**Design database for Zen class programme**

1. users
2. codekata
3. attendance
4. topics
5. tasks
6. company\_drives
7. mentors

**Create database**

use zen\_class;

**Create collection and insert data – “USERS”**

db.createCollection("users");

db.users.insertMany([

{

userid: 1,

name: "Mahesh",

email: "Mahesh@gmail.com",

},

{

userid: 2,

name: "Surya",

email: "Surya@gmail.com",

},

{

userid: 3,

name: "Akash",

email: "Akash@gmail.com",

},

{

userid: 4,

name: "Mohan",

email: "Mohan@gmail.com",

},

{

userid: 5,

name: "Rohit",

email: "Rohit@gmail.com",

}

])

**Create collection and insert data – “CODEKATA”**

db.createCollection("codekata");

db.codekata.insertMany([

{

userid: 1,

problems: 30

},

{

userid: 2,

problems: 45

},

{

userid: 3,

problems: 90

},

{

userid: 4,

problems: 65

},

{

userid: 5,

problems: 80

}

])

**Create collection and insert data – “TOPICS”**

db.createCollection("topics");

db.topics.insertMany([

{

topicid: 1,

topic: "HTML",

topic\_date: new Date("18-oct-2020")

},

{

topicid: 2,

topic: "CSS",

topic\_date: new Date("28-oct-2020")

},

{

topicid: 3,

topic: "JavaScript",

topic\_date: new Date("05-nov-2020")

},

{

topicid: 4,

topic: "ReactJS",

topic\_date: new Date("15-nov-2020")

},

{

topicid: 5,

topic: "NodeJS",

topic\_date: new Date("25-nov-2020")

}

])

**Create collection and insert data – “TASKS”**

db.createCollection("tasks");

db.tasks.insertMany([{

taskid: 1,

topicid: 1,

userid: 1,

task: "HTML Task",

due\_date: new Date("18-oct-2020"),

submitted: true

},

{

taskid: 2,

topicid: 2,

userid: 2,

task: "CSS Task",

due\_date: new Date("28-oct-2020"),

submitted: true

},

{

taskid: 3,

topicid: 3,

userid: 3,

task: "Javascript Task",

due\_date: new Date("05-oct-2020"),

submitted: false

},

{

taskid: 4,

topicid: 4,

userid: 4,

task: "React Task",

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

submitted: true

},

{

taskid: 5,

topicid: 5,

userid: 5,

task: "NodeJS Task",

due\_date: new Date("25-nov-2020"),

submitted: false

}

])

**Create collection and insert data – “ATTENDANCE”**

db.createCollection("attendance");

db.attendance.insertMany([

{

userid: 1,

topicid: 2,

attended: true

},

{

userid: 2,

topicid: 1,

attended: true

},

{

userid: 3,

topicid: 5,

attended: false

},

{

userid: 4,

topicid: 3,

attended: true

},

{

userid: 5,

topicid: 4,

attended: false

}

])

**Create collection and insert data – “MENTORS”**

db.createCollection("mentors");

db.mentors.insertMany([

{

mentorid: 1,

mentorname: "Pavan",

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

mentee\_count: 20

},

{

mentorid: 2,

mentorname: "Manoj",

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

mentee\_count: 18

},

{

mentorid: 3,

mentorname: "Charan",

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

mentee\_count: 30

},

{

mentorid: 4,

mentorname: "Teja",

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

mentee\_count: 15

},

{

mentorid: 5,

mentorname: "Vicky",

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

mentee\_count: 20

}

])

**Create collection and insert data – “COMPANY DRIVES”**

db.createCollection("companydrives");

db.comapnydrives.insertMany([

{

userid: 1,

drive\_date: new Date("20-oct-2020"),

company: "Apple"

},

{

userid: 1,

drive\_date: new Date("22-oct-2020"),

company: "Amazon"

},

{

userid: 2,

drive\_date: new Date("25-oct-2020"),

company: "TCS"

},

{

userid: 3,

drive\_date: new Date("30-oct-2020"),

company: "BYJUS"

},

{

userid: 4,

drive\_date: new Date("05-nov-2020"),

company: "Infosys"

}

**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: [{$or: [{topic\_date: {$gt: new Date("30-sep-2020")}},{topic\_date: {$lt: new Date("1-nov-2020")}}]},

{$or: [{"taskinfo.due\_date": {$gt: new Date("30-sep-2020")}},{"taskinfo.due\_date": {$lt: new Date("1-nov-2020")}}]}

]}

}

])

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

db.comapnydrives.find({$or: [{drive\_date: {$gte: new Date("15-oct-2020")}},

{drive\_date: {$lte: new Date("31-0ct-2020")}}]})

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

db.comapnydrives.aggregate([

{

$lookup: {

from: "users",

localField: "userid",

foreignField: "userid",

as : "userinfo"

}

},

{

$project: {

\_id: 0,

"userinfo.name": 1,

company: 1,

drive\_date: 1,

"userinfo.email": 1,

"userinfo.userid": 1

}

}

])

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

db.codekata.aggregate([

{

$lookup: {

from: "users",

localField: "userid",

foreignField: "userid",

as: "userinfo"

}

},

{

$project:{

\_id: 0,

userid: 1,

problems: 1,

"userinfo.name": 1

}

}

])

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

db.users.aggregate([

{$lookup: {

from: "mentors",

localField: "mentorid",

foreignField: "mentorid",

as: "mentorInfo"

}},

{$group:{\_id:{mentorid:"$mentorInfo.mentorid",mentorname:"$mentorInfo.mentorname"},mentee\_count:{$sum:1}}},

{$match:{mentee\_count:{$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: "topics"

}},

{

$lookup: {

from: "tasks",

localField: "topicid",

foreignField: "topicid",

as: "tasks"

}

},

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

{$match: {$and: [{$or:[{"topics.topic\_date": {$gte: new Date("15-oct-2020")}},{"topics.topic\_date": {$lte: new Date("31-oct-2020")}}]},

{$or: [{"tasks.due\_date": {$gte: new Date("15-oct-2020")}},{"tasks.due\_date": {$lte: new Date("31-oct-2020")}}]}

]}},

{

$count: "No\_of\_students\_absent"

}

])

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_