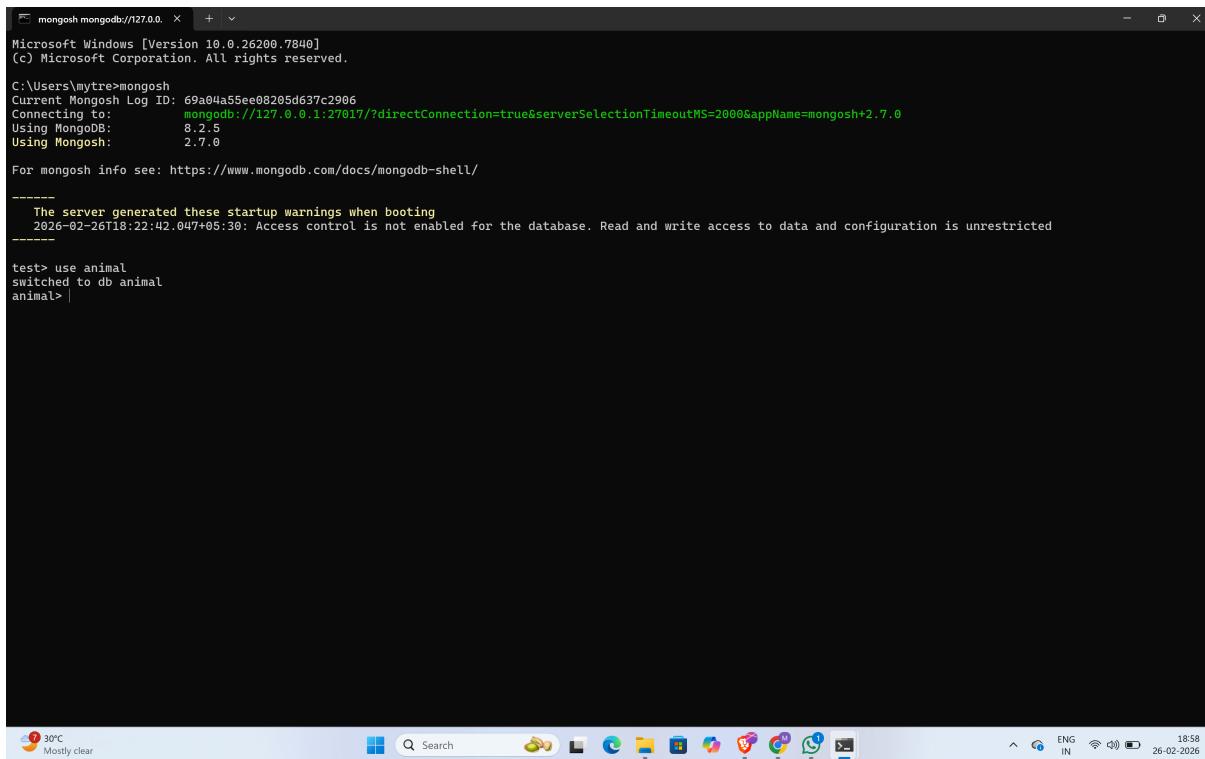


MONGODB - COMMANDS

Animals

2. Use MongoDB to implement the following DB operations for a Zoo

- Create a database called 'animal' and write a MongoDB query to select database as 'animal'.

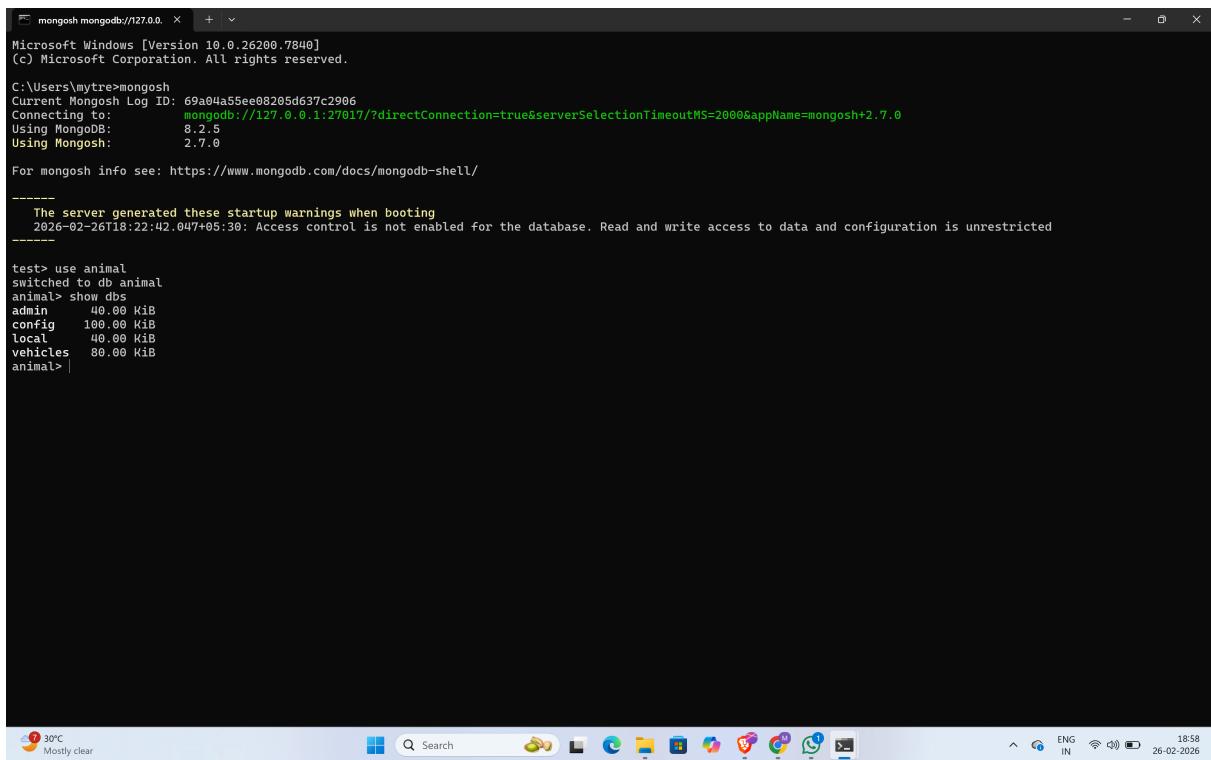


```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

C:\Users\mytre>mongosh
Current Mongosh Log ID: 69a04a55ee08205d637c2906
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Using MongoDB:      8.2.5
Using Mongosh:      2.7.0

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/
-----
The server generated these startup warnings when booting
2026-02-26T18:22:42.047+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----
test> use animal
switched to db animal
animal> |
```

- Write a MongoDB query to display all the databases.



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

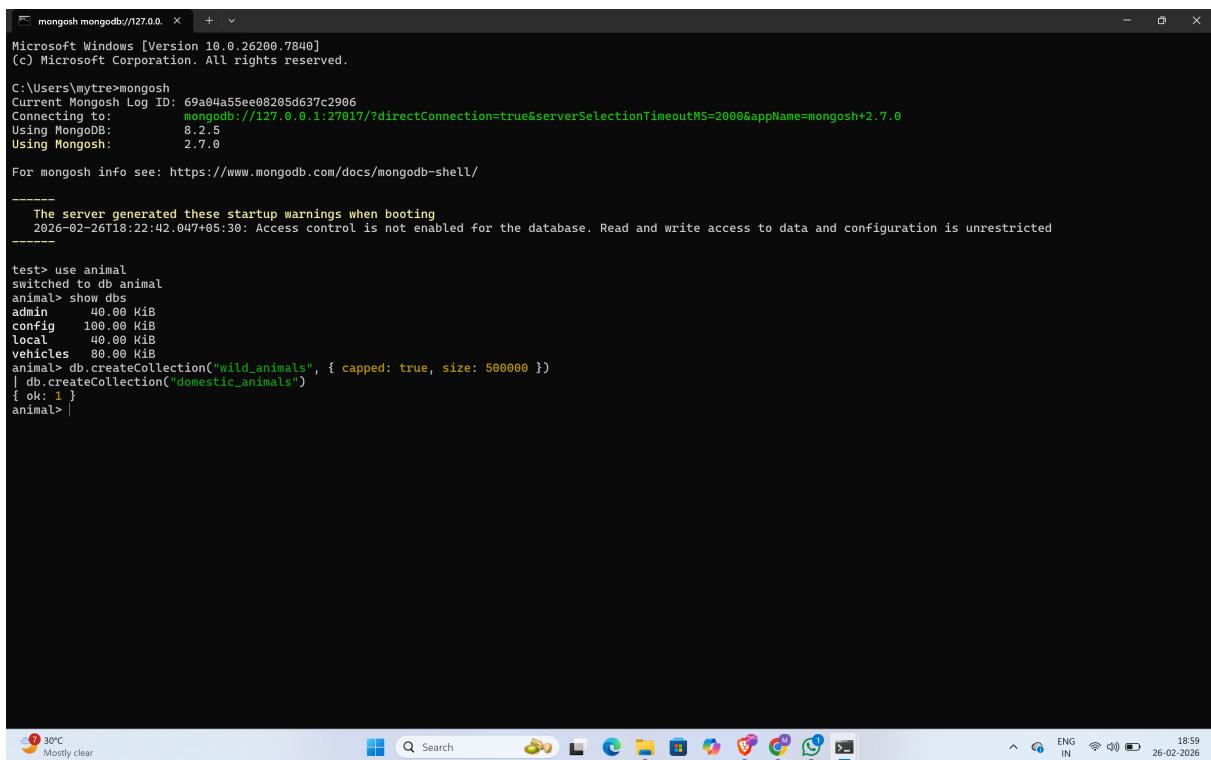
C:\Users\mytre>mongosh
Current Mongosh Log ID: 69a04a55ee08205d637c2906
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Using MongoDB: 8.2.5
Using Mongosh: 2.7.0

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

-----
The server generated these startup warnings when booting
2026-02-26T18:22:42.047+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

test> use animal
switched to db animal
animal> show dbs
admin   40.00 KiB
config  100.00 KiB
local   40.00 KiB
vehicles 80.00 KiB
animal> |
```

- Create a collection called 'wild_animals'.(use capping) and Create a collection called 'domestic_animals'.



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

C:\Users\mytre>mongosh
Current Mongosh Log ID: 69a04a55ee08205d637c2906
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.7.0
Using MongoDB: 8.2.5
Using Mongosh: 2.7.0

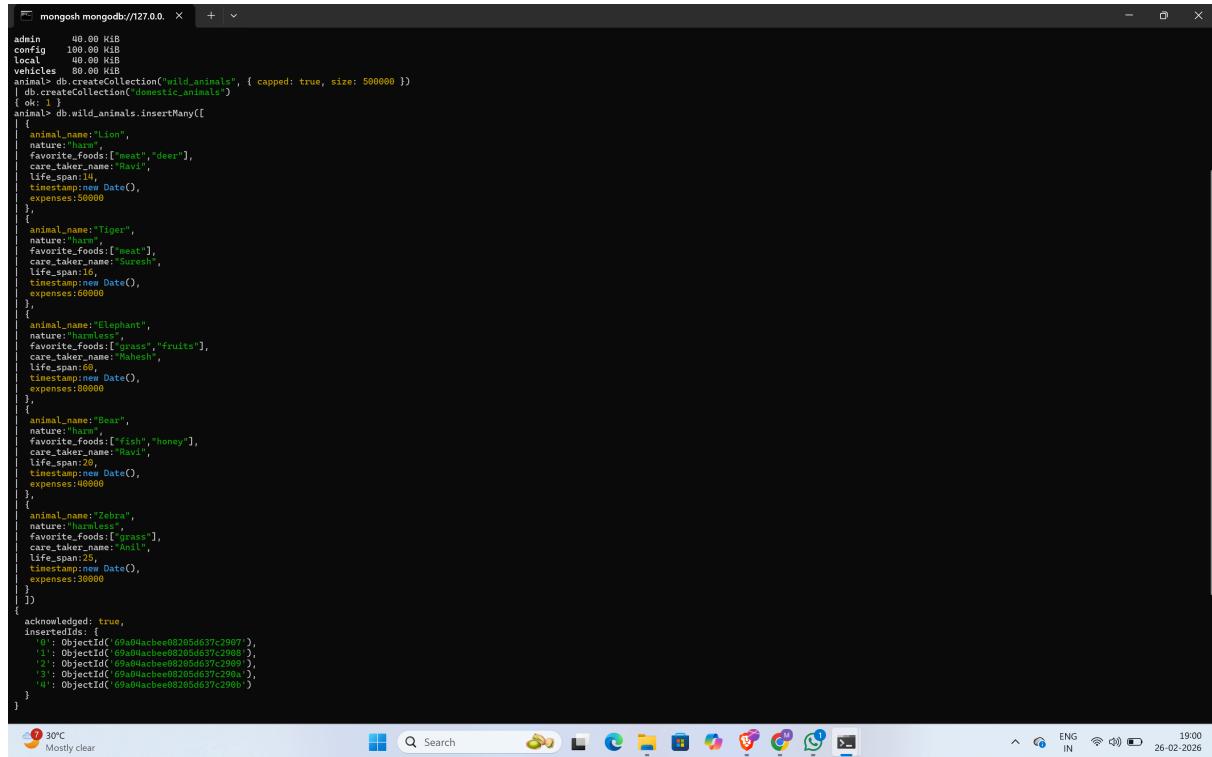
For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

-----
The server generated these startup warnings when booting
2026-02-26T18:22:42.047+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

test> use animal
switched to db animal
animal> show dbs
admin   40.00 KiB
config  100.00 KiB
local   40.00 KiB
vehicles 80.00 KiB
animal> db.createCollection("wild_animals", { capped: true, size: 500000 })
| db.createCollection("domestic_animals")
{ ok: 1 }
animal> |
```

- Add 5 wild_animal details to the collection named 'wild_animals'. Each document consists of following fields as animal_name, nature (harm or harmless),

favorite_foods (meat, rabbits, deer etc) as array, care_taker_name, life span (in years), timestamp (when the animal registered at the Zoo) and expenses.



```

mongosh mongoDB://127.0.0.1:27017
+ - x
admin 40.00 KiB
config 190.00 KiB
local 40.00 KiB
venues 40.00 KiB
wild_animals db.createCollection("wild_animals", { capped: true, size: 500000 })
| db.createCollection("domestic_animals")
{ ok: 1 }
animal> db.wild_animals.insertMany([
  {
    animal_name: "Lion",
    nature: "Harm",
    favorite_foods: ["meat", "deer"],
    care_taker_name: "Ravi",
    life_span: 14,
    timestamp: new Date(),
    expenses: 50000
  },
  {
    animal_name: "Tiger",
    nature: "Harm",
    favorite_foods: ["meat"],
    care_taker_name: "Suresh",
    life_span: 16,
    timestamp: new Date(),
    expenses: 60000
  },
  {
    animal_name: "Elephant",
    nature: "Harmless",
    favorite_foods: ["grass", "fruits"],
    care_taker_name: "Rakesh",
    life_span: 60,
    timestamp: new Date(),
    expenses: 80000
  },
  {
    animal_name: "Bear",
    nature: "Harm",
    favorite_foods: ["fish", "honey"],
    care_taker_name: "Ravi",
    life_span: 20,
    timestamp: new Date(),
    expenses: 40000
  },
  {
    animal_name: "Zebra",
    nature: "Harmless",
    favorite_foods: ["grass"],
    care_taker_name: "Anil",
    life_span: 25,
    timestamp: new Date(),
    expenses: 30000
  }
])
{
  acknowledged: true,
  insertedIds: [
    0: ObjectId('69a9acabee08205d637c2907'),
    1: ObjectId('69a9acabee08205d637c2908'),
    2: ObjectId('69a9acabee08205d637c2909'),
    3: ObjectId('69a9acabee08205d637c290a'),
    4: ObjectId('69a9acabee08205d637c290b')
  ]
}

```

The screenshot shows a MongoDB shell window with the command history and output. The user creates a collection named 'domestic_animals' with a capped size of 500,000 documents. Then, they insert five documents representing different animals: Lion, Tiger, Elephant, Bear, and Zebra. Each document includes fields for name, nature, favorite foods, caretaker name, life span, timestamp, and expenses. The inserted IDs for each document are also listed.

- Add 5 domestic-animal details to the collection named 'domestic_animals'. Each document consists of following fields as animal_name, gender (male or female), favorite_foods (meat, rabbits, deer etc) as array, animal_petname, life span (in years), timestamp (when the animal registered at the Zoo) and expenses.

```

mongosh mongodb://127.0.0.1:27017
[1]: animal> db.domestic_animals.insertMany([
  {
    "id": ObjectId("69a04afadee08205d637c290c"),
    "2": ObjectId("69a04afadee08205d637c290d"),
    "3": ObjectId("69a04afadee08205d637c290e"),
    "4": ObjectId("69a04afadee08205d637c290f")
  }
])
animal> db.domestic_animals.insertMany([
  {
    "animal_name": "Dog",
    "gender": "male",
    "favorite_foods": ["meat", "biscuits"],
    "animal_petname": "Tommy",
    "life_span": 12,
    "timestamp": new Date(),
    "expenses": 10000
  },
  {
    "animal_name": "Cat",
    "gender": "female",
    "favorite_foods": ["milk", "fish"],
    "animal_petname": "Kitty",
    "life_span": 15,
    "timestamp": new Date(),
    "expenses": 8000
  },
  {
    "animal_name": "Cow",
    "gender": "female",
    "favorite_foods": ["grass"],
    "animal_petname": "Ganga",
    "life_span": 20,
    "timestamp": new Date(),
    "expenses": 15000
  },
  {
    "animal_name": "Goat",
    "gender": "male",
    "favorite_foods": ["leaves"],
    "animal_petname": "Ramu",
    "life_span": 18,
    "timestamp": new Date(),
    "expenses": 7000
  },
  {
    "animal_name": "Rabbit",
    "gender": "female",
    "favorite_foods": ["carrot"],
    "animal_petname": "Bunny",
    "life_span": 8,
    "timestamp": new Date(),
    "expenses": 5000
  }
])
animal> [
  {
    acknowledged: true,
    insertedIds: [
      "0": ObjectId("69a04afadee08205d637c290c"),
      "1": ObjectId("69a04afadee08205d637c290d"),
      "2": ObjectId("69a04afadee08205d637c290e"),
      "3": ObjectId("69a04afadee08205d637c290f"),
      "4": ObjectId("69a04afadee08205d637c2910")
    ]
  }
]
animal> |

```

- Write a MongoDB query to display all documents available in wild_animals and domestic_animals.

```

mongosh mongodb://127.0.0.1:27017
[1]: animal> db.wild_animals.find()
| db.domestic_animals.find()
[{
  {
    _id: ObjectId('69a04afadee08205d637c290c'),
    animal_name: 'Dog',
    gender: 'male',
    favorite_foods: [ 'meat', 'biscuits' ],
    animal_petname: 'Tommy',
    life_span: 12,
    timestamp: ISODate('2026-02-26T13:39:37.291Z'),
    expenses: 10000
  },
  {
    _id: ObjectId('69a04afadee08205d637c290d'),
    animal_name: 'Cat',
    gender: 'female',
    favorite_foods: [ 'milk', 'fish' ],
    animal_petname: 'Kitty',
    life_span: 15,
    timestamp: ISODate('2026-02-26T13:30:37.291Z'),
    expenses: 8000
  },
  {
    _id: ObjectId('69a04afadee08205d637c290e'),
    animal_name: 'Cow',
    gender: 'female',
    favorite_foods: [ 'grass' ],
    animal_petname: 'Ganga',
    life_span: 20,
    timestamp: ISODate('2026-02-26T13:30:37.291Z'),
    expenses: 15000
  },
  {
    _id: ObjectId('69a04afadee08205d637c290f'),
    animal_name: 'Goat',
    gender: 'male',
    favorite_foods: [ 'leaves' ],
    animal_petname: 'Ramu',
    life_span: 18,
    timestamp: ISODate('2026-02-26T13:30:37.291Z'),
    expenses: 7000
  },
  {
    _id: ObjectId('69a04afadee08205d637c2910'),
    animal_name: 'Rabbit',
    gender: 'female',
    favorite_foods: [ 'carrot' ],
    animal_petname: 'Bunny',
    life_span: 8,
    timestamp: ISODate('2026-02-26T13:30:37.291Z'),
    expenses: 5000
  }
}]
animal> |

```

- Write a MongoDB query to display only animal name and expenses in all the collection of the database

```

mongosh mongodb://127.0.0.1:27017
expenses: 10000
},
{
  _id: ObjectId('69a04afdee08205d637c290d'),
  animal_name: 'Cat',
  gender: 'Female',
  favorite_foods: ['milk', 'fish'],
  animal_petname: 'Kitty',
  life_span: 15,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 8000
},
{
  _id: ObjectId('69a04afdee08205d637c290e'),
  animal_name: 'Cow',
  gender: 'Female',
  favorite_foods: ['grass'],
  animal_petname: 'Ganga',
  life_span: 20,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 15000
},
{
  _id: ObjectId('69a04afdee08205d637c290f'),
  animal_name: 'Goat',
  gender: 'Male',
  favorite_foods: ['leaves'],
  animal_petname: 'Ramu',
  life_span: 18,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 7000
},
{
  _id: ObjectId('69a04afdee08205d637c2910'),
  animal_name: 'Rabbit',
  gender: 'Female',
  favorite_foods: ['carrot'],
  animal_petname: 'Bunny',
  life_span: 8,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 5000
}
]
animal> db.wild_animals.find({}, {animal_name:1, expenses:1, _id:0})
| db.domestic_animals.find({}, {animal_name:1, expenses:1, _id:0})
[
  { animal_name: 'Dog', expenses: 10000 },
  { animal_name: 'Cat', expenses: 8000 },
  { animal_name: 'Cow', expenses: 15000 },
  { animal_name: 'Goat', expenses: 7000 },
  { animal_name: 'Rabbit', expenses: 5000 }
]
animal> |

```

30°C Mostly clear Search ENG IN 19:01 26-02-2026

- Write a MongoDB query to display domestic_animals whose life is a particular year

```

mongosh mongodb://127.0.0.1:27017
{
  _id: ObjectId('69a04afdee08205d637c290e'),
  animal_name: 'Cow',
  gender: 'Female',
  favorite_foods: ['grass'],
  animal_petname: 'Ganga',
  life_span: 20,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 15000
},
{
  _id: ObjectId('69a04afdee08205d637c290f'),
  animal_name: 'Goat',
  gender: 'Male',
  favorite_foods: ['leaves'],
  animal_petname: 'Ramu',
  life_span: 18,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 7000
},
{
  _id: ObjectId('69a04afdee08205d637c2910'),
  animal_name: 'Rabbit',
  gender: 'Female',
  favorite_foods: ['carrot'],
  animal_petname: 'Bunny',
  life_span: 8,
  timestamp: ISODate('2026-02-26T13:30:37.291Z'),
  expenses: 5000
}
]
animal> db.wild_animals.find({}, {animal_name:1, expenses:1, _id:0})
| db.domestic_animals.find({}, {animal_name:1, expenses:1, _id:0})
[
  { animal_name: 'Dog', expenses: 10000 },
  { animal_name: 'Cat', expenses: 8000 },
  { animal_name: 'Cow', expenses: 15000 },
  { animal_name: 'Goat', expenses: 7000 },
  { animal_name: 'Rabbit', expenses: 5000 }
]
animal> db.domestic_animals.find({life_span:15})
[
  {
    _id: ObjectId('69a04afdee08205d637c290d'),
    animal_name: 'Cat',
    gender: 'Female',
    favorite_foods: ['milk', 'fish'],
    animal_petname: 'Kitty',
    life_span: 15,
    timestamp: ISODate('2026-02-26T13:30:37.291Z'),
    expenses: 8000
  }
]
animal> |

```

30°C Mostly clear Search ENG IN 19:02 26-02-2026

- Write a MongoDB query to display wild_animals available under a particular care_taker

```
mongosh mongodb://127.0.0.1:27017
[1] animal> db.wild_animals.find({}, {animal_name:1, expenses:1, _id:0})
[2] | db.domestic_animals.find({}, {animal_name:1, expenses:1, _id:0})
[3] [
[{"animal_name": "Dog", "expenses": 10000}, {"animal_name": "Cat", "expenses": 8000}, {"animal_name": "Cow", "expenses": 15000}, {"animal_name": "Goat", "expenses": 7000}, {"animal_name": "Rabbit", "expenses": 5000}], animal> db.domestic_animals.find({life_span:15})
[4] [
[{"_id": ObjectId("69a04acbee08205d637c290d"), "animal_name": "Cat", "gender": "female", "favorite_foods": ["milk", "fish"], "animal_petname": "kitty", "life_span": 15, "timestamp": ISODate("2026-02-26T13:30:37.291Z"), "expenses": 8000}], animal> db.wild_animals.find({care_taker_name:"Ravi"})
[5] [
[{"_id": ObjectId("69a04acbee08205d637c2907"), "animal_name": "Lion", "nature": "harm", "favorite_foods": ["meat", "deer"], "care_taker_name": "Ravi", "life_span": 14, "timestamp": ISODate("2026-02-26T13:29:47.967Z"), "expenses": 50000}, {"_id": ObjectId("69a04acbee08205d637c290a"), "animal_name": "Bear", "nature": "harm", "favorite_foods": ["fish", "honey"], "care_taker_name": "Ravi", "life_span": 20, "timestamp": ISODate("2026-02-26T13:29:47.967Z"), "expenses": 40000}], animal> |

```

- Write a MongoDB query to display animal name, favorite_foods and expenses details whose lifespan is more than 5 years.

```
mongosh mongodb://127.0.0.1:27017
[1] animal> db.wild_animals.find({care_taker_name:"Ravi"})
[2] [
[{"_id": ObjectId("69a04acbee08205d637c2907"), "animal_name": "Lion", "nature": "harm", "favorite_foods": ["meat", "deer"], "care_taker_name": "Ravi", "life_span": 14, "timestamp": ISODate("2026-02-26T13:29:47.967Z"), "expenses": 50000}, {"_id": ObjectId("69a04acbee08205d637c290a"), "animal_name": "Bear", "nature": "harm", "favorite_foods": ["fish", "honey"], "care_taker_name": "Ravi", "life_span": 20, "timestamp": ISODate("2026-02-26T13:29:47.967Z"), "expenses": 40000}], animal> db.wild_animals.find(
[{"life_span:{gt:5}}, {"animal_name:1, favorite_foods:1, expenses:1, _id:0}
])
[3] |
[4] db.domestic_animals.find(
[{"life_span:{gt:5}}, {"animal_name:1, favorite_foods:1, expenses:1, _id:0}
])
[5] [
[{"animal_name": "Dog", "favorite_foods": ["meat", "biscuits"], "expenses": 10000}, {"animal_name": "Cat", "favorite_foods": ["milk", "fish"], "expenses": 8000}, {"animal_name": "Cow", "favorite_foods": ["grass"], "expenses": 15000}, {"animal_name": "Goat", "favorite_foods": ["leaves"], "expenses": 7000}, {"animal_name": "Rabbit", "favorite_foods": ["carrot"], "expenses": 5000}], animal> |

```