# Homework 4

## Question 1

```
$ brew services start mongodb-community
$ mongoimport --db hw4 --collection zipcodes --file zips.json
```

```
use hw4
show collections
db.zipcodes.find().limit(10)
```

```
hw4> show collections
zipcodes
```

```
hw4> db.zipcodes.find().limit(10)
[
  {
    _id: '01008',
    city: 'BLANDFORD',
    loc: [ -72.936114, 42.182949 ],
    pop: 1240,
    state: 'MA'
  },
  {
    _id: '01013',
    city: 'CHICOPEE',
    loc: [ -72.607962, 42.162046 ],
    pop: 23396,
    state: 'MA'
  },
  {
    _id: '01020',
    city: 'CHICOPEE',
    loc: [ -72.576142, 42.176443 ],
    pop: 31495,
    state: 'MA'
  },
  {
    _id: '01022',
    city: 'WESTOVER AFB'
```

## Question 2

```
db.zipcodes
    .find({
        pop: { $gt: 50000 },
        state: "CA",
```

```
      "loc.1": { $gt: 35 }
    })
    .limit(5)
    .sort({ pop: -1 })
```

```
[
  {
    _id: '94501',
    city: 'COAST GUARD ISLA',
    loc: [ -122.260516, 37.764783 ],
    pop: 76110,
    state: 'CA'
  },
  {
    _id: '94110',
    city: 'SAN FRANCISCO',
    loc: [ -122.415344, 37.750858 ],
    pop: 70770,
    state: 'CA'
  },
  {
    _id: '95351',
    city: 'MODESTO',
    loc: [ -121.006033, 37.625022 ],
    pop: 69275,
    state: 'CA'
  },
  {
    _id: '95076',
    city: 'LA SELVA BEACH',
    loc: [ -121.763437, 36.920515 ],
    pop: 68295,
    state: 'CA'
  },
  {
    _id: '94533',
    city: 'FAIRFIELD',
    loc: [ -122.03565, 38.267084 ],
    pop: 65455,
    state: 'CA'
  }
]
```

## Question 3

```
db.zipcodes
    .find({
        state: { $ne: "CA" },
        $or: [
            { "loc.0": { $lt: -120 } },
            { "loc.1": { $lt: 40 } }
        ]
    })
    .limit(5)
```

```
[
  {
    _id: '08002',
    city: 'CHERRY HILL',
    loc: [ -75.017538, 39.930808 ],
    pop: 21271,
    state: 'NJ'
  },
  {
    _id: '08003',
    city: 'CHERRY HILL',
    loc: [ -74.970568, 39.880453 ],
    pop: 29058,
    state: 'NJ'
  },
  {
    _id: '08004',
    city: 'WINSLOW',
    loc: [ -74.879368, 39.770909 ],
    pop: 14312,
    state: 'NJ'
  },
  {
    _id: '08005',
    city: 'BARNEGAT',
    loc: [ -74.246988, 39.755248 ],
    pop: 13036,
    state: 'NJ'
  },
  {
    _id: '08007',
    city: 'BARRINGTON',
    loc: [ -75.056361, 39.865062 ],
    pop: 5185,
    state: 'NJ'
  }
]
```

## Question 4

```
db.zipcodes.aggregate([
  {
    $group: {
      _id: "$city",
      population: { $sum: "$pop" }
    }
  },
  {
    $sort: { population: -1 }
  },
  {
    $limit: 5
  },
])
```

```
[
  { _id: 'CHICAGO', population: 2452177 },
  { _id: 'BROOKLYN', population: 2341387 },
  { _id: 'HOUSTON', population: 2123053 },
  { _id: 'LOS ANGELES', population: 2102295 },
  { _id: 'PHILADELPHIA', population: 1639862 }
]
```

## Question 5

```
db.zipcodes.aggregate([
  {
    $group: {
      _id: "$state",
      zipCodeCount: { $sum: 1 }
    }
  },
  {
    $match: {
      zipCodeCount: { $gt: 300, $lt: 500 }
    }
  },
  {
    $sort: { zipCodeCount: 1 }
  },
])
```

```
[
  { _id: 'MT', zipCodeCount: 314 },
  { _id: 'SC', zipCodeCount: 350 },
  { _id: 'MS', zipCodeCount: 363 },
  { _id: 'SD', zipCodeCount: 384 },
  { _id: 'OR', zipCodeCount: 384 },
  { _id: 'ND', zipCodeCount: 391 },
  { _id: 'ME', zipCodeCount: 410 },
  { _id: 'CO', zipCodeCount: 414 },
  { _id: 'MD', zipCodeCount: 420 },
  { _id: 'LA', zipCodeCount: 464 },
  { _id: 'MA', zipCodeCount: 474 },
  { _id: 'WA', zipCodeCount: 484 }
]
```

## Question 6

```
db.createCollection("customers", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["name", "zipcode", "avg_rating"],
      properties: {
        name: {
          bsonType: "string",
        },
        zipcode: {
          bsonType: "string",
        },
        avg_rating: {
          bsonType: "double",
          minimum: 0.0,
          maximum: 10.0,
        },
        last_order: {
          bsonType: "object",
          required: ["year"],
          properties: {
            year: {
```

```
        bsonType: "int",
      },
      tags: {
        bsonType: "array",
        items: {
          bsonType: "string"
        },
      }
    },
  }
  }
  }
})
```

```
...        },
...        tags: {
...          bsonType: "array",
...          items: {
...            bsonType: "string"
...          },
...        }
...      },
...    }
...    }
...    }
... })
{ ok: 1 }
```

## Question 7

```
db.customers.insertMany([
  {
    name: "Altun Hasanli",
    zipcode: "99503",
    avg_rating: 8.3
  },
  {
    name: "Andrei T.",
    zipcode: "90025",
    avg_rating: 3.5,
    last_order: {
      year: 2009
    }
  },
  {
    name: "Bela T.",
    zipcode: "33126",
    avg_rating: 4.9,
    last_order: {
      year: 2019,
      tags: ["art", "melancholy"]
    }
  },
  {
    name: "Nuri Bilge C.",
    zipcode: "90010",
    avg_rating: 6.5,
```

```
      last_order: {
        year: 3005
      }
    }
  }
])
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6648b828088e137340063bfe'),
    '1': ObjectId('6648b828088e137340063bff'),
    '2': ObjectId('6648b828088e137340063c00'),
    '3': ObjectId('6648b828088e137340063c01')
  }
}
```

```
db.customers.find()
```

```
[
  {
    _id: ObjectId('6648b828088e137340063bfe'),
    name: 'Altun Hasanli',
    zipcode: '99503',
    avg_rating: 8.3
  },
  {
    _id: ObjectId('6648b828088e137340063bff'),
    name: 'Andrei T.',
    zipcode: '90025',
    avg_rating: 3.5,
    last_order: { year: 2009 }
  },
  {
    _id: ObjectId('6648b828088e137340063c00'),
    name: 'Bela T.',
    zipcode: '33126',
    avg_rating: 4.9,
    last_order: { year: 2019, tags: [ 'art', 'melancholy' ] }
  },
  {
    _id: ObjectId('6648b828088e137340063c01'),
    name: 'Nuri Bilge C.',
    zipcode: '90010',
    avg_rating: 6.5,
    last_order: { year: 3005 }
  }
]
```

## Question 8

```
db.customers.deleteMany({
  "last_order.year": { $gt: 2024 }
})
```

```
db.customers.find()
```

```
hw4> db.customers.deleteMany({
...     "last_order.year": { $gt: 2024 }
... })
{ acknowledged: true, deletedCount: 1 }
hw4> db.customers.find()
[
  {
    _id: ObjectId('6648b828088e137340063bfe'),
    name: 'Altun Hasanli',
    zipcode: '99503',
    avg_rating: 8.3
  },
  {
    _id: ObjectId('6648b828088e137340063bff'),
    name: 'Andrei T.',
    zipcode: '90025',
    avg_rating: 3.5,
    last_order: { year: 2009 }
  },
  {
    _id: ObjectId('6648b828088e137340063c00'),
    name: 'Bela T.',
    zipcode: '33126',
    avg_rating: 4.9,
    last_order: { year: 2019, tags: [ 'art', 'meland
  }
]
```

## Question 9

```
db.customers.updateOne(
  { name: "Altun Hasanli" },
  { $set: { avg_rating: 15 } }
)
```

```
hw4> db.customers.updateOne(
...    { name: "Altun Hasanli" },
...    { $set: { avg_rating: 15 } }
... )
Uncaught:
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('6648b828088e137340063bfe'),
  details: {
    operatorName: '$jsonSchema',
    schemaRulesNotSatisfied: [
      {
        operatorName: 'properties',
        propertiesNotSatisfied: [
          {
            propertyName: 'avg_rating',
            details: [
              {
                operatorName: 'maximum',
                specifiedAs: { maximum: 10 },
                reason: 'comparison failed',
                consideredValue: 15
              },
              {
                operatorName: 'bsonType',
                specifiedAs: { bsonType: 'double' },
                reason: 'type did not match',
                consideredValue: 15,
                consideredType: 'int'
              }
            ]
          }
        ]
      }
    ]
  }
}
```

## Question 10

```
db.customers.aggregate([
  {
    $lookup: {
      from: "zipcodes",
      localField: "zipcode",
      foreignField: "_id",
      as: "zipcode_info"
    }
  },
  {
    $unwind: "$zipcode_info"
  },
  {
```

```
      $project: {
        name: 1,
        city: "$zipcode_info.city",
        state: "$zipcode_info.state",
        zipcode: "$zipcode_info._id"
      }
    },
    {
      $sort: { zipcode: 1 }
    }
  ])
```

```
hw4> db.customers.aggregate([ { $lookup: { from: "zipcodes", localField: "zipcode", foreignField: "_id", as: "zipcode_info" } }, { $unwind: "$zipcode_info" }, { $project: { nam
e: 1, city: "$zipcode_info.city", state: "$zipcode_info.state", zipcode: "$zipcode_info._id" } }, { $sort: { zipcode: 1 } }] )
[
  {
    _id: ObjectId('6648b828088e137340063c00'),
    name: 'Bela T.',
    city: 'MIAMI',
    state: 'FL',
    zipcode: '33126'
  },
  {
    _id: ObjectId('6648b828088e137340063bff'),
    name: 'Andrei T.',
    city: 'LOS ANGELES',
    state: 'CA',
    zipcode: '90025'
  },
  {
    _id: ObjectId('6648b828088e137340063bfe'),
    name: 'Altun Hasanli',
    city: 'ANCHORAGE',
    state: 'AK',
    zipcode: '99503'
  }
]
```

## Question 11

```
$ mongoexport --db hw4 --collection customers --out customers.json
```

```
→  homework4 mongoexport --db hw4 --collection customers --out customers.json
2024-05-18T17:56:23.779+0300    connected to: mongodb://localhost/
2024-05-18T17:56:23.788+0300    exported 3 records
```