ZEN CLASS MONGODB PROJECT (Command Line Version)

```
-----------------------------
1. CREATE DATABASE & COLLECTIONS
-----------------------------

> mongosh
> use zenClass

// USERS
db.users.insertMany([
  { _id: 1, name: "Alice", email: "alice@gmail.com", batch: "FSD52" },
  { _id: 2, name: "Bob", email: "bob@gmail.com", batch: "FSD52" },
  { _id: 3, name: "Charlie", email: "charlie@gmail.com", batch: "FSD52" }
]);

// CODEKATA
db.codekata.insertMany([
  { user_id: 1, problems_solved: 150 },
  { user_id: 2, problems_solved: 120 },
  { user_id: 3, problems_solved: 90 }
]);

// ATTENDANCE
db.attendance.insertMany([
  { user_id: 1, date: ISODate("2020-10-16"), status: "Absent" },
  { user_id: 2, date: ISODate("2020-10-17"), status: "Absent" },
  { user_id: 3, date: ISODate("2020-10-18"), status: "Present" }
]);

// TOPICS
db.topics.insertMany([
  { _id: 1, name: "HTML", date: ISODate("2020-10-10") },
  { _id: 2, name: "CSS", date: ISODate("2020-10-15") },
  { _id: 3, name: "JavaScript", date: ISODate("2020-10-25") }
]);

// TASKS
db.tasks.insertMany([
  { topic_id: 1, user_id: 1, task_name: "HTML Task", submitted: true, due_date: ISODate("2020-
  { topic_id: 2, user_id: 1, task_name: "CSS Task", submitted: false, due_date: ISODate("2020-
  { topic_id: 2, user_id: 2, task_name: "CSS Task", submitted: false, due_date: ISODate("2020-
  { topic_id: 3, user_id: 3, task_name: "JS Task", submitted: true, due_date: ISODate("2020-10
]);

// COMPANY DRIVES
db.company_drives.insertMany([
  { _id: 1, company_name: "Google", drive_date: ISODate("2020-10-20") },
  { _id: 2, company_name: "Amazon", drive_date: ISODate("2020-10-30") },
  { _id: 3, company_name: "Microsoft", drive_date: ISODate("2020-11-10") }
]);

// DRIVE PARTICIPATION
```

```
db.drive_participation.insertMany([
  { user_id: 1, drive_id: 1 },
  { user_id: 2, drive_id: 1 },
  { user_id: 2, drive_id: 2 },
  { user_id: 3, drive_id: 3 }
]);

// MENTORS
db.mentors.insertMany([
  { _id: 1, name: "John", email: "john@zen.com" },
  { _id: 2, name: "Emma", email: "emma@zen.com" }
]);

// MENTOR ASSIGNMENT (16 mentees for mentor_id 1)
db.mentor_assignment.insertMany([
  { mentor_id: 1, user_id: 1 },
  { mentor_id: 1, user_id: 2 },
  { mentor_id: 1, user_id: 3 },
  { mentor_id: 1, user_id: 4 },
  { mentor_id: 1, user_id: 5 },
  { mentor_id: 1, user_id: 6 },
  { mentor_id: 1, user_id: 7 },
  { mentor_id: 1, user_id: 8 },
  { mentor_id: 1, user_id: 9 },
  { mentor_id: 1, user_id: 10 },
  { mentor_id: 1, user_id: 11 },
  { mentor_id: 1, user_id: 12 },
  { mentor_id: 1, user_id: 13 },
  { mentor_id: 1, user_id: 14 },
  { mentor_id: 1, user_id: 15 },
  { mentor_id: 1, user_id: 16 }
]);

----------------------------
2. QUERIES (with command lines)
----------------------------

// 1. Topics and tasks taught in October
db.topics.find({ date: { $gte: ISODate("2020-10-01"), $lte: ISODate("2020-10-31") } });
db.tasks.find({ due_date: { $gte: ISODate("2020-10-01"), $lte: ISODate("2020-10-31") } });

// 2. Company drives between 15-Oct and 31-Oct 2020
db.company_drives.find({ drive_date: { $gte: ISODate("2020-10-15"), $lte: ISODate("2020-10-31"

// 3. Company drives and students who appeared
db.drive_participation.aggregate([
  {
    $lookup: {
      from: "company_drives",
      localField: "drive_id",
      foreignField: "_id",
      as: "drive_info"
    }
  },
  {
    $lookup: {
      from: "users",
      localField: "user_id",
      foreignField: "_id",
      as: "user_info"
    }
  }
]);

// 4. Number of problems solved by each user
db.users.aggregate([
  {
    $lookup: {
      from: "codekata",
      localField: "_id",
      foreignField: "user_id",
      as: "codekata_data"
    }
  },
  {
    $project: {
      name: 1,
      problems_solved: { $first: "$codekata_data.problems_solved" }
    }
  }
]);

// 5. Mentors with mentees count > 15
db.mentor_assignment.aggregate([
  { $group: { _id: "$mentor_id", mentee_count: { $sum: 1 } } },
  { $match: { mentee_count: { $gt: 15 } } },
  {
    $lookup: {
      from: "mentors",
      localField: "_id",
      foreignField: "_id",
      as: "mentor_info"
    }
  }
]);

// 6. Users absent and task not submitted between 15-Oct and 31-Oct
db.attendance.aggregate([
  {
    $match: {
      date: { $gte: ISODate("2020-10-15"), $lte: ISODate("2020-10-31") },
      status: "Absent"
    }
  },
  {
    $lookup: {
      from: "tasks",
      let: { uid: "$user_id" },
      pipeline: [
        {
          $match: {
            $expr: {
              $and: [
                { $eq: ["$user_id", "$$uid"] },
                { $eq: ["$submitted", false] },
                { $gte: ["$due_date", ISODate("2020-10-15")] },
                { $lte: ["$due_date", ISODate("2020-10-31")] }
              ]
            }
          }
        }
      ],
      as: "unsubmitted_tasks"
    }
  },
  { $match: { unsubmitted_tasks: { $ne: [] } } },
  { $count: "user_count" }
]);
```