Scenario 4

Patients are identified by an SSN, and their names, addresses, and ages must be recorded. Doctors are identified by an SSN. For each doctor, the name, specialty, and years of experience must be recorded. Each pharmaceutical company is identified by name; it has an address and one phone number. For each drug, the trade name and formula must be recorded. Each drug is sold by a given pharmaceutical company, and the trade name identifies a drug uniquely from among the products of that company. Each pharmacy has a name, address, and phone number. Each patient is checked up by some doctor. Every doctor has at least one patient. Each pharmacy sells several drugs and has a price for each. A drug could be sold at several pharmacies, and the price could vary from one pharmacy to another. Doctors prescribe drugs for patients. A doctor could prescribe one or more drugs for several patients, and a patient could obtain prescriptions from several doctors. Each prescription has a date and a quantity associated with it. Pharmaceutical companies have long-term contracts with pharmacies. A pharmaceutical company can contract with several pharmacies, and a pharmacy can contract with several pharmaceutical companies. For each contract, you have to store a start date, an end date, supervisor and the text of the contract.

Inserting Data in Collections:

Patient:

```
db.patient.insertOne({ssn:312,name:'Jagan',city:'Bangalore',age:24,doctor:['D101'],prescribe:['P1']})
db.patient.insertOne({ssn:123,name:'Vikas',city:'Mysore',age:19,doctor:['D102','D101'],prescribe:['P2','P4'
]})
db.patient.insertOne({ssn:231,name:'Mansoor',city:'Mysore',age:20,doctor:['D103'],prescribe:['P3']})
```

DOCTOR:

```
db.doctor.insertOne({ssn:'D101',name:'Suresh',expert:'eye',yoe:12})
db.doctor.insertOne({ssn:'D102',name:'Ramesh',expert:'heart',yoe:21})
db.doctor.insertOne({ssn:'D103',name:'Ganesh',expert:'eye',yoe:23})
```

PRECRIBE:

```
db.prescibe.insertOne({pid:'P1',patient:312,doctor:'D101',drug:['aspirin','dolo'],date:new
Date('2013-10-23'),Quantity:1})
db.prescibe.insertOne({pid:'P2',patient:123,doctor:'D102',drug:['aspirin','crocin'],date:new
Date('2013-1-3'),Quantity:4})
db.prescibe.insertOne({pid:'P3',patient:231,doctor:'D103',drug:['morphine','dolo'],date:new
Date('2012-11-12'),Quantity:2})
db.prescibe.insertOne({pid:'P4',patient:123,doctor:'D101',drug:['aspirin','paracetomol'],date:new
Date('2013-6-5'),Quantity:1})
```

PCOM:

```
db.pcom.insertOne({name:'biogen',city:'bangalore',phone:423521,drug:['aspirin','morphine']})
db.pcom.insertOne({name:'elilily',city:'bangalore',phone:324324,drug:['dolo']})
db.pcom.insertOne({name:'novopharm',city:'mysore',phone:3424234,drug:['crocin']})
```

DRUG:

```
db.drug.insertOne({name:'aspirin',formula:'sd7d8'})
db.drug.insertOne({name:'paracetomol',formula:'dsd8sd9'})
db.drug.insertOne({name:'morphine',formula:'sd9sd9s'})
db.drug.insertOne({name:'dolo',formula:'sd0sds9'})
db.drug.insertOne({name:'crocin',formula:'sd9ds9'})
```

PHARMA:

```
db.pharm.insertOne({name:'corner
shop',city:'bangalore',phone:4324324,drug:[{name:'aspirin',price:'10'},{name:'dolo',price:5},{name:'morp
hine',price:'8'}],contract:['C1','C2']})
db.pharm.insertOne({name:'highland
shop',city:'bangalore',phone:3424234,drug:[{name:'aspirin',price:'8'},{name:'morphine',price:'10'},{name:'
crocin',price:3}],contract:['C3','C4']})
```

CONTRACT:

```
db.contract.insertOne({id:'C1',pcomname:'biogen',stdate:'2001-12-3',endate:'2031-12-3',sup:'Sup1',text:'h
bdvjasbdhgcsagdbjhsdvasvdas'})
db.contract.insertOne({id:'C2',pcomname:'novopharm',stdate:'2001-12-3',endate:'2031-12-3',sup:'Sup3',te
xt:'dsadjhasvhdhsadvgsahdvhsad'})
db.contract.insertOne({id:'C3',pcomname:'elilily',stdate:'2001-12-3',endate:'2031-12-3',sup:'Sup2',text:'dsa
hjdhsabvhdashdbhjsak'})
db.contract.insertOne({id:'C4',pcomname:'biogen',stdate:'2001-12-3',endate:'2031-12-3',sup:'Sup1',text:'d
hjsdhsabdhvsauydjhsav'})
```

ASSISTANT:

```
db.assistant.insertMany([
{
        ssn: 'AS001',
        name: 'Priyanka',
        address: 'Sirsi',
        yoe:"10",
        wards:["general","icu","op-theatre"]
},
{
        ssn: 'AS002',
        name: 'Shashant',
        address: 'Agra',
        yoe:"10",
        wards:["icu","op-theatre"]
},
{
        ssn: 'AS003',
        name: 'Prajna',
        address: 'mysuru',
 yoe:"2",
 wards:["general"]
},
{
        ssn: 'AS004',
        name: 'Preetam',
        address: 'Sagar',
        yoe:"12",
        wards:["icu"]
},
{
        ssn: 'AS005',
        name: 'Manoj',
        address: 'Hubballi',
 yoe:"5",
 wards:["general","icu"]
},
{
        ssn: 'AS006',
        name: 'Ashwin',
        address: 'Chennai',
        yoe:"7",
```

```
 wards:[]
 }])
```

Queries:

1. **List the details of patients who are 20 years old and have been checked by eye-specialists.**

```
db.patient.aggregate([
 {$match: {age: 20}},
 {$lookup: {
        from: "doctor",
        localField: "doctor",
        foreignField: "ssn",
        as: "doctorDetails"
        }},
 {$unwind: "$doctorDetails"},
 {$match: {"doctorDetails.expert": "eye"}},
 {$project: {
        _id: 0,
        ssn: 1,
        name: 1,
        city: 1,
        age: 1,
        doctor: "$doctorDetails.name"
        }}
])
```

OUTPUT:

```
[
 {
        ssn: 231,
        name: 'Mansoor',
        city: 'Mysore',
        age: 20,
        doctor: 'Ganesh'
 }
]
```

2. **List the details of doctors who have given the prescription to more than 2 patients in year 2013.**

Extra insertions as per requirements:

```
db.prescribe.insertMany([
{pid: 'P5', patient: 312, doctor: 'D101', drug: ['aspirin'], date: new Date('2013-02-15'), quantity: 1},
{pid: 'P6', patient: 123, doctor: 'D101', drug: ['crocin'], date: new Date('2013-05-20'), quantity: 2},
{pid: 'P7', patient: 231, doctor: 'D101', drug: ['dolo'], date: new Date('2013-07-30'), quantity: 3},
{pid: 'P8', patient: 312, doctor: 'D102', drug: ['aspirin'], date: new Date('2013-03-10'), quantity: 1},
{pid: 'P9', patient: 123, doctor: 'D102', drug: ['paracetamol'], date: new Date('2013-09-15'),
quantity: 2},
])
```

Query:

```
db.prescribe.aggregate([
 {
        // Filter prescriptions to include only those from the year 2013
        $match: {
        date: {
        $gte: new ISODate("2013-01-01T00:00:00Z"),
        $lt: new ISODate("2014-01-01T00:00:00Z")
        }
        }
 },
 {
        // Group by doctor and patient to count unique patients
        $group: {
        _id: { doctor: "$doctor", patient: "$patient" }
        }
 },
 {
        // Group by doctor and count the number of unique patients
        $group: {
        _id: "$_id.doctor",
        uniquePatientCount: { $sum: 1 }
        }
```

```
  },
  {
        // Filter doctors who have prescribed to more than 2 unique patients
        $match: {
        uniquePatientCount: { $gt: 2 }
        }
  },
  {
        // Join with the doctor collection to get doctor details
        $lookup: {
        from: "doctor",
        localField: "_id",
        foreignField: "ssn",
        as: "doctorDetails"
        }
  },
  {
        // Unwind the doctorDetails array
        $unwind: "$doctorDetails"
  },
  {
        // Project the required fields
        $project: {
        _id: 0,
        ssn: "$doctorDetails.ssn",
        name: "$doctorDetails.name",
        specialty: "$doctorDetails.specialty",
        yoe: "$doctorDetails.yoe",
        uniquePatientCount: 1
        }
  }
])
```

OUTPUT:

[{ uniquePatientCount: 3, ssn: 'D101', name: 'Suresh', yoe: 23 }]

3. **Retrieve the details of the doctor with the maximum experience.**

```
db.doctor.aggregate([
 {
        $group: {
        _id: null,
        maxExperience: { $max: "$yoe" }
        }
 },
 {
        $unwind: "$maxExperience"
 },
 {
        $lookup: {
        from: "doctor",
        localField: "maxExperience",
        foreignField: "yoe",
        as: "doctorsWithMaxExperience"
        }
 },
 {
        $project: {
        _id: 0,
        doctorsWithMaxExperience: 1
        }
 }
])
```

OUTPUT:

```
[
 {
        doctorsWithMaxExperience: [
        {
        _id: ObjectId("6661849a34cd169e66cb795f"),
        ssn: 'D101',
        name: 'Suresh',
        expert: 'eye',
        yoe: 23
        },
        {
        _id: ObjectId("6661849c34cd169e66cb7961"),
        ssn: 'D103',
```

```
        name: 'Ganesh',
        expert: 'eye',
        yoe: 23
        }
        ]
  }
]
```

4. **List the details of pharmaceutical companies who supply drugs to more than 1 pharmacies in the same city where the company is located.**

```
db.pcom.aggregate([
  {$unwind:"$drug"},
  {
        $group:{
        _id:{name:"$name",city:"$city"},
        count:{$sum:1}
        }
  },
  {$match:{count:{$gte:2}}},
  {
        $project:{
        _id:0,
        name:"$_id.name",
        city:"$_id.city"
        }
  }
])
```

OUTPUT:

```
[ { name: 'biogen', city: 'bangalore' } ]
```

5. **List the details of healthcare assistants who serve in the operation theatre.**

```
db.assistant.find({
  wards: "op-theatre"
})
```

OUTPUT:

```
[
  {
        _id: ObjectId("666186cf34cd169e66cb7978"),
        ssn: 'AS001',
        name: 'Priyanka',
        address: 'Sirsi',
        yoe: '10',
        wards: [ 'general', 'icu', 'op-theatre' ]
  },
  {
        _id: ObjectId("666186cf34cd169e66cb7979"),
        ssn: 'AS002',
        name: 'Shashant',
        address: 'Agra',
        yoe: '10',
        wards: [ 'icu', 'op-theatre' ]
  }
]
```

6. **List the details of drugs supplied by all pharmaceutical companies.**

```
db.pharm.aggregate([
{ $unwind: "$drug" },
{ $group:
        { _id: "$drug.name", pharmacies: { $addToSet: "$name" }, count: { $sum: 1 } } },
{ $match: { count: { $gte: db.pharm.find().count() } } },
{ $project: { _id: 0, drug: "$_id", pharmacies: 1 } }
] )
```

OUTPUT:

```
[
  { pharmacies: [ 'highland shop', 'corner shop' ], drug: 'aspirin' },
  { pharmacies: [ 'highland shop', 'corner shop' ], drug: 'morphine' }
]
```