The "Bengaluru Brigade" Service Marketplace

Check out the github repo for files and outputs :

https://github.com/LokeshBhaskarNR/Assignment_9---Bengaluru-Brigade-MarketPlace.git

Task 1: Schema Design & Data Creation

Schema Design for the application →

Database: `Bengaluru_Brigade`

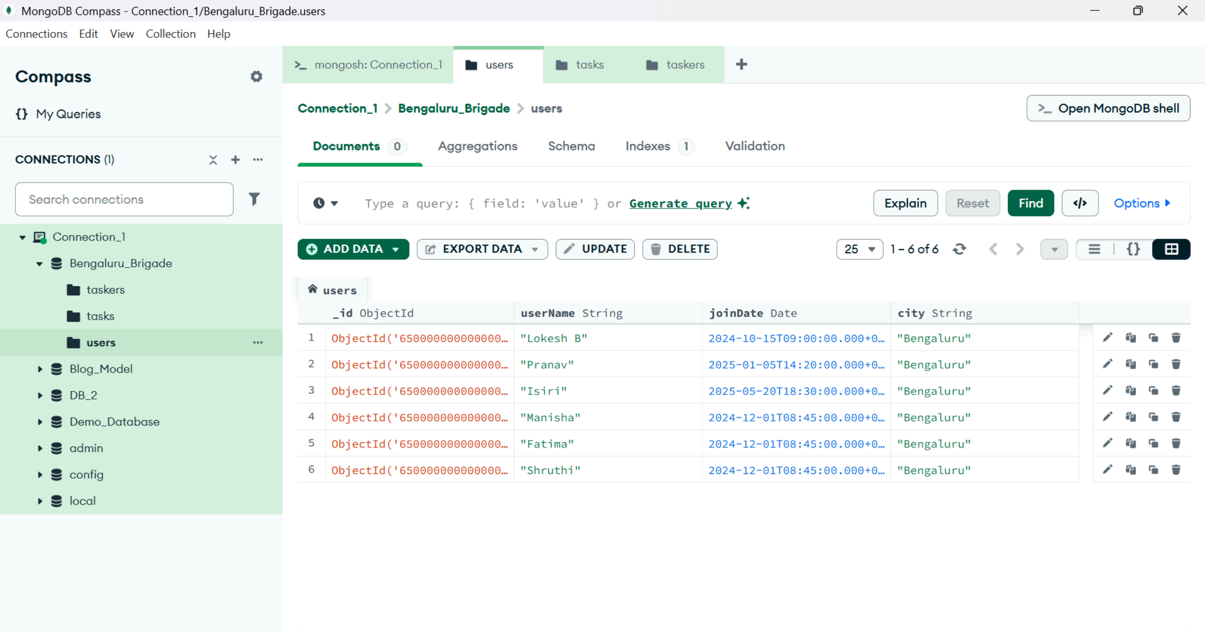
Collections & fields:

- **users**
  - `_id` (ObjectId)
  - `userName` (string)
  - `joinDate` (Date)
  - `city` (string) — e.g., `"Bengaluru"`
- **taskers**
  - `_id` (ObjectId)
  - `taskerName` (string)
  - `skills` (array of strings)
  - `area` (string) — e.g., `"Jayanagar"`
  - `rating` (number, 1–5) — overall rating
  - `hourlyRate` (number)
- **tasks**
  - `_id` (ObjectId)
  - `title` (string)
  - `description` (string)
  - `postedBy` (ObjectId) — reference to `users._id`
  - `assignedTo` (ObjectId or `null`) — reference to `taskers._id`
  - `category` (string)

    - `status` (string: `"Open"|"Assigned"|"Completed"`)
    - `cost` (number)
    - `completionDate` (Date) — added when completed

Creation And Insertion Of “Users” Collection :

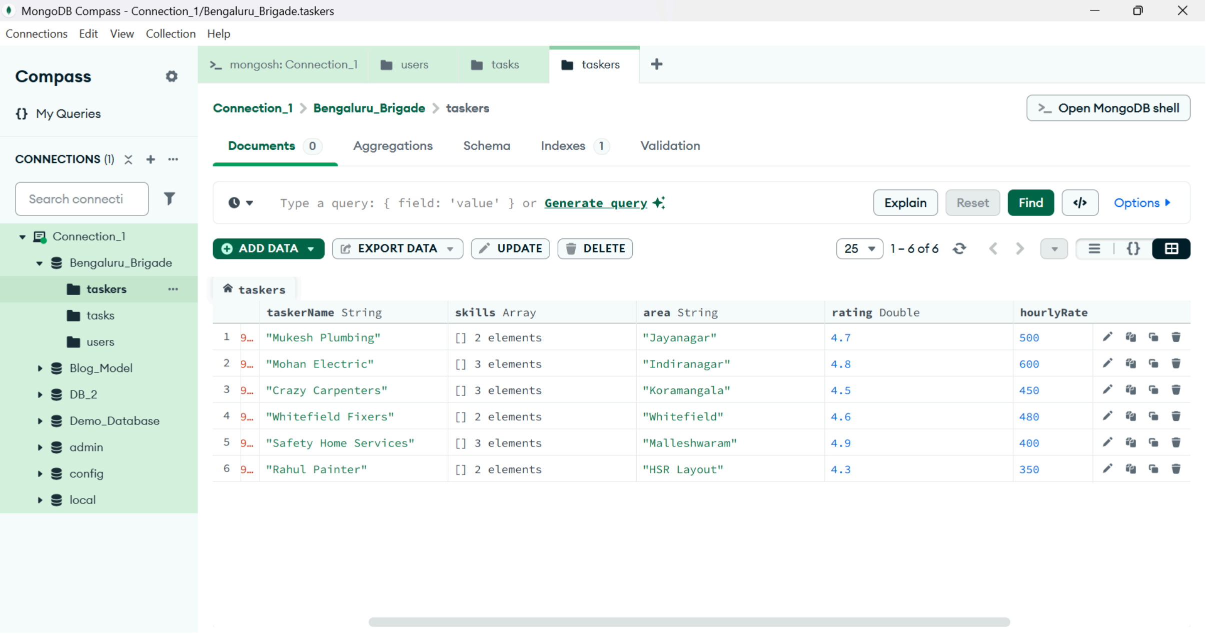
```
db.users.insertMany([
 {
   _id: ObjectId("650000000000000000000001"),
   userName: "Lokesh B",
   joinDate: ISODate("2024-10-15T09:00:00Z"),
   city: "Bengaluru"
 },
 {
   _id: ObjectId("650000000000000000000002"),
   userName: "Pranav",
   joinDate: ISODate("2025-01-05T14:20:00Z"),
   city: "Bengaluru"
 },
 {
   _id: ObjectId("650000000000000000000003"),
   userName: "Isiri",
   joinDate: ISODate("2025-05-20T18:30:00Z"),
   city: "Bengaluru"
 },
 {
   _id: ObjectId("650000000000000000000004"),
   userName: "Manisha",
   joinDate: ISODate("2024-12-01T08:45:00Z"),
   city: "Bengaluru"
 },
 {
  _id: ObjectId("650000000000000000000005"),
  userName: "Fatima",
  joinDate: ISODate("2024-12-01T08:45:00Z"),
  city: "Bengaluru"
 },
 {
  _id: ObjectId("650000000000000000000006"),
  userName: "Shruthi",
  joinDate: ISODate("2024-12-01T08:45:00Z"),
  city: "Bengaluru"
 }
]);
```



	_id ObjectId	userName String	joinDate Date	city String
1	ObjectId('650000000000000…	"Lokesh B"	2024-10-15T09:00:00.000+0…	"Bengaluru"
2	ObjectId('650000000000000…	"Pranav"	2025-01-05T14:20:00.000+0…	"Bengaluru"
3	ObjectId('650000000000000…	"Isiri"	2025-05-20T18:30:00.000+0…	"Bengaluru"
4	ObjectId('650000000000000…	"Manisha"	2024-12-01T08:45:00.000+0…	"Bengaluru"
5	ObjectId('650000000000000…	"Fatima"	2024-12-01T08:45:00.000+0…	"Bengaluru"
6	ObjectId('650000000000000…	"Shruthi"	2024-12-01T08:45:00.000+0…	"Bengaluru"

Creation And Insertion Of “Taskers” Collection :

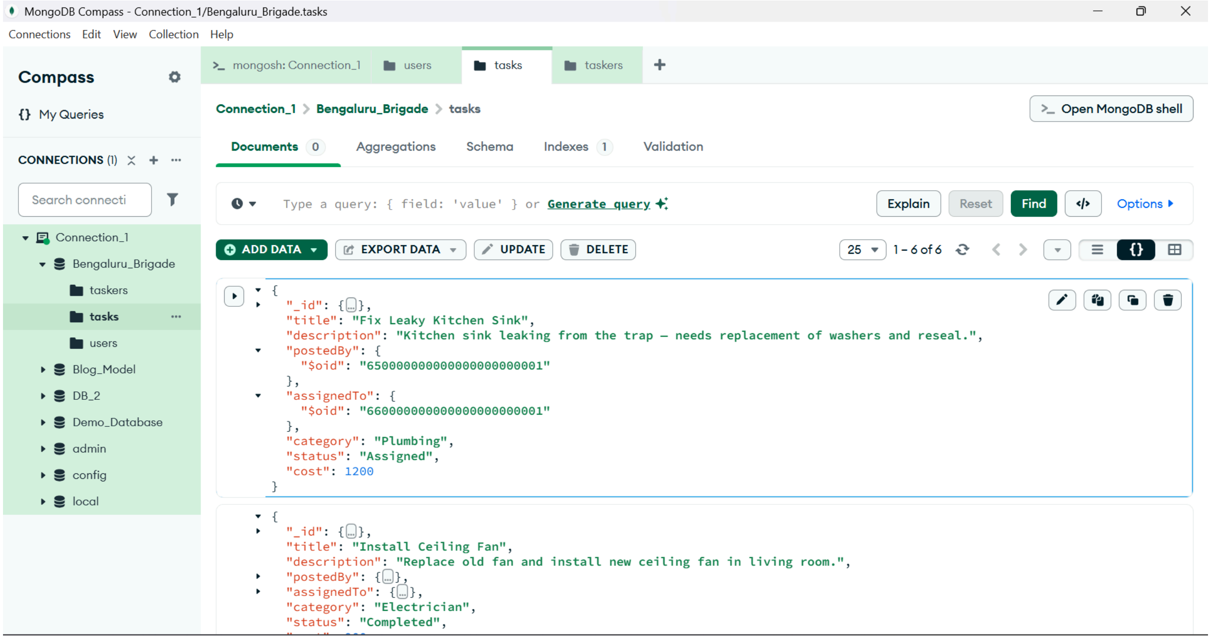
```
db.taskers.insertMany([
 {
  taskerName: "Suresh Plumbing Co.",
  skills: ["Plumbing", "Tiling"],
  area: "Jayanagar",
  rating: 4.7,
  hourlyRate: 500
},
{
  taskerName: "Manish Electric",
  skills: ["Electrician", "Home Wiring", "Appliance Repair"],
  area: "Indiranagar",
  rating: 4.8,
  hourlyRate: 600
},
{
  taskerName: "Divya Handyman",
  skills: ["Painting", "Carpentry", "Plumbing"],
  area: "Koramangala",
  rating: 4.5,
  hourlyRate: 450
},
{
  taskerName: "Whitefield Fixers",
  skills: ["Plumbing", "Electrician"],
  area: "Whitefield",
  rating: 4.6,
  hourlyRate: 480
},
{
    taskerName: "Anita Home Services",
    skills: ["Cleaning", "Cooking", "Laundry"],
    area: "Malleshwaram",
    rating: 4.9,
    hourlyRate: 400
  },
  {
    taskerName: "Rahul Painter",
    skills: ["Painting", "Wall Repair"],
    area: "HSR Layout",
    rating: 4.3,
    hourlyRate: 350
  }
]);
```



		taskerName String	skills Array	area String	rating Double	hourlyRate
1	9…	"Mukesh Plumbing"	[] 2 elements	"Jayanagar"	4.7	500
2	9…	"Mohan Electric"	[] 3 elements	"Indiranagar"	4.8	600
3	9…	"Crazy Carpenters"	[] 3 elements	"Koramangala"	4.5	450
4	9…	"Whitefield Fixers"	[] 2 elements	"Whitefield"	4.6	480
5	9…	"Safety Home Services"	[] 3 elements	"Malleshwaram"	4.9	400
6	9…	"Rahul Painter"	[] 2 elements	"HSR Layout"	4.3	350

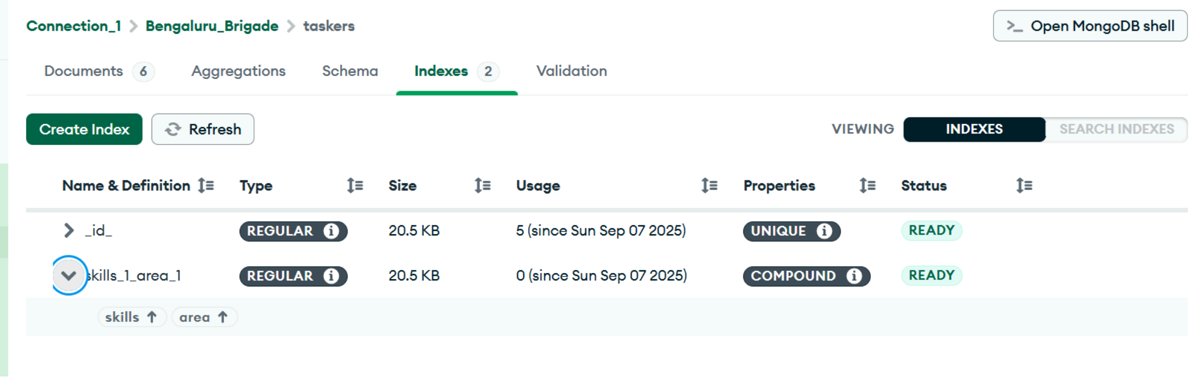
Creation And Insertion Of “Tasks” Collection :

```
db.tasks.insertMany([
  {
    _id: ObjectId("670000000000000000000001"),
    title: "Fix Leaky Kitchen Sink",
    description: "Kitchen sink leaking from the trap — needs replacement of washers and reseal.",
    postedBy: ObjectId("650000000000000000000001"), // Asha
    assignedTo: ObjectId("660000000000000000000001"), // Suresh
    category: "Plumbing",
    status: "Assigned",
    cost: 1200
  },
  {
    _id: ObjectId("670000000000000000000002"),
    title: "Install Ceiling Fan",
    description: "Replace old fan and install new ceiling fan in living room.",
    postedBy: ObjectId("650000000000000000000002"), // Vikram
    assignedTo: ObjectId("660000000000000000000002"), // Manish
    category: "Electrician",
    status: "Completed",
    cost: 900,
    completionDate: ISODate("2025-08-20T10:30:00Z")
  },
  {
    _id: ObjectId("670000000000000000000003"),
    title: "Bathroom Tile Repair",
    description: "Cracked tile near shower—replace two tiles and re-grout.",
    postedBy: ObjectId("650000000000000000000003"), // Neha
    assignedTo: null,
    category: "Tiling",
    status: "Open",
    cost: 600
  },
  {
    _id: ObjectId("670000000000000000000004"),
    title: "Wardrobe Door Fix",
    description: "Loose hinges on wardrobe doors; needs re-screwing and hinge replacement.",
    postedBy: ObjectId("650000000000000000000004"), // Ravi
    assignedTo: ObjectId("660000000000000000000003"), // Divya
    category: "Carpentry",
    status: "Completed",
    cost: 1500,
    completionDate: ISODate("2025-07-15T16:00:00Z")
  },
  {
    _id: ObjectId("670000000000000000000005"),
    title: "Full House Cleaning",
    description: "Deep cleaning of 2BHK apartment after housewarming ceremony.",
    postedBy: ObjectId("650000000000000000000005"), // Fatima
    assignedTo: ObjectId("660000000000000000000005"), // Anita Home Services
    category: "Cleaning",
    status: "Assigned",
    cost: 2000
  },
  {
    _id: ObjectId("670000000000000000000006"),
    title: "Living Room Painting",
    description: "Repaint living room with pastel colors before festival.",
    postedBy: ObjectId("650000000000000000000006"), // Shruthi
    assignedTo: ObjectId("660000000000000000000006"), // Rahul Painter

    category: "Painting",

    status: "Open",

    cost: 2500

  }

]);
```

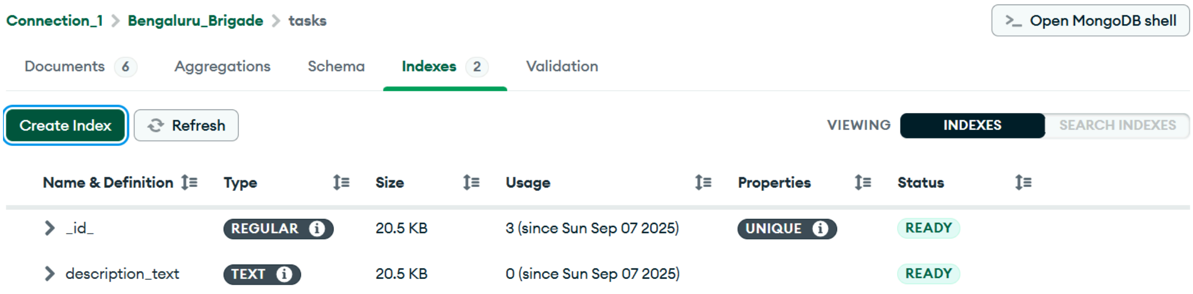


```
{
  "_id": {…},
  "title": "Fix Leaky Kitchen Sink",
  "description": "Kitchen sink leaking from the trap – needs replacement of washers and reseal.",
  "postedBy": {
    "$oid": "650000000000000000000001"
  },
  "assignedTo": {
    "$oid": "660000000000000000000001"
  },
  "category": "Plumbing",
  "status": "Assigned",
  "cost": 1200
}
{
  "_id": {…},
  "title": "Install Ceiling Fan",
  "description": "Replace old fan and install new ceiling fan in living room.",
  "postedBy": {…},
  "assignedTo": {…},
  "category": "Electrician",
  "status": "Completed",
```

Task 2: Indexing for Performance



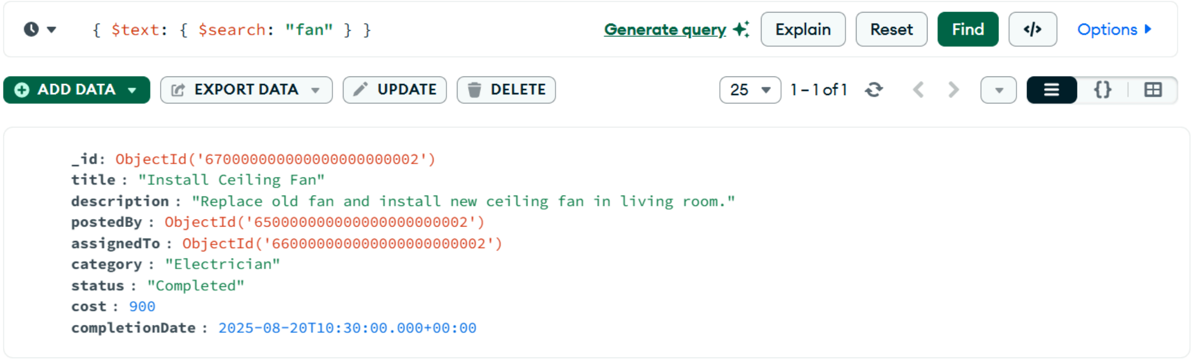
Name & Definition	Type	Size	Usage	Properties	Status
_id_	REGULAR	20.5 KB	5 (since Sun Sep 07 2025)	UNIQUE	READY
skills_1_area_1 skills ↑ area ↑	REGULAR	20.5 KB	0 (since Sun Sep 07 2025)	COMPOUND	READY



Name & Definition	Type	Size	Usage	Properties	Status
_id_	REGULAR	20.5 KB	3 (since Sun Sep 07 2025)	UNIQUE	READY
description_text	TEXT	20.5 KB	0 (since Sun Sep 07 2025)		READY

Querying using the keyword “fan” in tasks collection →

Shows related keywords (“Install Fan” , “Ceiling Fan”)

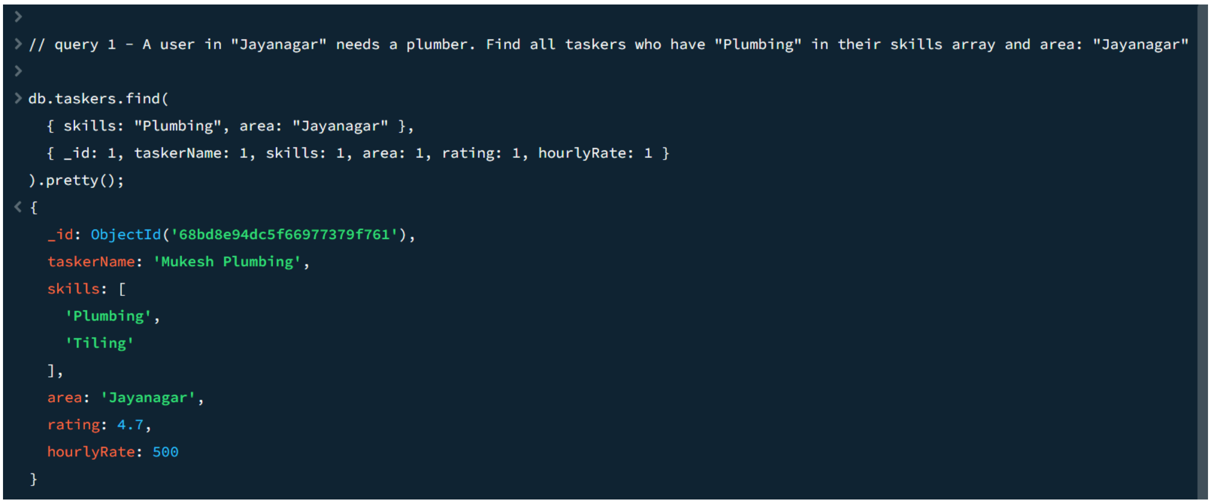


```
{ $text: { $search: "fan" } }
```

```
_id: ObjectId('670000000000000000000002')
title : "Install Ceiling Fan"
description : "Replace old fan and install new ceiling fan in living room."
postedBy : ObjectId('650000000000000000000002')
assignedTo : ObjectId('660000000000000000000002')
category : "Electrician"
status : "Completed"
cost : 900
completionDate : 2025-08-20T10:30:00.000+00:00
```

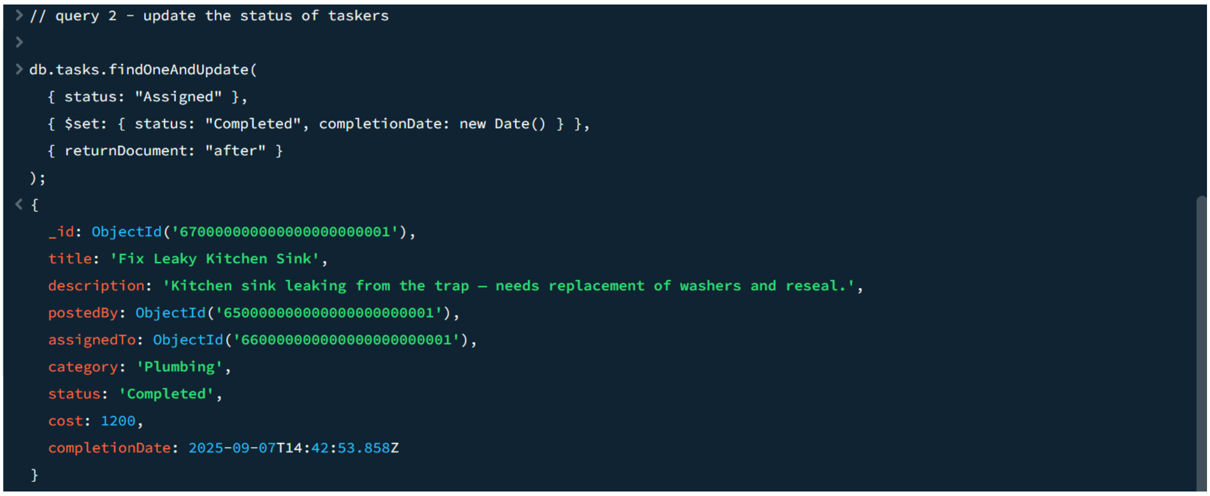
Task 3: Practical CRUD Queries

READ: A user in "Jayanagar" needs a plumber. Write a query to find all available taskers who have "Plumbing" in their skills array and work in the "Jayanagar" area.



```
>
> // query 1 - A user in "Jayanagar" needs a plumber. Find all taskers who have "Plumbing" in their skills array and area: "Jayanagar"
>
> db.taskers.find(
    { skills: "Plumbing", area: "Jayanagar" },
    { _id: 1, taskerName: 1, skills: 1, area: 1, rating: 1, hourlyRate: 1 }
  ).pretty();
< {
    _id: ObjectId('68bd8e94dc5f66977379f761'),
    taskerName: 'Mukesh Plumbing',
    skills: [
      'Plumbing',
      'Tiling'
    ],
    area: 'Jayanagar',
    rating: 4.7,
    hourlyRate: 500
  }
```

UPDATE: A tasker has just finished a job. Find a task with the status "Assigned" and update it. Use $set to change the status to "Completed" and add a new field called completionDate with the current date.



```
> // query 2 - update the status of taskers
>
> db.tasks.findOneAndUpdate(
    { status: "Assigned" },
    { $set: { status: "Completed", completionDate: new Date() } },
    { returnDocument: "after" }
  );
< {
    _id: ObjectId('670000000000000000000001'),
    title: 'Fix Leaky Kitchen Sink',
    description: 'Kitchen sink leaking from the trap – needs replacement of washers and reseal.',
    postedBy: ObjectId('650000000000000000000001'),
    assignedTo: ObjectId('660000000000000000000001'),
    category: 'Plumbing',
    status: 'Completed',
    cost: 1200,
    completionDate: 2025-09-07T14:42:53.858Z
  }
```

Task 4: The Aggregation Challenge

Business Question: "We want to identify our top-performing taskers. Who are our top 3 highest-rated taskers who have completed at least one task? For each of them, show their name, their average rating, and calculate their total earnings from all completed tasks."

Add stages for aggregation :

- `$lookup` (join with `tasks`)

  ```
  {
    "from": "tasks",
    "localField": "_id",
    "foreignField": "assignedTo",
    "as": "joinedTasks"
  }
  ```

- `$unwind`

  ```
  {
    "path": "$joinedTasks"
  }
  ```

- `$match` (status = Completed)

  ```
  {
    "joinedTasks.status": "Completed"
  }
  ```

- `$group`

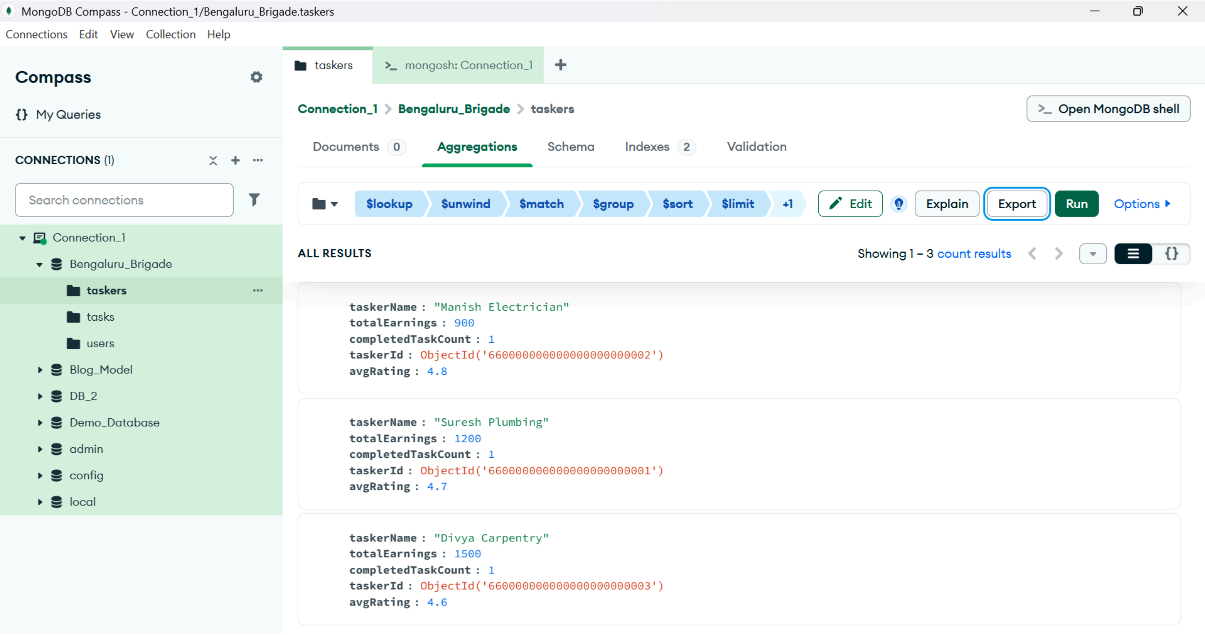
  ```
  {
    "_id": "$_id",
    "taskerName": { "$first": "$taskerName" },
    "avgRating": { "$avg": "$rating" },
    "totalEarnings": { "$sum": "$joinedTasks.cost" },
    "completedTaskCount": { "$sum": 1 }
  }
  ```

- $sort

  ```
  {
    "avgRating": -1,
    "totalEarnings": -1
  }
  ```

- $limit → number = 3
- $project

  ```
  {
    "_id": 0,
    "taskerId": "$_id",
    "taskerName": 1,
    "avgRating": { "$round": ["$avgRating", 2] },
    "totalEarnings": 1,
    "completedTaskCount": 1
  }
  ```



```
taskerName : "Manish Electrician"
totalEarnings : 900
completedTaskCount : 1
taskerId : ObjectId('660000000000000000000002')
avgRating : 4.8

taskerName : "Suresh Plumbing"
totalEarnings : 1200
completedTaskCount : 1
taskerId : ObjectId('660000000000000000000001')
avgRating : 4.7

taskerName : "Divya Carpentry"
totalEarnings : 1500
completedTaskCount : 1
taskerId : ObjectId('660000000000000000000003')
avgRating : 4.6
```

Task 5: Conceptual Justification (The Architect's Mindset)

Question: Explain your schema design choice for the tasks collection. You stored the postedBy and assignedTo fields as _id references. What is the main advantage of this approach? What is the main disadvantage compared to embedding the full user and tasker documents inside the task?

ANSWER →

In the **tasks** collection, using `_id` references for `postedBy` and `assignedTo` keeps the schema **normalized** and avoids duplication of user or tasker information across multiple task documents.

Main advantage is **consistency and maintainability**: if a tasker updates their hourly rate or a user changes their name, we only update it once in the `taskers` or `users` collection, and all related tasks stay accurate.

Also keep the `tasks` documents lighter and avoids storing large embedded objects repeatedly.

The major disadvantage is that queries often require **joins** to fetch complete details, which can add complexity and slightly reduce performance compared to having all user/tasker info directly embedded in each task.