

# Design Database for Zen Class Programme

## # Create database

```
use zen_class
```

---

## # 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" }  
])
```

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



```
>_MONGOSH  
  
> 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" }  
])  
< {  
  acknowledged: true,  
  insertedIds: {
```

```
> _MONGOSH
> db.users.find()
< {
  _id: ObjectId('662a622e93389aeb141c3220'),
  userid: 1,
  name: 'Santhosh',
  email: 'santhosh@gmail.com'
}
{
  _id: ObjectId('662a622e93389aeb141c3221'),
  userid: 2,
  name: 'Surya',
  email: 'surya@gmail.com'
}
{
  _id: ObjectId('662a622e93389aeb141c3222'),
  userid: 3,
  name: 'Shiyam',
  email: 'shiyam@gmail.com'
}
```

---

## # 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 }
])
```

```

>_MONGOSH
> db.codekata.find()
< {
  _id: ObjectId('662a634693389aeb141c3225'),
  userid: 1,
  problems: 50
}
{
  _id: ObjectId('662a634693389aeb141c3226'),
  userid: 2,
  problems: 60
}
{
  _id: ObjectId('662a634693389aeb141c3227'),
  userid: 3,
  problems: 90
}
{
  _id: ObjectId('662a634693389aeb141c3228'),
  userid: 4,
  problems: 51
}

```

---

## # 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.attendance.find()
< {
  _id: ObjectId('662a643293389aeb141c322a'),
  userid: 1,
  topicid: 2,
  attended: true
}
{
  _id: ObjectId('662a643293389aeb141c322b'),
  userid: 2,
  topicid: 1,
  attended: true
}
{
  _id: ObjectId('662a643293389aeb141c322c'),
  userid: 3,
  topicid: 5,
  attended: true
}
```

---

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

```
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-Oct-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") }
])
```

```

>_MONGOSH
> db.topics.find()
< {
  _id: ObjectId('662a649c93389aeb141c322f'),
  topicid: 1,
  topic: 'HTML',
  topic_date: 2020-10-17T18:30:00.000Z
}
{
  _id: ObjectId('662a649c93389aeb141c3230'),
  topicid: 2,
  topic: 'CSS',
  topic_date: 2020-10-27T18:30:00.000Z
}
{
  _id: ObjectId('662a649c93389aeb141c3231'),
  topicid: 3,
  topic: 'JavaScript',
  topic_date: 2020-11-04T18:30:00.000Z
}

```

## # 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-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 }

```

```
])
```

```
>_MONGOSH
< {
  _id: ObjectId('662a652d93389aeb141c3234'),
  taskid: 1,
  topicid: 1,
  userid: 1,
  task: 'HTML Task',
  due_date: 2020-10-17T18:30:00.000Z,
  submitted: true
}
{
  _id: ObjectId('662a652d93389aeb141c3235'),
  taskid: 2,
  topicid: 2,
  userid: 2,
  task: 'CSS Task',
  due_date: 2020-10-27T18:30:00.000Z,
  submitted: false
}
```

---

**# 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.companydrives.find()
< {
  _id: ObjectId('662a65af93389aeb141c3239'),
  userid: 1,
  drive_date: 2020-10-19T18:30:00.000Z,
  company: 'Apple'
}
{
  _id: ObjectId('662a65af93389aeb141c323a'),
  userid: 2,
  drive_date: 2020-10-21T18:30:00.000Z,
  company: 'Amazon'
}
{
  _id: ObjectId('662a65af93389aeb141c323b'),
  userid: 3,
  drive_date: 2020-10-24T18:30:00.000Z,
  company: 'TCS'
}
{

```

## # 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.mentors.find()
< {
  _id: ObjectId('662a666b93389aeb141c323e'),
  mentorid: 1,
  mentorname: 'Rupan',
  mentor_email: 'rupan@gmail.com',
  mentee_count: 20
}
{
  _id: ObjectId('662a666b93389aeb141c323f'),
  mentorid: 2,
  mentorname: 'Nagaraj',
  mentor_email: 'nagaraj@gmail.com',
  mentee_count: 18
}
{
  _id: ObjectId('662a666b93389aeb141c3240'),
  mentorid: 3,
  mentorname: 'Krishna',
  mentor_email: 'krishna@gmail.com',
```

---

## CREATED DATABASE

The screenshot displays the MongoDB Atlas web interface. On the left, a sidebar shows the 'cluster0.cpqs8q...' and a list of databases. The 'zen\_class' database is selected, showing a list of collections: attendance, codekata, companydrives, mentors, tasks, topics, and users. The main panel shows details for the 'attendance' collection, including storage size (20.48 kB), documents (5), avg. document size (58.00 B), indexes (1), and total index size (20.48 kB). Below this, a table lists the other collections with their respective statistics.

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	83.00 B	1	36.86 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"
    }
}
])

```

```

{
  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") } }
  ]
})

```

```
< {
  _id: ObjectId('662a65af93389aeb141c3239'),
  userid: 1,
  drive_date: 2020-10-19T18:30:00.000Z,
  company: 'Apple'
}
{
  _id: ObjectId('662a65af93389aeb141c323a'),
  userid: 1,
  drive_date: 2020-10-21T18:30:00.000Z,
  company: 'Amazon'
}
{
  _id: ObjectId('662a65af93389aeb141c323b'),
  userid: 2,
  drive_date: 2020-10-24T18:30:00.000Z,
  company: 'TCS'
}
{
```

### 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"
    }
}])

```

```

    drive_date: 2020-10-19T18:30:00.000Z,
    company: 'Apple',
    students: [
      {
        _id: ObjectId('662a622e93389aeb141c3220'),
        userid: 1,
        name: 'Santhosh',
        email: 'santhosh@gmail.com'
      }
    ]
  }
}
{
  drive_date: 2020-10-21T18:30:00.000Z,
  company: 'Amazon',
  students: [
    {
      _id: ObjectId('662a622e93389aeb141c3221'),
      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
    }
  }
])

```

```

{
  total_problems_solved: 50,
  userid: 1,
  username: [
    'Santhosh'
  ]
}
{
  total_problems_solved: 60,
  userid: 2,
  username: [
    'Surya'
  ]
}
{
  total_problems_solved: 51,
  userid: 4,
  username: [
    'Muhib'
  ]
}

```

---

**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
    }
  }
])
```

```

    },
    > 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"
  })
```



```
<
> 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") } }
      ]
    }
  }
])
```