MongoDB Task

Design database for Zen class programme

users
codekata
attendance
topics
tasks
company_drives
mentors

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

query:

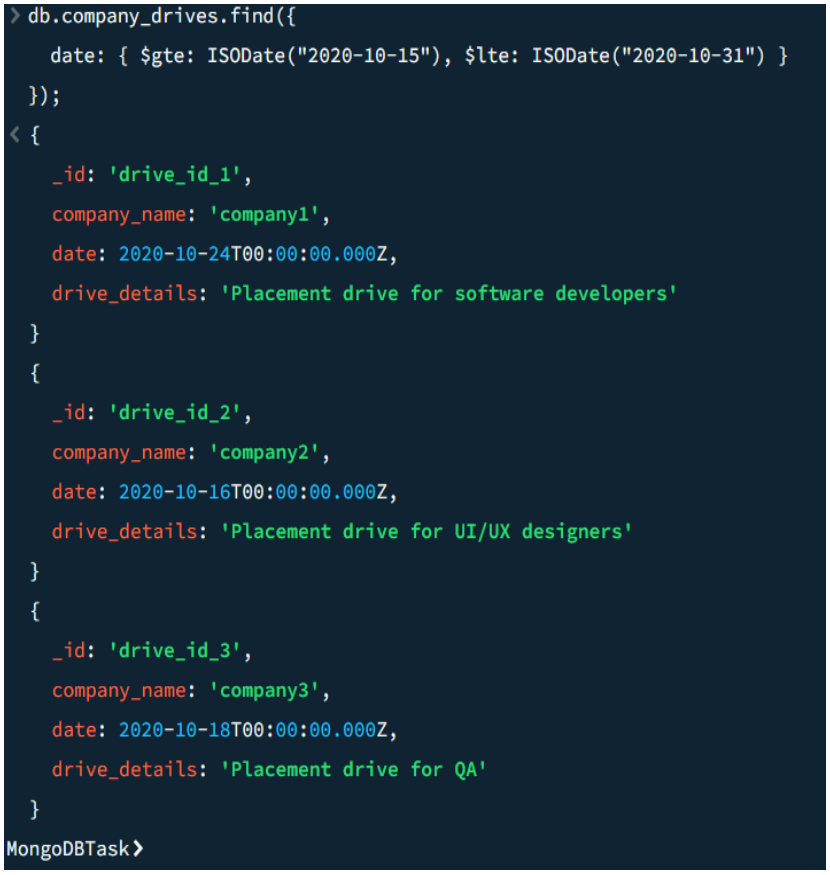
```
db.topics.aggregate([
  {
    $match: {
      "date": {
        $gte: ISODate("2020-10-01"),
        $lte: ISODate("2020-10-31")
      }
    }
  },
  {
    $lookup: {
      from: "tasks",
      localField: "task_id",
      foreignField: "_id",
      as: "task_details"
    }
  }
]);
```

The aggregation pipeline filters topics to select those with a date in October 2024. It then performs a $lookup to join the tasks collection based on the task_id field in topics. The resulting topics include an array of matched task details for each topic, providing a combined view of both topics and related tasks.

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

query:

db.company_drives.find({ date: { $gte: ISODate("2020-10-15"), $lte: ISODate("2020-10-31") } });



This query retrieves all company drives that occurred between October 15, 2020, and October 31, 2020. It filters documents in the company_drives collection where the date field falls within the specified range. The result provides details of the drives conducted during that period.

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

query:

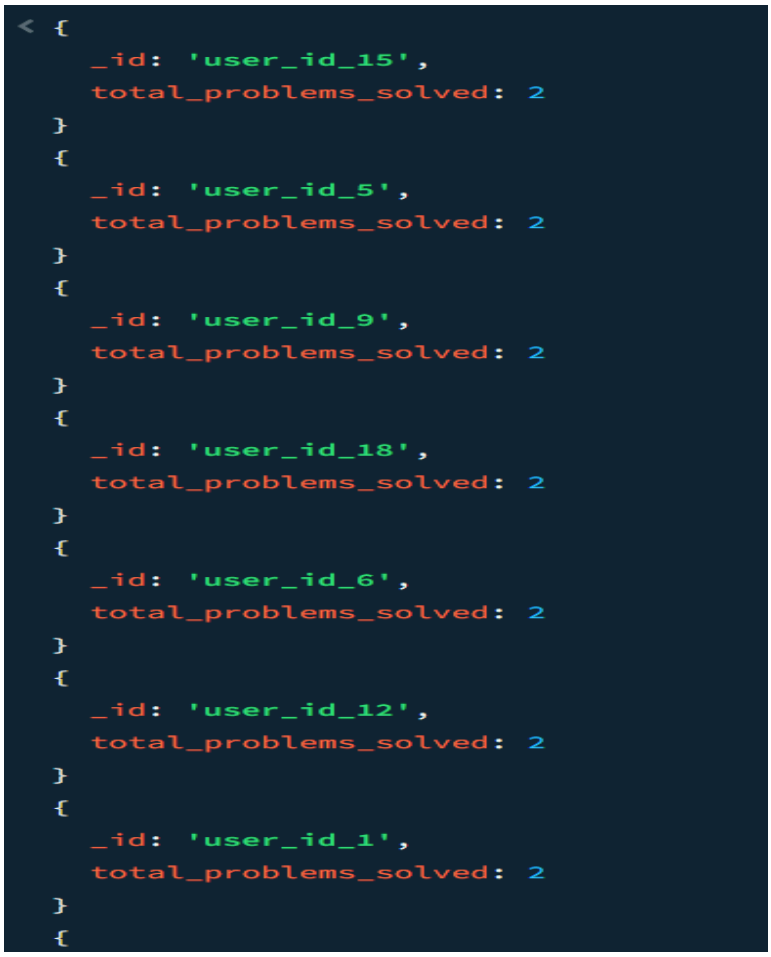
```
db.company_drives.aggregate([
  {
    $lookup: {
      from: "users",
      localField: "_id",
      foreignField: "appeared_drive_id",
      as: "students"
    }
  },
  {
    $project: {
      _id: 1,
      company_name: 1,
      date: 1,
      students: {
        _id: 1,
        name: 1,
        email: 1
      }
    }
  }
]);
```

This query retrieves all company drives along with the details of students who appeared for each drive. It uses $lookup to join the company_drives collection with the users collection based on the appeared_drive_id field. The $project stage limits the output to essential fields, showing drive details and an array of students (_id, name, email) for each drive.

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

query:

```
db.codekata.aggregate([
  {
    $unwind: "$user_id"   },
  {
    $group: {
      _id: "$user_id",
      total_problems_solved: { $sum: 1 }
    }
  },
  {
    $sort: { total_problems_solved: -1 }
  }
]);
```



This MongoDB query calculates the total number of problems solved by each user in the codekata collection. The $unwind stage expands the user_id array into individual entries, allowing each user to be counted separately. The $group stage aggregates the data by user_id and counts the number of challenges solved by each user, while the $sort stage (optional) organizes the results in descending order of problems solved.

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

```
db.users.aggregate([
 {
  $match: { role: "student" }  },
 {
  $group: {
    _id: "$mentor_id",
    mentee_count: { $sum: 1 }     }
 },
 {
  $match: {
    mentee_count: { $gte: 15 }
  }
 },
 {
  $lookup: {
    from: "mentors",
    localField: "_id",
    foreignField: "_id",
    as: "mentor_details"
  }
 },
 {
  $project: {
    _id: 1,
    mentee_count: 1,
    mentor_details: { name: 1, email: 1, expertise: 1 }
  }
 }
]);
```

This query filters users with the role "student" and groups them by their mentor ID, counting the number of mentees for each mentor. It then filters to find mentors with 15 or more mentees and joins the mentors' details from the "mentors" collection. Finally, it projects the mentor's details (name, email, expertise) along with the mentee count.

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: "2020-10-15",
            $lte: "2020-10-31"
          },
          "status": "absent"
        }
      },
      "role": "student"
    }
  },
  {
    $lookup: {
      from: "tasks",
      localField: "_id",
      foreignField: "submitted_users",
      as: "task_submissions"
    }
  },
  {
    $match: {
      "task_submissions": { $size: 0 }
    }
  },
  {
    $count: "absent_and_not_submitted_count"
  }
]);
```

```
      localField: "_id",
      foreignField: "submitted_users",
      as: "task_submissions"
    }
  },
  {
    $match: {
      "task_submissions": { $size: 0 } }
  },
  {
    $count: "absent_and_not_submitted_count"  }
]);
{
  absent_and_not_submitted_count: 3
}
```

This query first filters users who were absent between October 15, 2020, and October 31, 2020, and whose role is "student." It then performs a lookup with the tasks collection to find users who haven't submitted any tasks, based on the submitted_users array. Finally, it counts the number of users who were absent and have not submitted any tasks during that period.