MongoDB Task

Design database for Zen class programme

Inserting Data into the collections:

Users:

```
db.users.insertMany([
  {
    "user_id": 1,
    "name": "Alice",
    "email": "alice@example.com",
    "attendance": [
      { "date": ISODate("2020-10-16T00:00:00Z"), "status": "absent" },
      { "date": ISODate("2020-10-17T00:00:00Z"), "status": "present" }
    ],
    "tasks_submitted": [1, 2],
    "drive_id": 1,
    "codekata_problems": [101, 102, 103]
  },
  {
    "user_id": 2,
    "name": "Bob",
    "email": "bob@example.com",
    "attendance": [
      { "date": ISODate("2020-10-16T00:00:00Z"), "status": "present" },
      { "date": ISODate("2020-10-18T00:00:00Z"), "status": "absent" }
    ],
    "tasks_submitted": [2, 3],
    "drive_id": 2,
    "codekata_problems": [101, 104]
  },
  {
    "user_id": 3,
    "name": "Charlie",
    "email": "charlie@example.com",
    "attendance": [
      { "date": ISODate("2020-10-17T00:00:00Z"), "status": "absent" },
      { "date": ISODate("2020-10-19T00:00:00Z"), "status": "absent" }
    ],
    "tasks_submitted": [],
    "drive_id": 3,
```

```
    "codekata_problems": []
  },
  {
    "user_id": 4,
    "name": "David",
    "email": "david@example.com",
    "attendance": [
      { "date": ISODate("2020-10-16T00:00:00Z"), "status": "present" },
      { "date": ISODate("2020-10-17T00:00:00Z"), "status": "present" }
    ],
    "tasks_submitted": [1, 4],
    "drive_id": 1,
    "codekata_problems": [105, 106, 107, 108]
  },
  {
    "user_id": 5,
    "name": "Eve",
    "email": "eve@example.com",
    "attendance": [
      { "date": ISODate("2020-10-18T00:00:00Z"), "status": "absent" },
      { "date": ISODate("2020-10-19T00:00:00Z"), "status": "present" }
    ],
    "tasks_submitted": [5],
    "drive_id": 4,
    "codekata_problems": [101, 102, 103, 104, 105]
  }
]);
```

Codekata:

```
db.codekata.insertMany([
  {
    "codekata_id": 1,
    "user_id": 1,
    "problem_id": 101
  },
  {
    "codekata_id": 2,
    "user_id": 1,
    "problem_id": 102
  },
  {
    "codekata_id": 3,
    "user_id": 2,
    "problem_id": 101
```

```
  },
  {
    "codekata_id": 4,
    "user_id": 2,
    "problem_id": 102
  },
  {
    "codekata_id": 5,
    "user_id": 3,
    "problem_id": 103
  }])
```

Attendance:

```
db.attendance.insertMany([
  {
    "attendance_id": 1,
    "user_id": 1,
    "date": ISODate("2020-10-15T00:00:00Z"),
    "status": "absent"
  },
  {
    "attendance_id": 2,
    "user_id": 1,
    "date": ISODate("2020-10-16T00:00:00Z"),
    "status": "present"
  },
  {
    "attendance_id": 3,
    "user_id": 2,
    "date": ISODate("2020-10-15T00:00:00Z"),
    "status": "present"
  },
  {
    "attendance_id": 4,
    "user_id": 2,
    "date": ISODate("2020-10-16T00:00:00Z"),
    "status": "absent"
  },
  {
    "attendance_id": 5,
    "user_id": 3,
    "date": ISODate("2020-10-15T00:00:00Z"),
    "status": "absent"
  }])
```

Topics:

```
db.topics.insertMany([
  {
    "topic_id": 1,
    "name": "MongoDB Basics",
    "date_taught": ISODate("2020-10-05T00:00:00Z")
  },
  {
    "topic_id": 2,
    "name": "Indexing in MongoDB",
    "date_taught": ISODate("2020-10-10T00:00:00Z")
  },
  {
    "topic_id": 3,
    "name": "Aggregation in MongoDB",
    "date_taught": ISODate("2020-10-15T00:00:00Z")
  },
  {
    "topic_id": 4,
    "name": "Replication in MongoDB",
    "date_taught": ISODate("2020-10-20T00:00:00Z")
  },
  {
    "topic_id": 5,
    "name": "Sharding in MongoDB",
    "date_taught": ISODate("2020-10-25T00:00:00Z")
  }
]);
```

Tasks:

```
db.tasks.insertMany([
  {
    "task_id": 1,
    "name": "Task 1",
    "due_date": ISODate("2020-10-15T00:00:00Z")
  },
  {
    "task_id": 2,
    "name": "Task 2",
    "due_date": ISODate("2020-10-20T00:00:00Z")
```

```
  },
  {
    "task_id": 3,
    "name": "Task 3",
    "due_date": ISODate("2020-10-25T00:00:00Z")
  },
  {
    "task_id": 4,
    "name": "Task 4",
    "due_date": ISODate("2020-10-30T00:00:00Z")
  },
  {
    "task_id": 5,
    "name": "Task 5",
    "due_date": ISODate("2020-10-31T00:00:00Z")
  }
]);
```

company_drives:

```
db.company_drives.insertMany([
  {
    "drive_id": 1,
    "company_name": "Company A",
    "date": ISODate("2020-10-16T00:00:00Z")
  },
  {
    "drive_id": 2,
    "company_name": "Company B",
    "date": ISODate("2020-10-18T00:00:00Z")
  },
  {
    "drive_id": 3,
    "company_name": "Company C",
    "date": ISODate("2020-10-20T00:00:00Z")
  },
  {
    "drive_id": 4,
    "company_name": "Company D",
    "date": ISODate("2020-10-25T00:00:00Z")
  }
]);
```

Mentors:

```
db.mentors.insertMany([
  {
    "mentor_id": 1,
    "name": "Mentor A",
    "mentees": ["Alice", "Bob", "Charlie", "David", "Eve", "Frank", "Grace", "Hank", "Ivy", "Jack",
"Karen", "Leo", "Mona", "Nick", "Oscar", "Pam"]
  },
  {
    "mentor_id": 2,
    "name": "Mentor B",
    "mentees": ["Quinn", "Ray", "Sara", "Tom"]
  },
  {
    "mentor_id": 3,
    "name": "Mentor C",
    "mentees": ["Uma", "Victor", "Wendy", "Xander", "Yara", "Zane", "Ava", "Ben", "Cody", "Diana",
"Eli", "Fay", "Gina", "Hugo", "Iris", "Jack"]
  },
  {
    "mentor_id": 4,
    "name": "Mentor D",
    "mentees": ["Kyle", "Liam", "Mia", "Nina", "Owen", "Paul", "Quincy", "Rita", "Sam", "Tina", "Ursula",
"Vince", "Wade", "Xena"]
  }
]);
```

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

Query:

```
db.topics.aggregate([
  {
    $match: {
      "date_taught": {
        $gte: ISODate("2020-10-01T00:00:00Z"),
        $lte: ISODate("2020-10-31T23:59:59Z")
      }
    }
  },
  {
    $project: {
      _id: 0,
      type: { $literal: "topic" },
      date: "$date_taught",
      name: 1
    }
```

```
  },
  {
    $unionWith: {
      coll: "tasks",
      pipeline: [
        {
          $match: {
            "due_date": {
              $gte: ISODate("2020-10-01T00:00:00Z"),
              $lte: ISODate("2020-10-31T23:59:59Z")
            }
          }
        },
        {
          $project: {
            _id: 0,
            type: { $literal: "task" },
            date: "$due_date",
            name: 1
          }
        }
      ]
    }
  }
]).pretty()
```

Result:

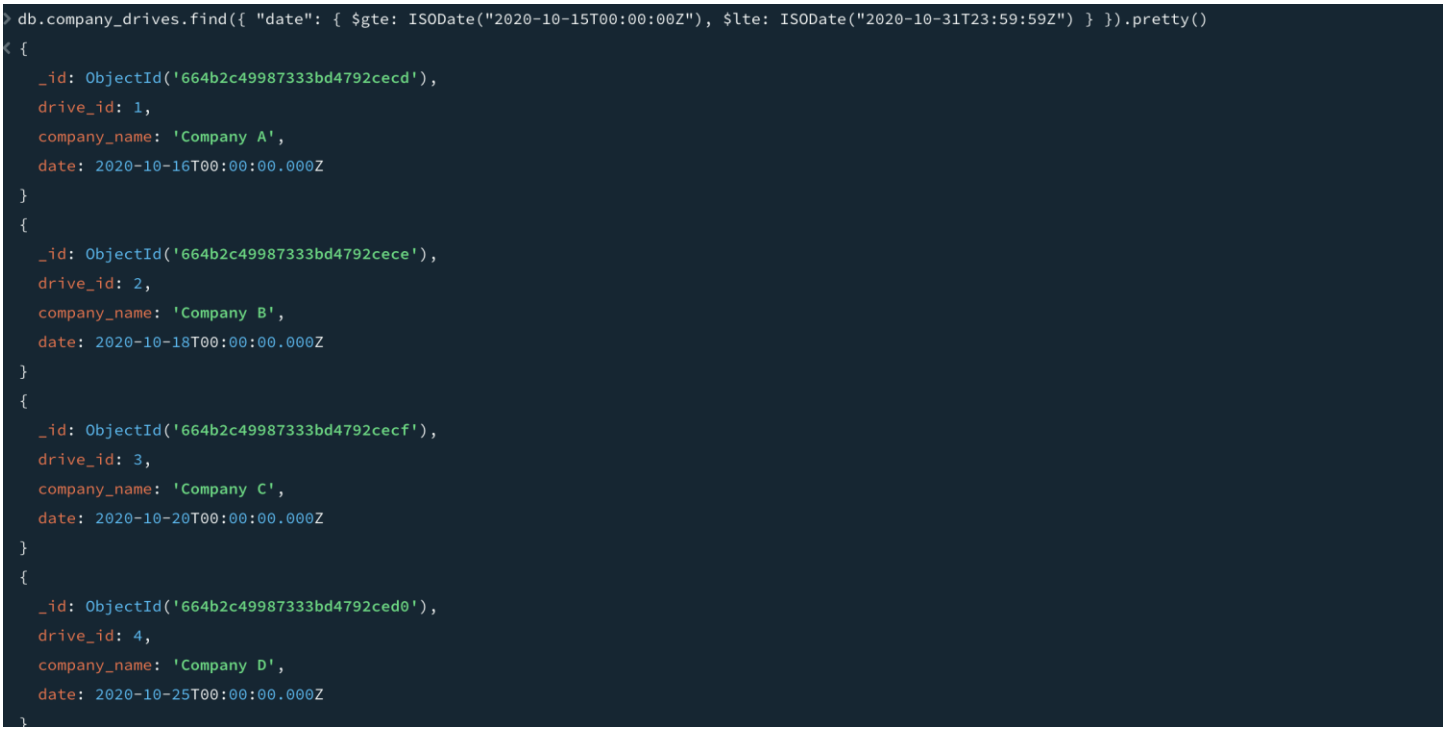
```
> db.topics.aggregate([{ $match: { "date_taught": { $gte: ISODate("2020-10-01T00:00:00Z"), $lte: ISODate("2020-10-31T23:59:59Z") } } }, { $project: { _id: 0, type: "topic", date: "$date_taugh
< {
    name: 'MongoDB Basics',
    type: 'topic',
    date: 2020-10-05T00:00:00.000Z
  }
  {
    name: 'Indexing in MongoDB',
    type: 'topic',
    date: 2020-10-10T00:00:00.000Z
  }
  {
    name: 'Aggregation in MongoDB',
    type: 'topic',
    date: 2020-10-15T00:00:00.000Z
  }
  {
    name: 'Replication in MongoDB',
    type: 'topic',
    date: 2020-10-20T00:00:00.000Z
  }
  {
    name: 'Sharding in MongoDB',
    type: 'topic',
    date: 2020-10-25T00:00:00.000Z
  }
  {
    name: 'Task 1',
    type: 'task',
    date: 2020-10-15T00:00:00.000Z
  }
  {
```

```
  }
  {
    name: 'Task 2',
    type: 'task',
    date: 2020-10-20T00:00:00.000Z
  }
  {
    name: 'Task 3',
    type: 'task',
    date: 2020-10-25T00:00:00.000Z
  }
  {
    name: 'Task 4',
    type: 'task',
    date: 2020-10-30T00:00:00.000Z
  }
  {
    name: 'Task 5',
    type: 'task',
    date: 2020-10-31T00:00:00.000Z
```

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

Query: db.company_drives.find({ "date": { $gte: ISODate("2020-10-15T00:00:00Z"), $lte: ISODate("2020-10-31T23:59:59Z") } }).pretty()

Result:



```
> db.company_drives.find({ "date": { $gte: ISODate("2020-10-15T00:00:00Z"), $lte: ISODate("2020-10-31T23:59:59Z") } }).pretty()
< {
    _id: ObjectId('664b2c49987333bd4792cecd'),
    drive_id: 1,
    company_name: 'Company A',
    date: 2020-10-16T00:00:00.000Z
  }
  {
    _id: ObjectId('664b2c49987333bd4792cece'),
    drive_id: 2,
    company_name: 'Company B',
    date: 2020-10-18T00:00:00.000Z
  }
  {
    _id: ObjectId('664b2c49987333bd4792cecf'),
    drive_id: 3,
    company_name: 'Company C',
    date: 2020-10-20T00:00:00.000Z
  }
  {
    _id: ObjectId('664b2c49987333bd4792ced0'),
    drive_id: 4,
    company_name: 'Company D',
    date: 2020-10-25T00:00:00.000Z
  }
```

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

Query: db.company_drives.aggregate([{ $lookup: { from: "users", localField: "drive_id", foreignField: "drive_id", as: "students" } }]).pretty()

Result:

```
> db.company_drives.aggregate([
    {
      $lookup: {
        from: "users",
        localField: "drive_id",
        foreignField: "drive_id",
        as: "students"
      }
    }
  ]).pretty()
< {
    _id: ObjectId('664b2b705e53b3cb48769a79'),
    drive_id: 1,
    company: 'Company A',
    date: '2020-10-16',
    students_appeared: [
      1,
      2
    ],
    students: [
      {
        _id: ObjectId('664b2d30987333bd4792ced2'),
        user_id: 1,
        name: 'Alice',
        email: 'alice@example.com',
        attendance: [
          {
            date: 2020-10-16T00:00:00.000Z,
            status: 'absent'
          },
          {
            date: 2020-10-17T00:00:00.000Z,
            status: 'present'
```

```
        }
      ],
      tasks_submitted: [
        1,
        2
      ],
      drive_id: 1
    },
    {
      _id: ObjectId('664b2d30987333bd4792ced5'),
      user_id: 4,
      name: 'David',
      email: 'david@example.com',
      attendance: [
        {
          date: 2020-10-16T00:00:00.000Z,
          status: 'present'
        },
        {
          date: 2020-10-17T00:00:00.000Z,
          status: 'present'
        }
      ],
      tasks_submitted: [
        1,
        4
      ],
      drive_id: 1
    }
  ]
}
{
  _id: ObjectId('664b2b705e53b3cb48769a7a'),
```

```
  _id: ObjectId('664b2b705e53b3cb48769a7a'),
  drive_id: 2,
  company: 'Company B',
  date: '2020-10-18',
  students_appeared: [
    3,
    4
  ],
  students: [
    {
      _id: ObjectId('664b2d30987333bd4792ced3'),
      user_id: 2,
      name: 'Bob',
      email: 'bob@example.com',
      attendance: [
        {
          date: 2020-10-16T00:00:00.000Z,
          status: 'present'
        },
        {
          date: 2020-10-18T00:00:00.000Z,
          status: 'absent'
        }
      ],
      tasks_submitted: [
        2,
        3
      ],
      drive_id: 2
    }
  ]
}
```

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

Query: db.users.aggregate([{ $project: { _id: 0, name: 1, email: 1, problems_solved: { $size: { $ifNull: ["$codekata_problems", []] } } } }]).pretty()

Result:

```
> db.users.aggregate([{ $project: { _id: 0, name: 1, email: 1, problems_solved: { $size: { $ifNull: ["$codekata_problems", []] } } } }]).pretty()
< {
    name: 'Alice',
    email: 'alice@example.com',
    problems_solved: 0
  }
  {
    name: 'Bob',
    email: 'bob@example.com',
    problems_solved: 0
  }
  {
    name: 'Charlie',
    email: 'charlie@example.com',
    problems_solved: 0
  }
  {
    name: 'David',
    email: 'david@example.com',
    problems_solved: 0
  }
  {
    name: 'Eve',
    email: 'eve@example.com',
    problems_solved: 0
  }
  {
    name: 'Alice',
    email: 'alice@example.com',
    problems_solved: 3
  }
```

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

Query: db.mentors.aggregate([{ $project: { _id: 0, name: 1, mentees_count: { $size: "$mentees" } } }, { $match: { mentees_count: { $gt: 15 } } }]).pretty()

Result:

```
> db.mentors.aggregate([
    {
      $project: {
        _id: 0,
        name: 1,
        mentees_count: { $size: "$mentees" }
      }
    },
    {
      $match: {
        mentees_count: { $gt: 15 }
      }
    }
  ]).pretty()
< {
    name: 'Mentor A',
    mentees_count: 16
  }
  {
    name: 'Mentor C',
    mentees_count: 16
  }
```

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

Query: db.users.aggregate([{ $match: { "attendance": { $elemMatch: { "date": { $gte: ISODate("2020-10-15T00:00:00Z"), $lte: ISODate("2020-10-31T23:59:59Z") }, "status": "absent" } }, "tasks_submitted": { $size: 0 } } }, { $count: "total_users" }]).pretty()

Result:

```
> db.users.aggregate([
    {
      $match: {
        "attendance": { $elemMatch: { "date": { $gte: ISODate("2020-10-15T00:00:00Z"), $lte: ISODate("2020-10-31T23:59:59Z") }, "status": "absent" } },
        "tasks_submitted": { $size: 0 }
      }
    },
    {
      $count: "total_users"
    }
  ])
< {
    total_users: 1
  }
```