```
db.createCollection("companydrives");
```

```

db.companydrives.insertMany([
  { userid: 1, drive_date: new Date("20-Oct-2020"), company: "Apple" },
  { userid: 1, drive_date: new Date("22-Oct-2020"), company: "Amazon" },
  { userid: 2, drive_date: new Date("25-Oct-2020"), company: "TCS" },
  { userid: 3, drive_date: new Date("30-Oct-2020"), company: "Flipkart" },
  { userid: 4, drive_date: new Date("05-Nov-2020"), company: "Zomato" }
])

```

```

>_MONGOSH
> db.createCollection("companydrives");
db.companydrives.insertMany([
  { userid: 1, drive_date: new Date("20-Oct-2020"), company: "Apple" },
  { userid: 1, drive_date: new Date("22-Oct-2020"), company: "Amazon" },
  { userid: 2, drive_date: new Date("25-Oct-2020"), company: "TCS" },
  { userid: 3, drive_date: new Date("30-Oct-2020"), company: "Flipkart" },
  { userid: 4, drive_date: new Date("05-Nov-2020"), company: "Zomato" }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65aa0db7a0512bdd54721307'),
    '1': ObjectId('65aa0db7a0512bdd54721308'),
    '2': ObjectId('65aa0db7a0512bdd54721309'),
    '3': ObjectId('65aa0db7a0512bdd5472130a'),
    '4': ObjectId('65aa0db7a0512bdd5472130b')
  }
}
zen_class>

```

## Create collection and insert data – “MENTORS” :

```
db.createCollection("mentors");
```

```
db.mentors.insertMany([
```

```
    { mentorid: 1, mentorname: "Rupan", mentor_email: "rupan@gmail.com",  
      mentee_count: 20 },
```

```
    { mentorid: 2, mentorname: "Nagaraj", mentor_email: "nagaraj@gmail.com",  
      mentee_count: 18 },
```

```
    { mentorid: 3, mentorname: "Krishna", mentor_email: "krishna@gmail.com",  
      mentee_count: 30 },
```

```
    { mentorid: 4, mentorname: "Sabhari", mentor_email: "sabhari@gmail.com",  
      mentee_count: 15 },
```

```
    { mentorid: 5, mentorname: "Manoj", mentor_email: "manoj@gmail.com",  
      mentee_count: 20 }]
```

```
)
```

```
>_MONGOSH
> db.createCollection("mentors");
db.mentors.insertMany([
  { mentorid: 1, mentorname: "Rupan", mentor_email: "rupan@gmail.com", mentee_count: 20 },
  { mentorid: 2, mentorname: "Nagaraj", mentor_email: "nagaraj@gmail.com", mentee_count: 18 },
  { mentorid: 3, mentorname: "Krishna", mentor_email: "krishna@gmail.com", mentee_count: 30 },
  { mentorid: 4, mentorname: "Sabhari", mentor_email: "sabhari@gmail.com", mentee_count: 15 },
  { mentorid: 5, mentorname: "Manoj", mentor_email: "manoj@gmail.com", mentee_count: 20 }
])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65aa0e3aa0512bbd5472130c'),
    '1': ObjectId('65aa0e3aa0512bbd5472130d'),
    '2': ObjectId('65aa0e3aa0512bbd5472130e'),
    '3': ObjectId('65aa0e3aa0512bbd5472130f'),
    '4': ObjectId('65aa0e3aa0512bbd54721310')
  }
}
zen_class>
```

# CREATED DATABASE

The screenshot shows the MongoDB Compass interface for a database named 'zen\_class' on 'localhost:27017'. The left sidebar lists the collections: attendance, codekata, companydrives, mentors, tasks, and users. The main panel displays details for five collections:

Collection Name	Storage size	Documents	Avg. document size	Indexes	Total index size
attendance	20.48 kB	5	58.00 B	1	20.48 kB
codekata	20.48 kB	5	48.00 B	1	20.48 kB
companydrives	20.48 kB	5	73.00 B	1	20.48 kB
mentors	20.48 kB	5	112.00 B	1	20.48 kB
tasks	20.48 kB	5	110.00 B	1	20.48 kB

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"
    }
  }
])

```



```
>_MONGOSH
< {
  topicid: 1,
  topic: 'HTML',
  topic_date: 2020-10-17T18:30:00.000Z,
  tasks: [
    'HTML Task'
  ],
  due_dates: [
    2020-10-17T18:30:00.000Z
  ]
}
{
  topicid: 2,
  topic: 'CSS',
  topic_date: 2020-10-27T18:30:00.000Z,
  tasks: [
    'CSS Task'
  ],
  due_dates: [
    2020-10-27T18:30:00.000Z
  ]
}
```

---

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

**Solution :**

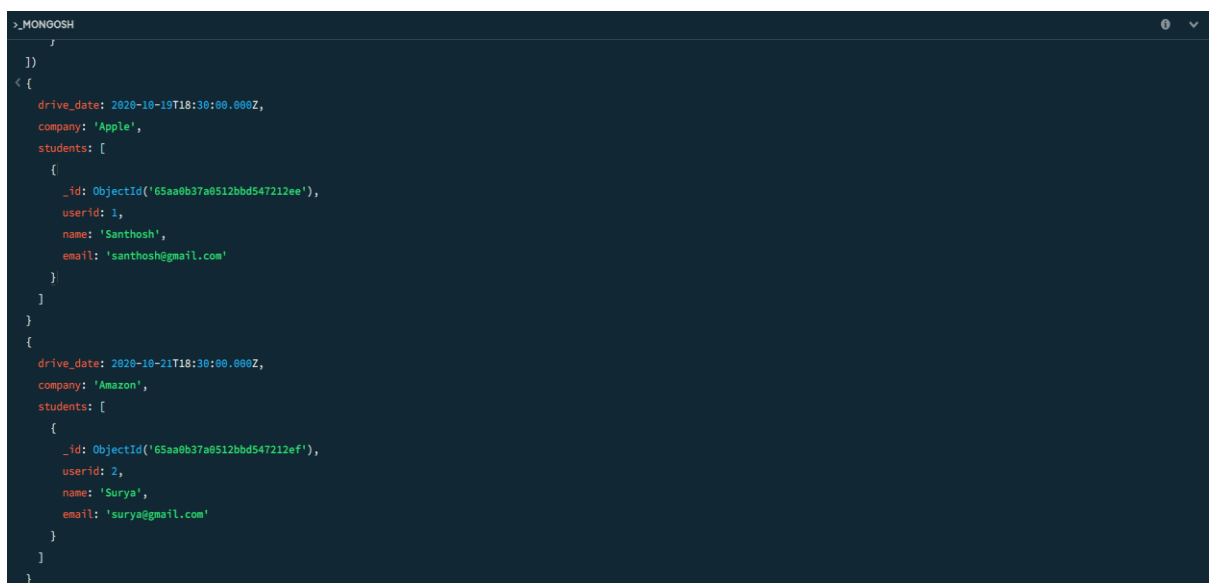
```
db.companydrives.find({
  $or: [
    { drive_date: { $gte: new Date("15-oct-2020") } },
    { drive_date: { $lte: new Date("31-oct-2020") } }
  ]
})
```

```
>_MONGOSH
> db.companydrives.find({
  $or: [
    { drive_date: { $gte: new Date("15-oct-2020") } },
    { drive_date: { $lte: new Date("31-oct-2020") } }
  ]
})
< {
  _id: ObjectId('65aa0db7a0512bbd54721307'),
  userid: 1,
  drive_date: 2020-10-19T18:30:00.000Z,
  company: 'Apple'
}
{
  _id: ObjectId('65aa0db7a0512bbd54721308'),
  userid: 2,
  drive_date: 2020-10-21T18:30:00.000Z,
  company: 'Amazon'
}
{
  _id: ObjectId('65aa0db7a0512bbd54721309'),
  userid: 5,
```

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

#### Solution :

```
db.companydrives.aggregate([
  {
    $lookup: {
      from: "users",
      localField: "userid",
      foreignField: "userid",
      as: "userinfo"
    }
  },
  {
    $project: {
      _id: 0,
      company: 1,
      drive_date: 1,
      students: "$userinfo"
    }
  }
])
```



```
>_MONGOSH
/
})
< {
  drive_date: 2020-10-19T18:30:00.000Z,
  company: 'Apple',
  students: [
    {
      _id: ObjectId('65aa0b37a0512bhd547212ee'),
      userid: 1,
      name: 'Santhosh',
      email: 'santhosh@gmail.com'
    }
  ]
}
{
  drive_date: 2020-10-21T18:30:00.000Z,
  company: 'Amazon',
  students: [
    {
      _id: ObjectId('65aa0b37a0512bhd547212ef'),
      userid: 2,
      name: 'Surya',
      email: 'surya@gmail.com'
    }
  ]
}
```

#### 4. 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
    }
}
])

```

```

> MONGOSH
total_problems_solved: 51,
userid: 4,
username: [
  'Muhib'
]
}
{
  total_problems_solved: 61,
  userid: 5,
  username: [
    'Surya Kumar'
  ]
}
{
  total_problems_solved: 50,
  userid: 1,
  username: [
    'Santhosh'
  ]
}
{
  total_problems_solved: 60,
  userid: 2,
  username: [
    'Surya'
  ]
}

```

**5. 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
    }
}
}
D)

```

```

>_MONGOSH
> 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
    }
  }
])

```

**6. 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"}}))

```

```

>_MONGOOSH
zen_class> 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") } }
      ]
    }
  }
])

```