Design database for Zen class programme

> users :-

```
> db.users.insert({
...   studentID: "GSID1001",
...   name: "Sarah Walker",
...   placements: "Appeared",
...   codekatasolved: 123 })
WriteResult({ "nInserted" : 1 })
```

```
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1002", name: "John Casey", placements: "Not Appeared", codekatasolved: 138 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1003", name: "Bryce Larkin", placements: "Not Appeared", codekatasolved: 131 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1004", name: "Morgan Olvaria", placements: "Appeared", codekatasolved: 112 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1005", name: "Dinesh Patil", placements: "Appeared", codekatasolved: 134 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1006", name: "Amelia Girsby", placements: "Appeared", codekatasolved: 146 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1007", name: "Neville Bones", placements: "Appeared", codekatasolved: 149 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1008", name: "Tim Tabbers", placements: "Appeared", codekatasolved: 126 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1009", name: "Ginny Pewters", placements: "Appeared", codekatasolved: 108 })
WriteResult({ "nInserted" : 1 })
> db.users.insert({ studentID: "GSID1010", name: "Chuck Duville", placements: "Appeared", codekatasolved: 134 })
WriteResult({ "nInserted" : 1 })
```

```
> db.users.find().pretty()
{
        "_id" : ObjectId("625a87d886fe8b72d0652e3e"),
        "studentID" : "GSID1001",
        "name" : "Sarah Walker",
        "placements" : "Appeared",
        "codekatasolved" : 123
}
{
        "_id" : ObjectId("625a880286fe8b72d0652e3f"),
        "studentID" : "GSID1002",
        "name" : "John Casey",
        "placements" : "Not Appeared",
        "codekatasolved" : 138
}
{
        "_id" : ObjectId("625a8bf086fe8b72d0652e40"),
        "studentID" : "GSID1003",
        "name" : "Bryce Larkin",
        "placements" : "Not Appeared",
        "codekatasolved" : 131
}
{
        "_id" : ObjectId("625a8c1986fe8b72d0652e41"),
        "studentID" : "GSID1004",
        "name" : "Morgan Olvaria",
        "placements" : "Appeared",
        "codekatasolved" : 112
}
{
        "_id" : ObjectId("625a8c3286fe8b72d0652e42"),
        "studentID" : "GSID1005",
        "name" : "Dinesh Patil",
        "placements" : "Appeared",
        "codekatasolved" : 134
}
```

```json
{
        "_id" : ObjectId("625a8c4786fe8b72d0652e43"),
        "studentID" : "GSID1006",
        "name" : "Amelia Girsby",
        "placements" : "Appeared",
        "codekatasolved" : 146
}
{
        "_id" : ObjectId("625a8c5c86fe8b72d0652e44"),
        "studentID" : "GSID1007",
        "name" : "Neville Bones",
        "placements" : "Appeared",
        "codekatasolved" : 149
}
{
        "_id" : ObjectId("625a8c6f86fe8b72d0652e45"),
        "studentID" : "GSID1008",
        "name" : "Tim Tabbers",
        "placements" : "Appeared",
        "codekatasolved" : 126
}
{
        "_id" : ObjectId("625a8c8986fe8b72d0652e46"),
        "studentID" : "GSID1009",
        "name" : "Ginny Pewters",
        "placements" : "Appeared",
        "codekatasolved" : 108
}
{
        "_id" : ObjectId("625a8ca286fe8b72d0652e47"),
        "studentID" : "GSID1010",
        "name" : "Chuck Duville",
        "placements" : "Appeared",
        "codekatasolved" : 134
}
```

> codekata:-

```
> db.codekata.insert({ studentID: "GSID1001", name:"Sarah Walker", questionSolved: 123, questionsAttempted: 141 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1002", name:"John Casey", questionSolved: 138, questionsAttempted: 157 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1003", name:"Bryce Larkin", questionSolved: 131, questionsAttempted: 148 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1004", name:"Morgan Olvaria", questionSolved: 112, questionsAttempted: 125 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1005", name:"Dinesh Patil", questionSolved: 134, questionsAttempted: 151 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1006", name:"Amelia Girsby", questionSolved: 146, questionsAttempted: 161 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1007", name:"Neville Bones", questionSolved: 149, questionsAttempted: 171 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1008", name:"Tim Tabbers", questionSolved: 126, questionsAttempted: 153 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1009", name:"Ginny Pewters", questionSolved: 108, questionsAttempted: 149 })
WriteResult({ "nInserted" : 1 })
> db.codekata.insert({ studentID: "GSID1010", name:"Chuck Duville", questionSolved: 134, questionsAttempted: 139 })
WriteResult({ "nInserted" : 1 })
> db.codekata.find().pretty()
```

```
> db.codekata.find().pretty()
{
        "_id" : ObjectId("62595fb2ac066c6dfdad2b94"),
        "studentID" : "GSID1001",
        "name" : "Sarah Walker",
        "questionSolved" : 123,
        "questionsAttempted" : 141
}
{
        "_id" : ObjectId("62595fcbac066c6dfdad2b95"),
        "studentID" : "GSID1002",
        "name" : "John Casey",
        "questionSolved" : 138,
        "questionsAttempted" : 157
}
{
        "_id" : ObjectId("62595fe2ac066c6dfdad2b96"),
        "studentID" : "GSID1003",
        "name" : "Bryce Larkin",
        "questionSolved" : 131,
        "questionsAttempted" : 148
}
{
        "_id" : ObjectId("62595ffdac066c6dfdad2b97"),
        "studentID" : "GSID1004",
        "name" : "Morgan Olvaria",
        "questionSolved" : 112,
        "questionsAttempted" : 125
}
{
        "_id" : ObjectId("62596023ac066c6dfdad2b98"),
        "studentID" : "GSID1005",
        "name" : "Dinesh Patil",
        "questionSolved" : 134,
        "questionsAttempted" : 151
}
```

```
{
        "_id" : ObjectId("62596040ac066c6dfdad2b99"),
        "studentID" : "GSID1006",
        "name" : "Amelia Girsby",
        "questionSolved" : 146,
        "questionsAttempted" : 161
}
{
        "_id" : ObjectId("62596055ac066c6dfdad2b9a"),
        "studentID" : "GSID1007",
        "name" : "Neville Bones",
        "questionSolved" : 149,
        "questionsAttempted" : 171
}
{
        "_id" : ObjectId("62596073ac066c6dfdad2b9b"),
        "studentID" : "GSID1008",
        "name" : "Tim Tabbers",
        "questionSolved" : 126,
        "questionsAttempted" : 153
}
{
        "_id" : ObjectId("62596096ac066c6dfdad2b9c"),
        "studentID" : "GSID1009",
        "name" : "Ginny Pewters",
        "questionSolved" : 108,
        "questionsAttempted" : 149
}
{
        "_id" : ObjectId("625960ccac066c6dfdad2b9d"),
        "studentID" : "GSID1010",
        "name" : "Chuck Duville",
        "questionSolved" : 134,
        "questionsAttempted" : 139
}
```

> attendance:-

```
> db.attendance.insert({ studentID: "GSID1001", name: "Sarah Walker", classesAttended: 38, classesMissed: 7 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1002", name: "John Casey", classesAttended: 42, classesMissed: 3 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1003", name: "Bryce Larkin", classesAttended: 33, classesMissed: 12 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1004", name: "Morgan Olvaria", classesAttended: 37, classesMissed: 8 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1005", name: "Dinesh Patil", classesAttended: 44, classesMissed: 1 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1006", name: "Amelia Girsby", classesAttended: 45, classesMissed: 0 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1007", name: "Neville Bones", classesAttended: 40, classesMissed: 5 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1008", name: "Tim Tabbers", classesAttended: 39, classesMissed: 6 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1009", name: "Ginny Pewters", classesAttended: 41, classesMissed: 4 })
WriteResult({ "nInserted" : 1 })
> db.attendance.insert({ studentID: "GSID1010", name: "Chuck Duville", classesAttended: 45, classesMissed: 0 })
WriteResult({ "nInserted" : 1 })
```

```
> db.attendance.find().pretty()
{
        "_id" : ObjectId("625a927786fe8b72d0652e48"),
        "studentID" : "GSID1001",
        "name" : "Sarah Walker",
        "classesAttended" : 38,
        "classesMissed" : 7
}
{
        "_id" : ObjectId("625a92b386fe8b72d0652e49"),
        "studentID" : "GSID1002",
        "name" : "John Casey",
        "classesAttended" : 42,
        "classesMissed" : 3
}
{
        "_id" : ObjectId("625a92d286fe8b72d0652e4a"),
        "studentID" : "GSID1003",
        "name" : "Bryce Larkin",
        "classesAttended" : 33,
        "classesMissed" : 12
}
{
        "_id" : ObjectId("625a92f286fe8b72d0652e4b"),
        "studentID" : "GSID1004",
        "name" : "Morgan Olvaria",
        "classesAttended" : 37,
        "classesMissed" : 8
}
{
        "_id" : ObjectId("625a930f86fe8b72d0652e4c"),
        "studentID" : "GSID1005",
        "name" : "Dinesh Patil",
        "classesAttended" : 44,
        "classesMissed" : 1
}
```

```
{
        "_id" : ObjectId("625a932886fe8b72d0652e4d"),
        "studentID" : "GSID1006",
        "name" : "Amelia Girsby",
        "classesAttended" : 45,
        "classesMissed" : 0
}
{
        "_id" : ObjectId("625a934586fe8b72d0652e4e"),
        "studentID" : "GSID1007",
        "name" : "Neville Bones",
        "classesAttended" : 40,
        "classesMissed" : 5
}
{
        "_id" : ObjectId("625a935d86fe8b72d0652e4f"),
        "studentID" : "GSID1008",
        "name" : "Tim Tabbers",
        "classesAttended" : 39,
        "classesMissed" : 6
}
{
        "_id" : ObjectId("625a937a86fe8b72d0652e50"),
        "studentID" : "GSID1009",
        "name" : "Ginny Pewters",
        "classesAttended" : 41,
        "classesMissed" : 4
}
{
        "_id" : ObjectId("625a939186fe8b72d0652e51"),
        "studentID" : "GSID1010",
        "name" : "Chuck Duville",
        "classesAttended" : 45,
        "classesMissed" : 0
}
```

> topics:-

```
> use day38task
switched to db day38task
> db.topics.insert({
... topicID: "GD101",
... name: "Front-End Ddevelopment",
... description: "Usage of scripting language to develop basic landing page",
... month_taught: "September",
```

```
> db.topics.insert({ topicID: "GD102", name: "SQL Database Development", description: "Usage of mySQL to develop basic SQL database", month_taught: "October", credits: 3 })
WriteResult({ "nInserted" : 1 })
> db.topics.insert({ topicID: "GD103", name: "MongoDB Database Development", description: "Usage of mongoDB to develop basic unstructured database", month_taught: "October", credits: 3 })
WriteResult({ "nInserted" : 1 })
> db.topics.insert({ topicID: "GD104", name: "NodeJS express Server", description: "Usage of express module to develop a basic server", month_taught: "November", credits: 2 })
WriteResult({ "nInserted" : 1 })
> db.topics.insert({ topicID: "GD105", name: "Data Structures and Algorithms", description: "Understanding basic data structures and algorithms", month_taught: "November", credits: 3 })
WriteResult({ "nInserted" : 1 })
> db.topics.find().pretty()
```

```
> db.topics.find().pretty()
{
        "_id" : ObjectId("62594999ac066c6dfdad2b81"),
        "topicID" : "GD101",
        "name" : "Front-End Ddevelopment",
        "description" : "Usage of scripting language to develop basic landing page",
        "month_taught" : "September",
        "credits" : 4
}
{
        "_id" : ObjectId("62594b41ac066c6dfdad2b83"),
        "topicID" : "GD102",
        "name" : "SQL Database Development",
        "description" : "Usage of mySQL to develop basic SQL database",
        "month_taught" : "October",
        "credits" : 3
}
{
        "_id" : ObjectId("62594b64ac066c6dfdad2b84"),
        "topicID" : "GD103",
        "name" : "MongoDB Database Development",
        "description" : "Usage of mongoDB to develop basic unstructured database",
        "month_taught" : "October",
        "credits" : 3
}
{
        "_id" : ObjectId("62594badac066c6dfdad2b85"),
        "topicID" : "GD104",
        "name" : "NodeJS express Server",
        "description" : "Usage of express module to develop a basic server",
        "month_taught" : "November",
        "credits" : 2
}
{
        "_id" : ObjectId("62594bebac066c6dfdad2b86"),
        "topicID" : "GD105",
        "name" : "Data Structures and Algorithms",
        "description" : "Understanding basic data structures and algorithms",
        "month_taught" : "November",
        "credits" : 3
}
```

> tasks :-

```
> db.tasks_october.insert({ taskID: "TD101", task: "React SignUp Form", dueDate: 5, submissions: 14, totalStudents: 18, absentees: 4 })
WriteResult({ "nInserted" : 1 })
> db.tasks_october.insert({ taskID: "TD102", task: "React Landing Page", dueDate: 12, submissions: 16, totalStudents: 18, absentees: 2 })
WriteResult({ "nInserted" : 1 })
> db.tasks_october.insert({ taskID: "TD103", task: "NodeJS Form Validation", dueDate: 16, submissions: 11, totalStudents: 18, absentees: 7 })
WriteResult({ "nInserted" : 1 })
> db.tasks_october.insert({ taskID: "TD104", task: "NodeJS Auth Login PAge", dueDate: 20, submissions: 13, totalStudents: 18, absentees: 5 })
WriteResult({ "nInserted" : 1 })
> db.tasks_october.insert({ taskID: "TD105", task: "Gmail Clone Database Design", dueDate: 25, submissions: 17, totalStudents: 18, absentees: 1 })
WriteResult({ "nInserted" : 1 })
> db.tasks_october.insert({ taskID: "TD106", task: "Stack Overflow Database Design", dueDate: 29, submissions: 16, totalStudents: 18, absentees: 2 })
WriteResult({ "nInserted" : 1 })
```

```
> db.tasks_october.find().pretty()
{
        "_id" : ObjectId("62595173ac066c6dfdad2b87"),
        "taskID" : "TD101",
        "task" : "React SignUp Form",
        "dueDate" : 5,
        "submissions" : 14,
        "totalStudents" : 18,
        "absentees" : 4
}
{
        "_id" : ObjectId("62595190ac066c6dfdad2b88"),
        "taskID" : "TD102",
        "task" : "React Landing Page",
        "dueDate" : 12,
        "submissions" : 16,
        "totalStudents" : 18,
        "absentees" : 2
}
{
        "_id" : ObjectId("625951c3ac066c6dfdad2b89"),
        "taskID" : "TD103",
        "task" : "NodeJS Form Validation",
        "dueDate" : 16,
        "submissions" : 11,
        "totalStudents" : 18,
        "absentees" : 7
}
{
        "_id" : ObjectId("625951f6ac066c6dfdad2b8a"),
        "taskID" : "TD104",
        "task" : "NodeJS Auth Login PAge",
        "dueDate" : 20,
        "submissions" : 13,
        "totalStudents" : 18,
        "absentees" : 5
}
{
        "_id" : ObjectId("62595221ac066c6dfdad2b8b"),
        "taskID" : "TD105",
        "task" : "Gmail Clone Database Design",
        "dueDate" : 25,
        "submissions" : 17,
        "totalStudents" : 18,
        "absentees" : 1
}
{
        "_id" : ObjectId("62595241ac066c6dfdad2b8c"),
        "taskID" : "TD106",
        "task" : "Stack Overflow Database Design",
        "dueDate" : 29,
        "submissions" : 16,
        "totalStudents" : 18,
        "absentees" : 2
}
>
```

> company_drives:-

```
> db.company_drives.insert(
... {
... companyID: "CD7001",
... companyName: "Samsung",
... spotAvailability: 6,
... applicantCount: 23,
... interviewMonth: "September",
... interviewDate: 24 })
WriteResult({ "nInserted" : 1 })
```

```
> db.company_drives.insert( { companyID: "CD7002", companyName: "Greenlabz", spotAvailability: 5, applicantCount: 14, interviewMonth: "October", interviewDate: 2 })
WriteResult({ "nInserted" : 1 })
> db.company_drives.insert( { companyID: "CD7003", companyName: "Amazon", spotAvailability: 4, applicantCount: 28, interviewMonth: "October", interviewDate: 8 })
WriteResult({ "nInserted" : 1 })
> db.company_drives.insert( { companyID: "CD7005", companyName: "Systech", spotAvailability: 2, applicantCount: 10, interviewMonth: "October", interviewDate: 16 })
WriteResult({ "nInserted" : 1 })
> db.company_drives.insert( { companyID: "CD7006", companyName: "Capegemini", spotAvailability: 8, applicantCount: 18, interviewMonth: "October", interviewDate: 21 })
WriteResult({ "nInserted" : 1 })
> db.company_drives.insert( { companyID: "CD7007", companyName: "Facebook", spotAvailability: 1, applicantCount: 6, interviewMonth: "October", interviewDate: 28 })
WriteResult({ "nInserted" : 1 })
> db.company_drives.insert( { companyID: "CD7008", companyName: "Symbiosos", spotAvailability: 20, applicantCount: 32, interviewMonth: "November", interviewDate: 02 })
WriteResult({ "nInserted" : 1 })
```

```
> db.company_drives.find().pretty()
{
        "_id" : ObjectId("625a79a486fe8b72d0652e37"),
        "companyID" : "CD7001",
        "companyName" : "Samsung",
        "spotAvailability" : 6,
        "applicantCount" : 23,
        "interviewMonth" : "September",
        "interviewDate" : 24
}
{
        "_id" : ObjectId("625a79d286fe8b72d0652e38"),
        "companyID" : "CD7002",
        "companyName" : "Greenlabz",
        "spotAvailability" : 5,
        "applicantCount" : 14,
        "interviewMonth" : "October",
        "interviewDate" : 2
}
{
        "_id" : ObjectId("625a79e986fe8b72d0652e39"),
        "companyID" : "CD7003",
        "companyName" : "Amazon",
        "spotAvailability" : 4,
        "applicantCount" : 28,
        "interviewMonth" : "October",
        "interviewDate" : 8
}
{
        "_id" : ObjectId("625a7a0c86fe8b72d0652e3a"),
        "companyID" : "CD7005",
        "companyName" : "Systech",
        "spotAvailability" : 2,
        "applicantCount" : 10,
        "interviewMonth" : "October",
        "interviewDate" : 16
}
{
        "_id" : ObjectId("625a7a2686fe8b72d0652e3b"),
        "companyID" : "CD7006",
        "companyName" : "Capegemini",
        "spotAvailability" : 8,
        "applicantCount" : 18,
        "interviewMonth" : "October",
        "interviewDate" : 21
}
{
        "_id" : ObjectId("625a7a4886fe8b72d0652e3c"),
        "companyID" : "CD7007",
        "companyName" : "Facebook",
        "spotAvailability" : 1,
        "applicantCount" : 6,
        "interviewMonth" : "October",
        "interviewDate" : 28
}
{
        "_id" : ObjectId("625a7a6986fe8b72d0652e3d"),
        "companyID" : "CD7008",
        "companyName" : "Symbiosos",
        "spotAvailability" : 20,
        "applicantCount" : 32,
        "interviewMonth" : "November",
        "interviewDate" : 2
}
```

> mentors:-

```
> db.mentors.insert({ mentorID: "MTR1001", name:"James Keith", age:34, experience_in_years: 11, totalStudents: 14, specialization:"Data Science" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1002", name:"Kiera Anders", age:31, experience_in_years: 6, totalStudents: 18, specialization:"Data Science" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1003", name:"Toby McSanders", age:35, experience_in_years: 8, totalStudents: 28, specialization:"Web Development" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1004", name:"Jacob Villes", age:30, experience_in_years: 7, totalStudents: 25, specialization:"FrontEnd Development" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1005", name:"Tammy jones", age:28, experience_in_years: 3, totalStudents: 12, specialization:"FrontEnd Development" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1006", name:"Tobias Tills", age:34, experience_in_years: 8, totalStudents: 22, specialization:"BackEnd Development" })
WriteResult({ "nInserted" : 1 })
> db.mentors.insert({ mentorID: "MTR1007", name:"Samira Duilles", age:29, experience_in_years: 4, totalStudents: 13, specialization:"BackEnd Development" })
WriteResult({ "nInserted" : 1 })
```

```
> db.mentors.find().pretty()
{
        "_id" : ObjectId("62595be3ac066c6dfdad2b8d"),
        "mentorID" : "MTR1001",
        "name" : "James Keith",
        "age" : 34,
        "experience_in_years" : 11,
        "totalStudents" : 14,
        "specialization" : "Data Science"
}
{

        "_id" : ObjectId("62595c04ac066c6dfdad2b8e"),
        "mentorID" : "MTR1002",
        "name" : "Kiera Anders",
        "age" : 31,
        "experience_in_years" : 6,
        "totalStudents" : 18,
        "specialization" : "Data Science"
}
{

        "_id" : ObjectId("62595c2eac066c6dfdad2b8f"),
        "mentorID" : "MTR1003",
        "name" : "Toby McSanders",
        "age" : 35,
        "experience_in_years" : 8,
        "totalStudents" : 28,
        "specialization" : "Web Development"
}
{

        "_id" : ObjectId("62595c55ac066c6dfdad2b90"),
        "mentorID" : "MTR1004",
        "name" : "Jacob Villes",
        "age" : 30,
        "experience_in_years" : 7,
        "totalStudents" : 25,
        "specialization" : "FrontEnd Development"
}
{

        "_id" : ObjectId("62595c71ac066c6dfdad2b91"),
        "mentorID" : "MTR1005",
        "name" : "Tammy jones",
        "age" : 28,
        "experience_in_years" : 3,
        "totalStudents" : 12,
        "specialization" : "FrontEnd Development"
}
{

        "_id" : ObjectId("62595ca3ac066c6dfdad2b92"),
        "mentorID" : "MTR1006",
        "name" : "Tobias Tills",
        "age" : 34,
        "experience_in_years" : 8,
        "totalStudents" : 22,
        "specialization" : "BackEnd Development"
}
{

        "_id" : ObjectId("62595cc5ac066c6dfdad2b93"),
        "mentorID" : "MTR1007",
        "name" : "Samira Duilles",
        "age" : 29,
```

QUESTIONS:

Q1> Find all the topics and tasks which are thought in the month of October

```
> db.topics.find({"month_taught":"October"}).pretty()
{
        "_id" : ObjectId("62594b41ac066c6dfdad2b83"),
        "topicID" : "GD102",
        "name" : "SQL Database Development",
        "description" : "Usage of mySQL to develop basic SQL database",
        "month_taught" : "October",
        "credits" : 3
}
{
        "_id" : ObjectId("62594b64ac066c6dfdad2b84"),
        "topicID" : "GD103",
        "name" : "MongoDB Database Development",
        "description" : "Usage of mongoDB to develop basic unstructured database",
        "month_taught" : "October",
        "credits" : 3
}
```

```
> db.tasks_october.find().pretty()
{
        "_id" : ObjectId("62595173ac066c6dfdad2b87"),
        "taskID" : "TD101",
        "task" : "React SignUp Form",
        "dueDate" : 5,
        "submissions" : 14,
        "totalStudents" : 18,
        "absentees" : 4
}
{
        "_id" : ObjectId("62595190ac066c6dfdad2b88"),
        "taskID" : "TD102",
        "task" : "React Landing Page",
        "dueDate" : 12,
        "submissions" : 16,
        "totalStudents" : 18,
        "absentees" : 2
}
{
        "_id" : ObjectId("625951c3ac066c6dfdad2b89"),
        "taskID" : "TD103",
        "task" : "NodeJS Form Validation",
        "dueDate" : 16,
        "submissions" : 11,
        "totalStudents" : 18,
        "absentees" : 7
}
{
        "_id" : ObjectId("625951f6ac066c6dfdad2b8a"),
        "taskID" : "TD104",
        "task" : "NodeJS Auth Login PAge",
        "dueDate" : 20,
        "submissions" : 13,
        "totalStudents" : 18,
        "absentees" : 5
}
{
        "_id" : ObjectId("62595221ac066c6dfdad2b8b"),
        "taskID" : "TD105",
        "task" : "Gmail Clone Database Design",
        "dueDate" : 25,
        "submissions" : 17,
        "totalStudents" : 18,
        "absentees" : 1
}
{
        "_id" : ObjectId("62595241ac066c6dfdad2b8c"),
        "taskID" : "TD106",
        "task" : "Stack Overflow Database Design",
        "dueDate" : 29,
        "submissions" : 16,
        "totalStudents" : 18,
        "absentees" : 2
}
```

Q2> Find all the company drives which appeared between 15 oct-2020 and 31-oct-2020

```
> db.company_drives.find({interviewDate:{$gt:15,$lt:31},interviewMonth:"October"}).pretty()
{
        "_id" : ObjectId("625a7a0c86fe8b72d0652e3a"),
        "companyID" : "CD7005",
        "companyName" : "Systech",
        "spotAvailability" : 2,
        "applicantCount" : 10,
        "interviewMonth" : "October",
        "interviewDate" : 16
}
{
        "_id" : ObjectId("625a7a2686fe8b72d0652e3b"),
        "companyID" : "CD7006",
        "companyName" : "Capegemini",
        "spotAvailability" : 8,
        "applicantCount" : 18,
        "interviewMonth" : "October",
        "interviewDate" : 21
}
{
        "_id" : ObjectId("625a7a4886fe8b72d0652e3c"),
        "companyID" : "CD7007",
        "companyName" : "Facebook",
        "spotAvailability" : 1,
        "applicantCount" : 6,
        "interviewMonth" : "October",
        "interviewDate" : 28
}
```

Q3> Find all the company drives and students who are appeared for the placement.

```
> db.company_drives.find().pretty()
{
        "_id" : ObjectId("625a79a486fe8b72d0652e37"),
        "companyID" : "CD7001",
        "companyName" : "Samsung",
        "spotAvailability" : 6,
        "applicantCount" : 23,
        "interviewMonth" : "September",
        "interviewDate" : 24
}
{
        "_id" : ObjectId("625a79d286fe8b72d0652e38"),
        "companyID" : "CD7002",
        "companyName" : "Greenlabz",
        "spotAvailability" : 5,
        "applicantCount" : 14,
        "interviewMonth" : "October",
        "interviewDate" : 2
}
{
        "_id" : ObjectId("625a79e986fe8b72d0652e39"),
        "companyID" : "CD7003",
        "companyName" : "Amazon",
        "spotAvailability" : 4,
        "applicantCount" : 28,
        "interviewMonth" : "October",
        "interviewDate" : 8
}
{
        "_id" : ObjectId("625a7a0c86fe8b72d0652e3a"),
        "companyID" : "CD7005",
        "companyName" : "Systech",
        "spotAvailability" : 2,
        "applicantCount" : 10,
        "interviewMonth" : "October",
        "interviewDate" : 16
}
{
        "_id" : ObjectId("625a7a2686fe8b72d0652e3b"),
        "companyID" : "CD7006",
        "companyName" : "Capegemini",
        "spotAvailability" : 8,
        "applicantCount" : 18,
        "interviewMonth" : "October",
        "interviewDate" : 21
}
{
        "_id" : ObjectId("625a7a4886fe8b72d0652e3c"),
        "companyID" : "CD7007",
        "companyName" : "Facebook",
        "spotAvailability" : 1,
        "applicantCount" : 6,
        "interviewMonth" : "October",
        "interviewDate" : 28
}
{
        "_id" : ObjectId("625a7a6986fe8b72d0652e3d"),
        "companyID" : "CD7008",
        "companyName" : "Symbiosos",
        "spotAvailability" : 20,
        "applicantCount" : 32,
        "interviewMonth" : "November",
        "interviewDate" : 2
}
```

```
> db.users.find({"placements":"Appeared"}).pretty()
{
        "_id" : ObjectId("625a87d886fe8b72d0652e3e"),
        "studentID" : "GSID1001",
        "name" : "Sarah Walker",
        "placements" : "Appeared",
        "codekatasolved" : 123
}
{
        "_id" : ObjectId("625a8c1986fe8b72d0652e41"),
        "studentID" : "GSID1004",
        "name" : "Morgan Olvaria",
        "placements" : "Appeared",
        "codekatasolved" : 112
}
{
        "_id" : ObjectId("625a8c3286fe8b72d0652e42"),
        "studentID" : "GSID1005",
        "name" : "Dinesh Patil",
        "placements" : "Appeared",
        "codekatasolved" : 134
}
{
        "_id" : ObjectId("625a8c4786fe8b72d0652e43"),
        "studentID" : "GSID1006",
        "name" : "Amelia Girsby",
        "placements" : "Appeared",
        "codekatasolved" : 146
}
{
        "_id" : ObjectId("625a8c5c86fe8b72d0652e44"),
        "studentID" : "GSID1007",
        "name" : "Neville Bones",
        "placements" : "Appeared",
        "codekatasolved" : 149
}
{
        "_id" : ObjectId("625a8c6f86fe8b72d0652e45"),
        "studentID" : "GSID1008",
        "name" : "Tim Tabbers",
        "placements" : "Appeared",
        "codekatasolved" : 126
}
{
        "_id" : ObjectId("625a8c8986fe8b72d0652e46"),
        "studentID" : "GSID1009",
        "name" : "Ginny Pewters",
        "placements" : "Appeared",
        "codekatasolved" : 108
}
{
        "_id" : ObjectId("625a8ca286fe8b72d0652e47"),
        "studentID" : "GSID1010",
        "name" : "Chuck Duville",
        "placements" : "Appeared",
        "codekatasolved" : 134
}
```

Q4> Find the number of problems solved by the user in codekata

```
> db.codekata.find().pretty()
{
        "_id" : ObjectId("62595fb2ac066c6dfdad2b94"),
        "studentID" : "GSID1001",
        "name" : "Sarah Walker",
        "questionSolved" : 123,
        "questionsAttempted" : 141
}
{
        "_id" : ObjectId("62595fcbac066c6dfdad2b95"),
        "studentID" : "GSID1002",
        "name" : "John Casey",
        "questionSolved" : 138,
        "questionsAttempted" : 157
}
{
        "_id" : ObjectId("62595fe2ac066c6dfdad2b96"),
        "studentID" : "GSID1003",
        "name" : "Bryce Larkin",
        "questionSolved" : 131,
        "questionsAttempted" : 148
}
{
        "_id" : ObjectId("62595ffdac066c6dfdad2b97"),
        "studentID" : "GSID1004",
        "name" : "Morgan Olvaria",
        "questionSolved" : 112,
        "questionsAttempted" : 125
}
{
        "_id" : ObjectId("62596023ac066c6dfdad2b98"),
        "studentID" : "GSID1005",
        "name" : "Dinesh Patil",
        "questionSolved" : 134,
        "questionsAttempted" : 151
}
```

```
{
        "_id" : ObjectId("62596040ac066c6dfdad2b99"),
        "studentID" : "GSID1006",
        "name" : "Amelia Girsby",
        "questionSolved" : 146,
        "questionsAttempted" : 161
}
{
        "_id" : ObjectId("62596055ac066c6dfdad2b9a"),
        "studentID" : "GSID1007",
        "name" : "Neville Bones",
        "questionSolved" : 149,
        "questionsAttempted" : 171
}
{
        "_id" : ObjectId("62596073ac066c6dfdad2b9b"),
        "studentID" : "GSID1008",
        "name" : "Tim Tabbers",
        "questionSolved" : 126,
        "questionsAttempted" : 153
}
{
        "_id" : ObjectId("62596096ac066c6dfdad2b9c"),
        "studentID" : "GSID1009",
        "name" : "Ginny Pewters",
        "questionSolved" : 108,
        "questionsAttempted" : 149
}
{
        "_id" : ObjectId("625960ccac066c6dfdad2b9d"),
        "studentID" : "GSID1010",
        "name" : "Chuck Duville",
        "questionSolved" : 134,
        "questionsAttempted" : 139
}
```

Q5> Find all the mentors with who has the mentee's count more than 15

```
> db.mentors.find({"totalStudents":{$gt:15}}).pretty()
{
        "_id" : ObjectId("62595c04ac066c6dfdad2b8e"),
        "mentorID" : "MTR1002",
        "name" : "Kiera Anders",
        "age" : 31,
        "experience_in_years" : 6,
        "totalStudents" : 18,
        "specialization" : "Data Science"
}
{

        "_id" : ObjectId("62595c2eac066c6dfdad2b8f"),
        "mentorID" : "MTR1003",
        "name" : "Toby McSanders",
        "age" : 35,
        "experience_in_years" : 8,
        "totalStudents" : 28,
        "specialization" : "Web Development"
}
{

        "_id" : ObjectId("62595c55ac066c6dfdad2b90"),
        "mentorID" : "MTR1004",
        "name" : "Jacob Villes",
        "age" : 30,
        "experience_in_years" : 7,
        "totalStudents" : 25,
        "specialization" : "FrontEnd Development"
}
{

        "_id" : ObjectId("62595ca3ac066c6dfdad2b92"),
        "mentorID" : "MTR1006",
        "name" : "Tobias Tills",
        "age" : 34,
        "experience_in_years" : 8,
        "totalStudents" : 22,
        "specialization" : "BackEnd Development"
}
```

Q6> Find the number of users who are absent and task is not submitted   between 15 oct-2020 and 31-oct-2020

```
> db.tasks_october.find({dueDate:{$gt:15,$lt:31}}).pretty()
{
        "_id" : ObjectId("625951c3ac066c6dfdad2b89"),
        "taskID" : "TD103",
        "task" : "NodeJS Form Validation",
        "dueDate" : 16,
        "submissions" : 11,
        "totalStudents" : 18,
        "absentees" : 7
}
{
        "_id" : ObjectId("625951f6ac066c6dfdad2b8a"),
        "taskID" : "TD104",
        "task" : "NodeJS Auth Login PAge",
        "dueDate" : 20,
        "submissions" : 13,
        "totalStudents" : 18,
        "absentees" : 5
}
{
        "_id" : ObjectId("62595221ac066c6dfdad2b8b"),
        "taskID" : "TD105",
        "task" : "Gmail Clone Database Design",
        "dueDate" : 25,
        "submissions" : 17,
        "totalStudents" : 18,
        "absentees" : 1
}
{
        "_id" : ObjectId("62595241ac066c6dfdad2b8c"),
        "taskID" : "TD106",
        "task" : "Stack Overflow Database Design",
        "dueDate" : 29,
        "submissions" : 16,
        "totalStudents" : 18,
        "absentees" : 2
}
>
```