**MongoDB**

Name: Narayana N Gourav

|  |
| --- |
| Consider the following restaurant database with the following attributes -  Name, address –(building, street, area, pincode),id, cuisine, nearby landmarks, online delivery- yes/no, famous for(name of the dish).  Create 10 collections with data relevant to the following questions. Write and execute MongoDB queries:   1. List the name and address of all restaurants in Bangalore with Italian cuisine 2. List the name, address and nearby landmarks of all restaurants in Bangalore where north Indian thali is available. 3. List the name and address of restaurants and also the dish the restaurant is famous for, in Bangalore. 4. List the name and address of restaurants and also tdish the restaurant is famous for, in Bangalore where online delivery is available. |

|  |
| --- |
| use restaurant;  db.createCollection('res'); |
|  |
| db.res.insert([  {  name: "McDonalds",  address: "12 BEL Road 560095",  cuisine: "Fast food",  landmark: "MSR Hospital",  delivery: "Yes",  famousfor: "Cheese burger"  },  {  name: "Burger King",  address: "8 BEL Road, 560095",  cuisine: "Fast food",  landmark: "MSR Hospital",  delivery: "Yes",  famousfor: "Cheese burger"  },  {  name: "Naturals",  address: "34 St Marks Road, 560023",  cuisine: "Fast food",  landmark: "St Marks Hotel",  delivery: "Yes",  famousfor: "Ice Cream"  },  {  name: "Little Italy",  address: "21 Orchard Road, Sadashivnagar, 560017",  cuisine: "Italian",  landmark: "Bashim Circle",  delivery: "No",  famousfor: "Pizza"  },  {  name: "Nandhana",  address: "10 Street, Jayanagar, 560087",  cuisine: "South Indian",  landmark: "post office",  delivery: "Yes",  famousfor: "South Indian Thali"  },  {  name: "Chalukya",  address: "10 Street, Kumara Park, 560044",  cuisine: "North Indian",  landmark: "Hotel Samrat",  delivery: "Yes",  famousfor: "North Indian Thali"  },  {  name: "1522",  address: "31 Street, Rajajinagar, 560017",  cuisine: "North Indian",  landmark: "Post office",  delivery: "Yes",  famousfor: "Dragon Chicken"  },  {  name: "Stories",  address: "12 Street, Rajajinagar, 560021",  cuisine: "fast food",  landmark: "Post office",  delivery: "yes",  famousfor: "Cheese burger"  },  {  name: "The Bangalore Cafe",  address: "13 Street, Jayanagar, 560021",  cuisine: "Continental",  landmark: "Post office",  delivery: "Yes",  famousfor: "Pasta"  },  {  name: "Pizza Hut",  address: "12 Street, RT Nagar, 560021",  cuisine: "Italian",  landmark: "Post office",  delivery: "yes",  famousfor: "Pizza"  },  ]); |
|  |
|  |
|  |
| db.res.find({ cuisine: "Italian" }, { name: 1, address: 1, \_id: 0 }).pretty(); |
|  |
| db.res.find({ famousfor: "North Indian Thali" }, { name: 1, address: 1, landmarks: 1, \_id: 0 }).pretty(); |
|  |
| db.res.find({},{ name: 1, address: 1, famousfor: 1, \_id:0 }).pretty(); |
|  |
|  |
| db.res.find({delivery: "Yes"},{ name: 1, address: 1, famousfor: 1, \_id:0 }).pretty(); |
|  |

|  |
| --- |
| Consider the following Tourist places table with the following attributes -  Place, address – (state), id, tourist attractions,best time of the year to visit,modes of transport(include nearest airport, railway station etc), accommodation, food - what not to miss for sure.  Create 10 collections with data relevant to the following questions. Write and execute MongoDB queries:   1. List all the tourist places of Karnataka 2. List the tourist attractions of Kerala. Exclude accommodation and food 3. List the places sorted state wise. |

|  |
| --- |
| use tourist;  db.createCollection("tourist"); |
|  |
| db.tourist.insert([  {  place: "Bangalore",  address: "Karnataka",  id: 1,  tour\_att: "Vidhan Soudha",  time: "January",  mode: ["Train", "Air", "Road"],  accommodation: "Hotel",  food: "Chicken"  },  {  place: "Kochi",  address: "Kerala",  id: 2,  tour\_att: "Backwaters",  time: "Febuary",  mode: ["Sea", "Air", "Road", "Train"],  accommodation: "Hotel",  food: "Fish Fry"  },  {  place: "Agra",  address: "Delhi",  id: 3,  tour\_att: "Taj Mahal",  time: "March",  mode: ["Train", "Air", "Road"],  accommodation: "Hotel",  food: "Petha"  },  {  place: "Gangtok",  address: "Sikkim",  id: 4,  tour\_att: "Lake Tsomgo",  mode: ["Road"],  accommodation: "Resort",  food: "Momo"  },  {  place: "Srinagar",  address: "Jammu And Kashmir",  id: 5,  tour\_att: "Dal Lake",  mode: ["Air", "Road"],  accommodation: "Hotel",  food: "Dried Fruits"  },  {  place: "Shimla",  address: "Himachal Pradesh",  id: 6,  tour\_att: "Viceroy Lodge",  mode: ["Road"],  accommodation: "Hotel",  food: "Madra"  },  {  place: "Kodaikanal",  address: "Tamil Nadu",  id: 7,  tour\_att: "Kodaikanal Lake",  mode: ["Road"],  accommodation: "Resort",  food: "Moroccan Spice Tea"  },  {  place: "Corbet",  address: "Uttarakhand",  id: 8,  tour\_att: "Jim Corbett National Park",  mode: ["Road", "Train"],  accommodation: "Resort",  food: "Rice"  },  {  place: "Mumbai",  address: "Maharastra",  id: 9,  tour\_att: "Queen's Necklace",  mode: ["Air", "Ship", "Road", "Train"],  accommodation: "Hotel",  food: "Vada Pav"  },  {  place: "Hyderabad",  address: "Telangana",  id: 10,  tour\_att: "Salar Jung Museum",  mode: ["Air", "Road", "Train"],  accommodation: "Hotel",  food: "Biryani"  }  ]); |
|  |
|  |
|  |
|  |
| db.tourist.find({ address: "Karnataka" }, { tour\_att: 1, id: 1, \_id: 0 }).pretty(); |
|  |
| db.tourist.find({ address: "Kerala" }, { accommodation: 0, food: 0 }).pretty(); |
|  |
| db.tourist.find({}, { place: 1, \_id: 0 }).sort({ address: 1 }).pretty(); |
|  |

|  |
| --- |
| Consider the following Movie table with the following attributes -  Actor\_name,Actor\_id, Actor\_birthdate, Dirctor\_name,Director\_id, Director\_birthdate, film\_title, year of production ,type (thriller, comedy, etc.)  Create 10 collections with data relevant to the following questions. Write and execute MongoDB queries:   1. List all the movies acted by John in the year 2018 2. List only the actors names and type of the movie directed by Ram 3. List all the movies acted by John and Elly in the year 2012. 4. List only the name and type of the movie where Ram has acted, sorted by movie names. |

|  |
| --- |
| use movies;  db.createCollection("movie"); |
|  |
| db.movie.insert([  {  actor\_name: ["Ram", "John"],  actor\_id: 13,  actor\_bdate: "2/3/1997",  director\_name: "Williams",  director\_id: 101,  director\_bdate: "12/9/1987",  film: "Battleship",  year: 2015,  type: ["Thriller", "Action"]  },  {  actor\_name: ["John"],  actor\_id: 11,  actor\_bdate: "1/2/1998",  director\_name: "Ram",  director\_id: 100,  director\_bdate: "2/3/1997",  film: "John Wick",  year: 2012,  type: ["Action"]  },  {  actor\_name: ["Elly", "John"],  actor\_id: 12,  actor\_bdate: "4/12/1998",  director\_name: "Ram",  director\_id: 100,  director\_bdate: "2/3/1997",  film: "Aquaman",  year: 2012,  type: ["Action", "Sci-fi"]  },  {  actor\_name: "Ram",  actor\_id: 13,  actor\_bdate: "2/3/1997",  director\_name: "Thomas",  director\_id: 103,  director\_bdate: "12/3/1999",  film: "xxx",  year: 2018,  type: ["Action"]  },  {  actor\_name: "John",  actor\_id: 11,  actor\_bdate: "1/2/1998",  director\_name: "Ram",  director\_id: 100,  director\_bdate: "2/3/1997",  film: "Mr. Bean",  year: 2018,  type: ["Comedy"]  },  ]); |
|  |
|  |
| db.movie.find({ actor\_name: "John", year: 2018 }, { film: 1, \_id: 0 }).pretty(); |
|  |
| db.movie.find({ director\_name: "Ram" }, { actor\_name: 1, type: 1, \_id: 0 }).pretty(); |
|  |
| db.movie.find({ $and: [{ actor\_name: "John" }, { actor\_name: "Elly" }, { year: 2012 }] }, { film: 1,\_id: 0 }).pretty(); |
|  |
| db.movie.find({ actor\_name: "Ram" }, { film: 1, type: 1, \_id: 0 }).sort({ film: 1 }).pretty(); |
|  |