//Design database for Zen class program

//Zen Database

//use zendb

//1.Insert data for Users Collection

db.Users.insertMany([

    {

        userid: 1,

        name: "Thoufiq",

        email: "Thoufiq@gmail.com",

        mentorid: 1,

    },

    {

        userid: 2,

        name: "Irshath",

        email: "Irshath@gmail.com",

        mentorid: 2,

    },

    {

        userid: 3,

        name: "Ahamed",

        email: "Ahamed@gmail.com",

        mentorid: 3,

    },

    {

        userid: 4,

        name: "Mohammed",

        email: "Mohammed@gmail.com",

        mentorid: 1,

    },

    {

        userid: 5,

        name: "Imran",

        email: "Imran@gmail.com",

        mentorid: 2,

    },

]);

//2.Insert data for Codekata Collection

db.Codekata.insertMany([

    {

        userid: 1,

        problems: 10,

    },

    {

        userid: 2,

        problems: 15,

    },

    {

        userid: 3,

        problems: 20,

    },

    {

        userid: 4,

        problems: 25,

    },

    {

        userid: 5,

        problems: 35,

    },

]);

//3.Insert data for Attendance Collection

db.Attendance.insertMany([

    {

        userid: 1,

        topicid: 1,

        attended: true

    },

    {

        userid: 2,

        topicid: 2,

        attended: false

    },

    {

        userid: 3,

        topicid: 3,

        attended: true

    },

    {

        userid: 4,

        topicid: 4,

        attended: false

    },

    {

        userid: 5,

        topicid: 5,

        attended: true

    },

]);

//4.Insert data for Topics Collection

db.Topics.insertMany([

    {

        topicid: 1,

        topic: "HTML",

        topic\_date: new Date("1-jan-2020"),

    },

    {

        topicid: 2,

        topic: "CSS",

        topic\_date: new Date("1-feb-2020"),

    },

    {

        topicid: 3,

        topic: "Javascript",

        topic\_date: new Date("1-mar-2020"),

    },

    {

        topicid: 4,

        topic: "React",

        topic\_date: new Date("1-apr-2020"),

    },

    {

        topicid: 5,

        topic: "NodeJs",

        topic\_date: new Date("1-may-2020"),

    },

]);

//5.Insert data for Tasks Collection

db.Tasks.insertMany([

    {

        taskid: 1,

        topicid: 1,

        userid: 1,

        task: "HTML",

        due\_date: new Date("2023-jan-15"),

        submitted: false

    },

    {

        taskid: 2,

        topicid: 2,

        userid: 2,

        task: "CSS",

        due\_date: new Date("2023-feb-15"),

        submitted: true

    },

    {

        taskid: 3,

        topicid: 3,

        userid: 3,

        task: "JavaScript",

        due\_date: new Date("2023-oct-15"),

        submitted: false

    },

    {

        taskid: 4,

        topicid: 4,

        userid: 4,

        task: "React",

        due\_date: new Date("2020-apr-15"),

        submitted: true

    },

    {

        taskid: 5,

        topicid: 5,

        userid: 5,

        task: "NodeJS",

        due\_date: new Date("2020-oct-15"),

        submitted: false

    },

]);

//6.Insert data for CompanyDrives Collection

db.CompanyDrives.insertMany([

    {

        drive\_id: 1,

        drive\_date: new Date("2023-jan-1"),

        company: "Meta",

        student\_appeared: "Thoufiq",

    },

    {

        drive\_id: 2,

        drive\_date: new Date("2023-feb-1"),

        company: "Amazon",

        student\_appeared: "Irshath",

    },

    {

        drive\_id: 3,

        drive\_date: new Date("2023-mar-1"),

        company: "Apple",

        student\_appeared: "Ahamed",

    },

    {

        drive\_id: 4,

        drive\_date: new Date("2020-oct-15"),

        company: "Netflix",

        student\_appeared: "Mohammed",

    },

    {

        drive\_id: 5,

        drive\_date: new Date("2020-oct-31"),

        company: "Google",

        student\_appeared: "Imran",

    },

]);

//7.Insert data for mentors Collection

db.Mentors.insertMany([

    {

        mentorid: 1,

        mentorname: "SriLakshmi",

        mentor\_email: "sri@gmail.com",

        mentee\_count: 10,

    },

    {

        mentorid: 2,

        mentorname: "Mahesh",

        mentor\_email: "mahesh@gmail.com",

        mentee\_count: 16,

    },

    {

        mentorid: 3,

        mentorname: "Lakshan",

        mentor\_email: "lakshan@gmail.com",

        mentee\_count: 16,

    },

    {

        mentorid: 4,

        mentorname: "Gokul",

        mentor\_email: "gokul@gmail.com",

        mentee\_count: 20,

    },

    {

        mentorid: 5,

        mentorname: "Dhanushya",

        mentor\_email: "dhanushya@gmail.com",

        mentee\_count: 16,

    },

]);

//Queries

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

db.Topics.aggregate([

    {

        $lookup: {

            from: "Tasks",

            localField: "topicid",

            foreignField: "topicid",

            as: "Taskinfo",

        },

    },

    {

        $match: {

            $and: [

                {

                    $or: [

                        { "topic\_date": { "$gte": ISODate("2020-10-01T18:30:00.000+00:00") } },

                        { "topic\_date": { "$lte": ISODate("2020-10-30T18:30:00.000+00:00") } },

                    ],

                },

                {

                    $or: [

                        { "Taskinfo.due\_date": { "$gte": ISODate("2020-10-01T18:30:00.000+00:00") } },

                        { "Taskinfo.due\_date": { "$lte": ISODate("2020-10-30T18:30:00.000+00:00") } },

                    ],

                },

            ],

        },

    },

]);

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

db.CompanyDrives.find({

    $and: [{ drive\_date: { "$gte": ISODate("2020-10-15T18:30:00.000+00:00") } }, { drive\_date: { "$lte": ISODate("2020-10-31T18:30:00.000+00:00") } }],

})

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

db.CompanyDrives.aggregate([{

    $lookup: { from: 'Users', localField: 'userid', foreignField: 'userid', as: 'studentInfo' }

},

{

    $project: {

        \_id: 0,

        company: 1,

        drive\_date: 1,

        student\_appeared: 1,

    },

},

]);

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

db.Codekata.aggregate([

    {

        $lookup: {

            from: 'Users', localField: 'userid', foreignField: 'userid', as: 'studentInfo'

        }

    },

    {

        $project: {

            \_id: 0,

            "studentInfo.name": 1,

            problems: 1,

            "studentInfo.email": 1,

            "studentInfo.userid": 1

        },

    },

]);

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

db.Mentors.aggregate([

    {

        $match: {

            mentee\_count: { $gt: 15 }

        }

    },

]);

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

db.Attendance.aggregate([{

    $lookup: {

        from: "Topics",

        localField: "topicid",

        foreignField: "topicid",

        as: "TopicInfo",

    },

},

{

    $lookup: {

        from: "Tasks",

        localField: "topicid",

        foreignField: "topicid",

        as: "TaskInfo",

    },

},

{ $match: { $and: [{ attended: false }, { "Tasks.submitted": false }] } },

{

    $match: {

        $and: [{

            $or: [

                { "Topics.topic\_date": { "$gte": ISODate("2020-10-15T18:30:00.000+00:00") } },

                { "Topics.topic\_date": { "$lte": ISODate("2020-10-31T18:30:00.000+00:00") } },

            ],

        },

        {

            $or: [

                { "Tasks.due\_date": { "$gte": ISODate("2020-10-15T18:30:00.000+00:00") } },

                { "Tasks.due\_date": { "$lte": ISODate("2020-10-31T18:30:00.000+00:00") } },

            ],

        },

        ],

    },

},

{

    $count: "TopicInfo",

},

]);