

Mongodb-2 Design and Develop MongoDB Queries using CRUD operations: Create Employee collection by considering following Fields: i. Name: Embedded Doc (FName, LName) ii. Company Name: String iii. Salary: Number iv. Designation: String v. Age: Number vi. Expertise: Array vii. DOB: String or Date viii. Email id: String ix. Contact: String x. Address: Array of Embedded Doc (PAddr, LAddr) Insert at least 5 documents in collection by considering above attribute and execute following queries:

1. Final name of Employee where age is less than 30 and salary more than 50000.
2. Creates a new document if no document in the employee collection contains {Designation: "Tester", Company_name: "TCS", Age: 25}
3. Selects all documents in the collection where the field age has a value less than 30 or the value of the salary field is greater than 40000.
4. Find documents where Designation is not equal to "Developer".
5. Find _id, Designation, Address and Name from all documents where Company_name is "Infosys".
6. Display only FName and LName of all Employees

```
use companyDB
db.employee.insertMany([
  {
    Name: { FName: "Amit", LName: "Sharma" },
    Company_name: "Infosys",
    Salary: 45000,
    Designation: "Programmer",
    Age: 28,
    Expertise: ["MongoDB", "NodeJS", "Express"],
    DOB: ISODate("1997-05-12"),
    Email: "amit.sharma@infosys.com",
    Contact: "9876543210",
    Address: [
      { PAddr: "Pune, Maharashtra" },
      { LAddr: "Bangalore, Karnataka" }
    ]
  },
  {
    Name: { FName: "Sneha", LName: "Patil" },
    Company_name: "TCS",
    Salary: 32000,
    Designation: "Tester",
    Age: 25,
    Expertise: ["Java", "Selenium", "MySQL"],
    DOB: ISODate("1999-04-18"),
    Email: "sneha.patil@tcs.com",
    Contact: "9123456789",
    Address: [
      { PAddr: "Mumbai, Maharashtra" },
      { LAddr: "Chennai, Tamil Nadu" }
    ]
  },
  {
```

```

Name: { FName: "Rahul", LName: "Deshmukh" },
Company_name: "TCS",
Salary: 40000,
Designation: "Programmer",
Age: 30,
Expertise: ["Python", "MongoDB", "MySQL"],
DOB: ISODate("1995-01-15"),
Email: "rahul.deshmukh@tcs.com",
Contact: "9988776655",
Address: [
  { PAddr: "Nagpur, Maharashtra" },
  { LAddr: "Hyderabad, Telangana" }
]
},
{
  Name: { FName: "Priya", LName: "Nair" },
  Company_name: "Wipro",
  Salary: 28000,
  Designation: "Designer",
  Age: 26,
  Expertise: ["UI/UX", "Figma", "Photoshop"],
  DOB: ISODate("1998-07-20"),
  Email: "priya.nair@wipro.com",
  Contact: "9090909090",
  Address: [
    { PAddr: "Kochi, Kerala" },
    { LAddr: "Bangalore, Karnataka" }
  ]
},
{
  Name: { FName: "Vikas", LName: "Joshi" },
  Company_name: "Infosys",
  Salary: 60000,
  Designation: "Team Lead",
  Age: 35,
  Expertise: ["MongoDB", "MySQL", "Cassandra"],
  DOB: ISODate("1989-09-10"),
  Email: "vikas.joshi@infosys.com",
  Contact: "9876001234",
  Address: [
    { PAddr: "Pune, Maharashtra" },
    { LAddr: "Pune, Maharashtra" }
  ]
}
]);

```

Step 7: Test the Queries

Query 1 - Final Name of Employees where Age < 30 and Salary > 50000

```
db.employee.find(  
  { Age: { $lt: 30 }, Salary: { $gt: 50000 } },  
  { "Name.FName": 1, "Name.LName": 1, _id: 0 } );
```

Query 2 - Insert new document if not exists

If no document exists with {Designation: "Tester", Company_name: "TCS", Age: 25}, create one.

```
db.employee.updateOne(  
  { Designation: "Tester", Company_name: "TCS", Age: 25 },  
  {  
    $setOnInsert: {  
      Name: { FName: "Kiran", LName: "Verma" },  
      Salary: 30000,  
      Expertise: ["Java", "Selenium"],  
      DOB: ISODate("1999-08-10"),  
      Email: "kiran.verma@tcs.com",  
      Contact: "9012345678",  
      Address: [  
        { PAddr: "Delhi" },  
        { LAddr: "Gurgaon" }  
      ]  
    }  
  },  
  { upsert: true }  
);
```

```
db.employee.find({  
  Designation: "Tester",  
  Company_name: "TCS",  
  Age: 25  
}).pretty();
```

Query 3 - Select all where Age < 30 OR Salary > 40000

```
db.employee.find(  
  { $or: [ { Age: { $lt: 30 } }, { Salary: { $gt: 40000 } }  
] } );
```

Query 4 - Find documents where Designation ≠ "Developer"

```
db.employee.find({ Designation: { $ne: "Developer" } });
```

Query 5 - Find _id, Designation, Address, Name for employees where Company_name = "Infosys"

```
db.employee.find(  
  { Company_name: "Infosys" },  
  { _id: 1, Designation: 1, Address: 1, Name: 1 } );
```

Query 6-Display only FName and LName of all Employees

```
db.employee.find(  
    {},  
    { "Name.FName": 1, "Name.LName": 1, _id: 0 } );
```