VIT-AP UNIVERSITY, ANDHRA PRADESH

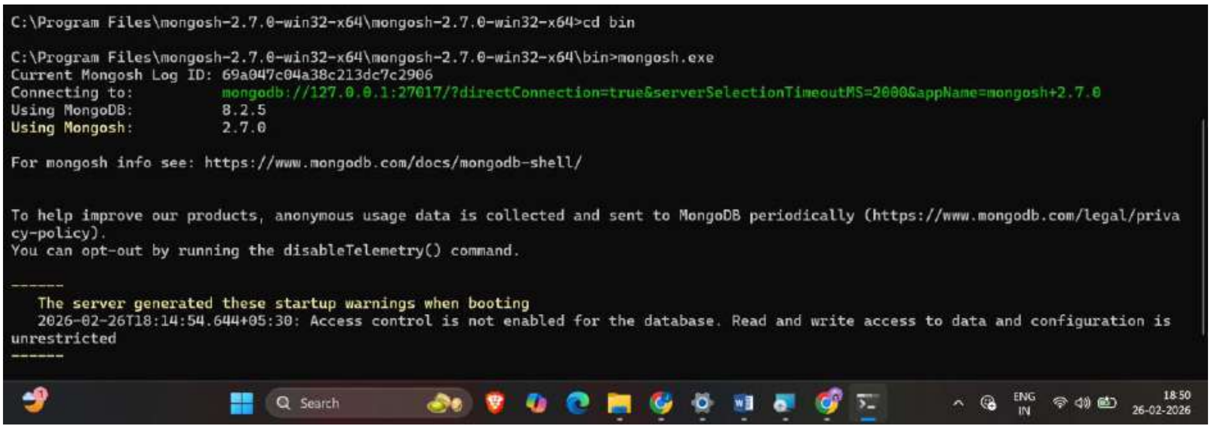
Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395

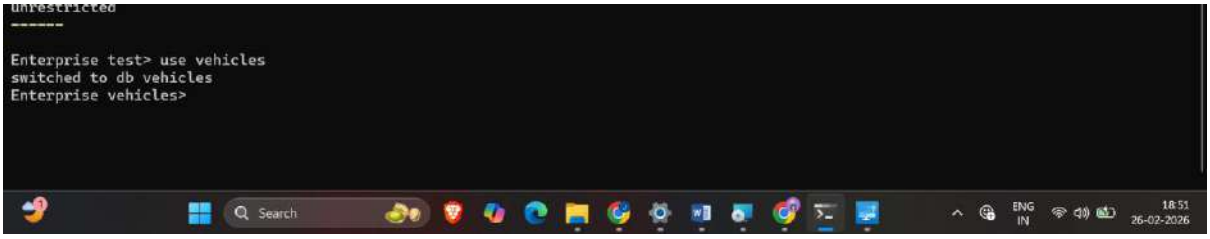
---

1. Use MongoDB to implement the following DB operations



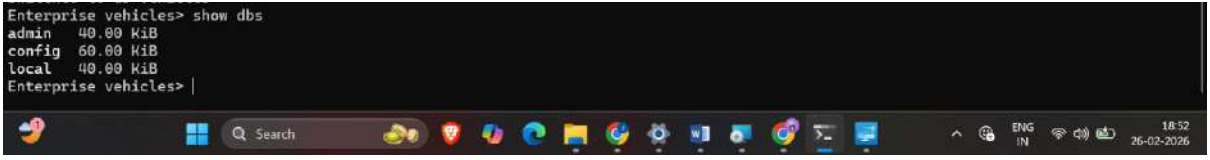
```
C:\Program Files\mongosh-2.7.0-win32-x64\mongosh-2.7.0-win32-x64>cd bin

C:\Program Files\mongosh-2.7.0-win32-x64\mongosh-2.7.0-win32-x64\bin>mongosh.exe
Current Mongosh Log ID: 69a047c04a38c213dc7c2906
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/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/priva
cy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2026-02-26T18:14:54.644+05:30: Access control is not enabled for the database. Read and write access to data and configuration is
unrestricted
------
```

1. Create a database called 'vehicles' and *write* a MongoDB query to select database as "vehicles".



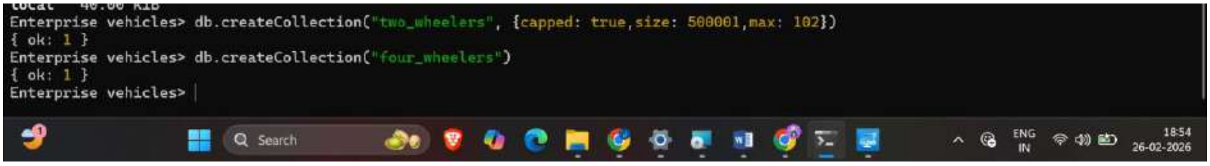
```
unrestricted
------

Enterprise test> use vehicles
switched to db vehicles
Enterprise vehicles>
```

2. Write a MongoDB query to display all the databases.



```
Enterprise vehicles> show dbs
admin   40.00 KiB
config  60.00 KiB
local   40.00 KiB
Enterprise vehicles> |
```

3. Create a collection called 'two_wheelers'. (use capping) and Create a collection called 'four_wheelers'.



```
local   40.00 KiB
Enterprise vehicles> db.createCollection("two_wheelers", {capped: true,size: 500001,max: 102})
{ ok: 1 }
Enterprise vehicles> db.createCollection("four_wheelers")
{ ok: 1 }
Enterprise vehicles> |
```

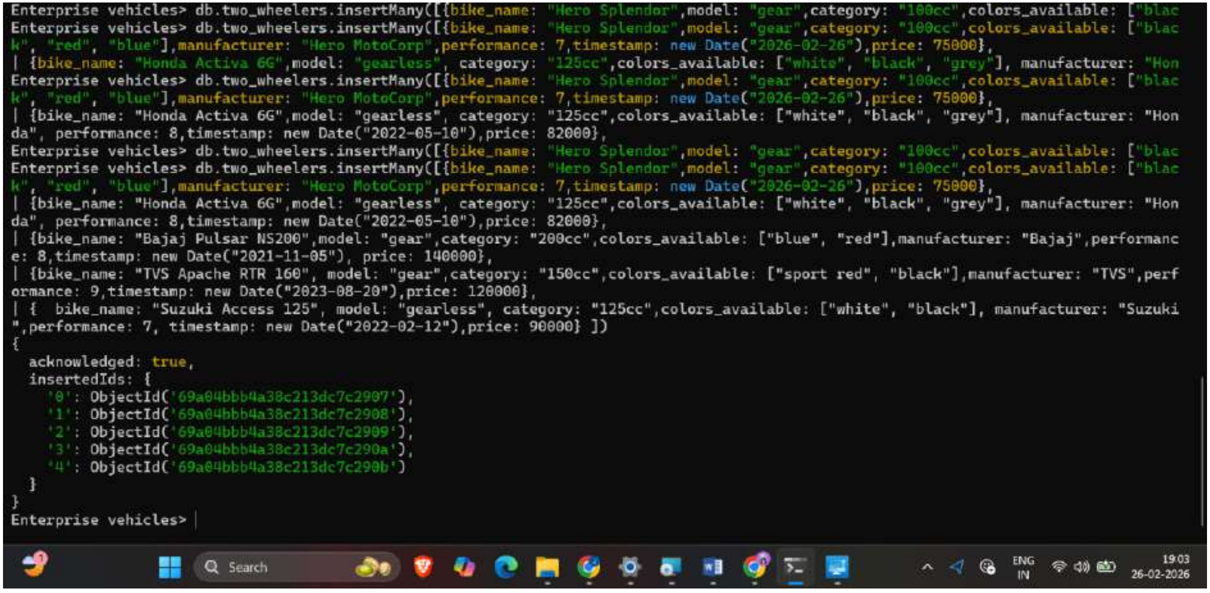
VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

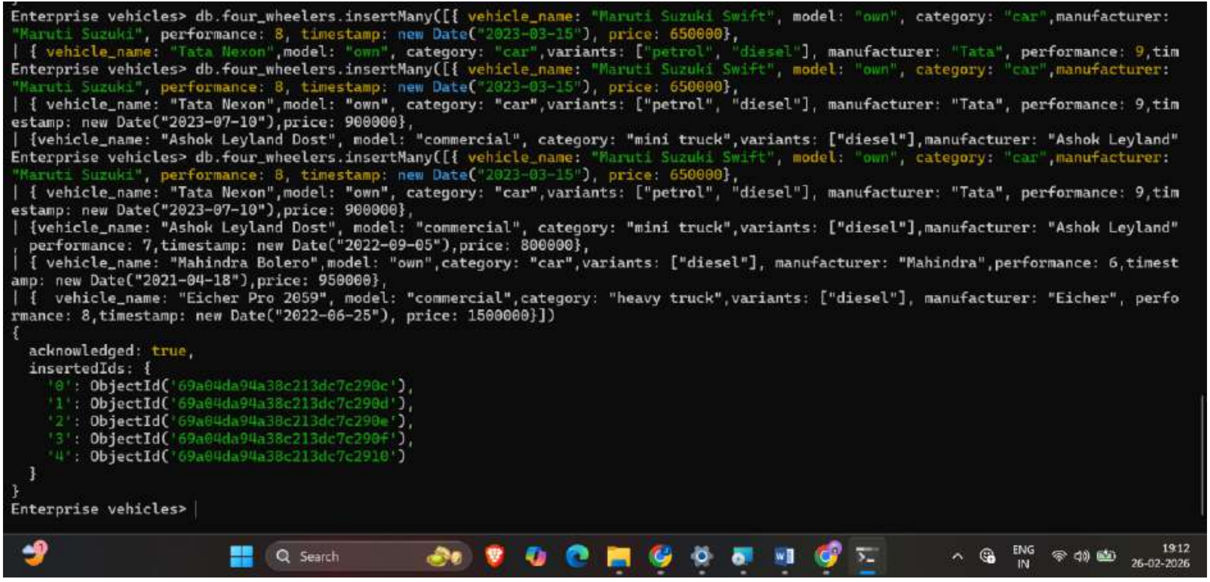
Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395

4. Add 5 two-wheeler details to the collection named 'two_wheelers'. Each document consists of following fields as bike_name, model (gear or gearless), category (100cc, 125cc, 150cc, 200cc), colors_available (red, black, blue, sport red etc) as array, manufacturer, performance (out of 10), timestamp (date and year release) and price.



```
Enterprise vehicles> db.two_wheelers.insertMany([{bike_name: "Hero Splendor",model: "gear",category: "100cc",colors_available: ["blac
Enterprise vehicles> db.two_wheelers.insertMany([{bike_name: "Hero Splendor",model: "gear",category: "100cc",colors_available: ["blac
k", "red", "blue"],manufacturer: "Hero MotoCorp",performance: 7,timestamp: new Date("2026-02-26"),price: 75000},
| {bike_name: "Honda Activa 6G",model: "gearless", category: "125cc",colors_available: ["white", "black", "grey"], manufacturer: "Hon
Enterprise vehicles> db.two_wheelers.insertMany([{bike_name: "Hero Splendor",model: "gear",category: "100cc",colors_available: ["blac
k", "red", "blue"],manufacturer: "Hero MotoCorp",performance: 7,timestamp: new Date("2026-02-26"),price: 75000},
| {bike_name: "Honda Activa 6G",model: "gearless", category: "125cc",colors_available: ["white", "black", "grey"], manufacturer: "Hon
da", performance: 8,timestamp: new Date("2022-05-10"),price: 82000},
Enterprise vehicles> db.two_wheelers.insertMany([{bike_name: "Hero Splendor",model: "gear",category: "100cc",colors_available: ["blac
Enterprise vehicles> db.two_wheelers.insertMany([{bike_name: "Hero Splendor",model: "gear",category: "100cc",colors_available: ["blac
k", "red", "blue"],manufacturer: "Hero MotoCorp",performance: 7,timestamp: new Date("2026-02-26"),price: 75000},
| {bike_name: "Honda Activa 6G",model: "gearless", category: "125cc",colors_available: ["white", "black", "grey"], manufacturer: "Hon
da", performance: 8,timestamp: new Date("2022-05-10"),price: 82000},
| {bike_name: "Bajaj Pulsar NS200",model: "gear",category: "200cc",colors_available: ["blue", "red"],manufacturer: "Bajaj",performanc
e: 8,timestamp: new Date("2021-11-05"), price: 140000},
| {bike_name: "TVS Apache RTR 160", model: "gear",category: "150cc",colors_available: ["sport red", "black"],manufacturer: "TVS",perf
ormance: 9,timestamp: new Date("2023-08-20"),price: 120000},
| {  bike_name: "Suzuki Access 125", model: "gearless", category: "125cc",colors_available: ["white", "black"], manufacturer: "Suzuki
",performance: 7, timestamp: new Date("2022-02-12"),price: 90000} ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('69a04bbb4a38c213dc7c2907'),
    '1': ObjectId('69a04bbb4a38c213dc7c2908'),
    '2': ObjectId('69a04bbb4a38c213dc7c2909'),
    '3': ObjectId('69a04bbb4a38c213dc7c290a'),
    '4': ObjectId('69a04bbb4a38c213dc7c290b')
  }
}
Enterprise vehicles>
```

5. Add 5 four-wheeler details to the collection named 'four_wheelers'. Each document consists of following fields as vehicle_name, model (commercial or own), category (car, lorry, bus, mini truck, heavy truck, containers), variants (vxi, zxi, petrol, diesel etc) as array, manufacturer, performance (out of 10), timestamp (date and year release) and price.



```
Enterprise vehicles> db.four_wheelers.insertMany([{ vehicle_name: "Maruti Suzuki Swift", model: "own", category: "car",manufacturer:
"Maruti Suzuki", performance: 8, timestamp: new Date("2023-03-15"), price: 650000},
| { vehicle_name: "Tata Nexon",model: "own", category: "car",variants: ["petrol", "diesel"], manufacturer: "Tata", performance: 9,tim
Enterprise vehicles> db.four_wheelers.insertMany([{ vehicle_name: "Maruti Suzuki Swift", model: "own", category: "car",manufacturer:
"Maruti Suzuki", performance: 8, timestamp: new Date("2023-03-15"), price: 650000},
| { vehicle_name: "Tata Nexon",model: "own", category: "car",variants: ["petrol", "diesel"], manufacturer: "Tata", performance: 9,tim
estamp: new Date("2023-07-10"),price: 900000},
| {vehicle_name: "Ashok Leyland Dost", model: "commercial", category: "mini truck",variants: ["diesel"],manufacturer: "Ashok Leyland"
Enterprise vehicles> db.four_wheelers.insertMany([{ vehicle_name: "Maruti Suzuki Swift", model: "own", category: "car",manufacturer:
"Maruti Suzuki", performance: 8, timestamp: new Date("2023-03-15"), price: 650000},
| { vehicle_name: "Tata Nexon",model: "own", category: "car",variants: ["petrol", "diesel"], manufacturer: "Tata", performance: 9,tim
estamp: new Date("2023-07-10"),price: 900000},
| {vehicle_name: "Ashok Leyland Dost", model: "commercial", category: "mini truck",variants: ["diesel"],manufacturer: "Ashok Leyland"
, performance: 7,timestamp: new Date("2022-09-05"),price: 800000},
| { vehicle_name: "Mahindra Bolero",model: "own",category: "car",variants: ["diesel"], manufacturer: "Mahindra",performance: 6,timest
amp: new Date("2021-04-18"),price: 950000},
| {  vehicle_name: "Eicher Pro 2059", model: "commercial",category: "heavy truck",variants: ["diesel"], manufacturer: "Eicher", perfo
rmance: 8,timestamp: new Date("2022-06-25"), price: 1500000}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('69a04da94a38c213dc7c290c'),
    '1': ObjectId('69a04da94a38c213dc7c290d'),
    '2': ObjectId('69a04da94a38c213dc7c290e'),
    '3': ObjectId('69a04da94a38c213dc7c290f'),
    '4': ObjectId('69a04da94a38c213dc7c2910')
  }
}
Enterprise vehicles>
```

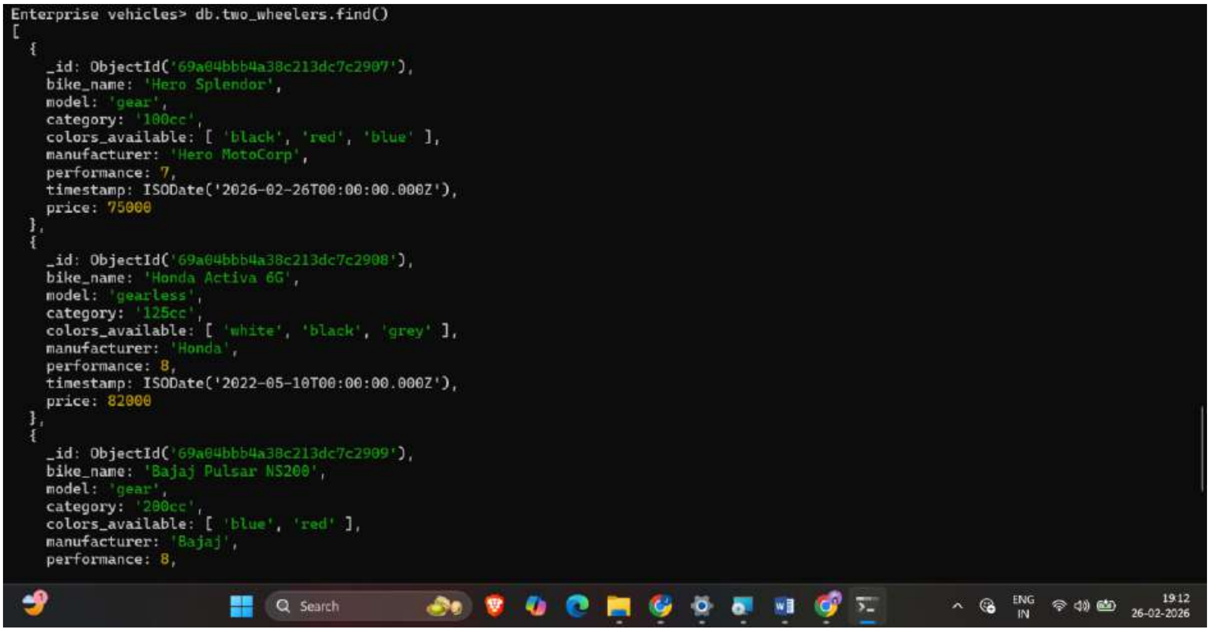
VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

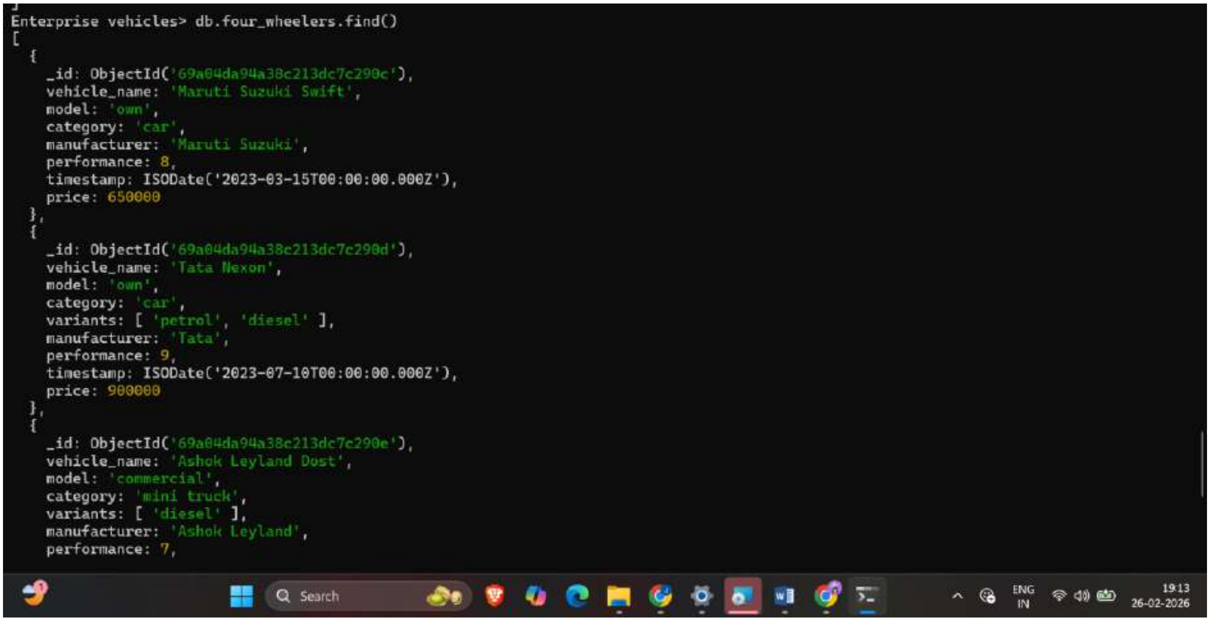
Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395

6. Write a MongoDB query to display all documents available in two_wheelers and four_wheelers.



```
Enterprise vehicles> db.two_wheelers.find()
[
  {
    _id: ObjectId('69a04bbb4a38c213dc7c2907'),
    bike_name: 'Hero Splendor',
    model: 'gear',
    category: '100cc',
    colors_available: [ 'black', 'red', 'blue' ],
    manufacturer: 'Hero MotoCorp',
    performance: 7,
    timestamp: ISODate('2026-02-26T00:00:00.000Z'),
    price: 75000
  },
  {
    _id: ObjectId('69a04bbb4a38c213dc7c2908'),
    bike_name: 'Honda Activa 6G',
    model: 'gearless',
    category: '125cc',
    colors_available: [ 'white', 'black', 'grey' ],
    manufacturer: 'Honda',
    performance: 8,
    timestamp: ISODate('2022-05-10T00:00:00.000Z'),
    price: 82000
  },
  {
    _id: ObjectId('69a04bbb4a38c213dc7c2909'),
    bike_name: 'Bajaj Pulsar NS200',
    model: 'gear',
    category: '200cc',
    colors_available: [ 'blue', 'red' ],
    manufacturer: 'Bajaj',
    performance: 8,
```



```
Enterprise vehicles> db.four_wheelers.find()
[
  {
    _id: ObjectId('69a04da94a38c213dc7c290c'),
    vehicle_name: 'Maruti Suzuki Swift',
    model: 'own',
    category: 'car',
    manufacturer: 'Maruti Suzuki',
    performance: 8,
    timestamp: ISODate('2023-03-15T00:00:00.000Z'),
    price: 650000
  },
  {
    _id: ObjectId('69a04da94a38c213dc7c290d'),
    vehicle_name: 'Tata Nexon',
    model: 'own',
    category: 'car',
    variants: [ 'petrol', 'diesel' ],
    manufacturer: 'Tata',
    performance: 9,
    timestamp: ISODate('2023-07-10T00:00:00.000Z'),
    price: 900000
  },
  {
    _id: ObjectId('69a04da94a38c213dc7c290e'),
    vehicle_name: 'Ashok Leyland Dost',
    model: 'commercial',
    category: 'mini truck',
    variants: [ 'diesel' ],
    manufacturer: 'Ashok Leyland',
    performance: 7,
```

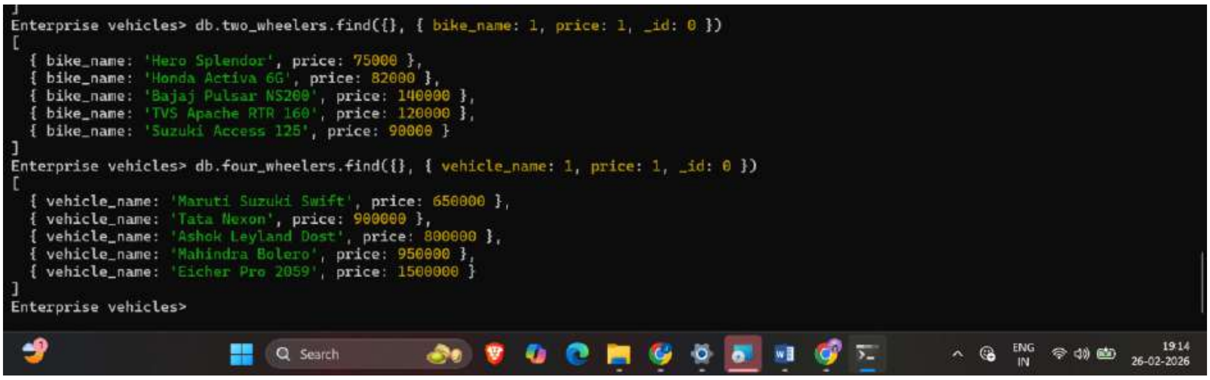
VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

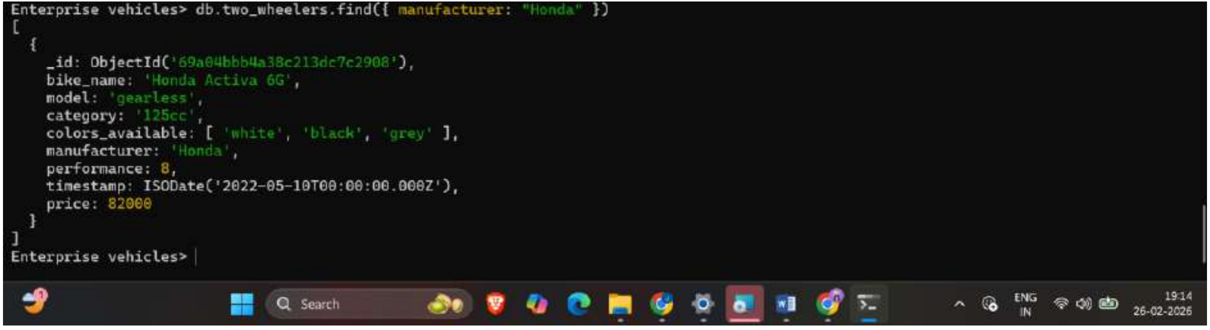
Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395

7. Write a MongoDB query to display only vehicle name and price in all the collection of the database



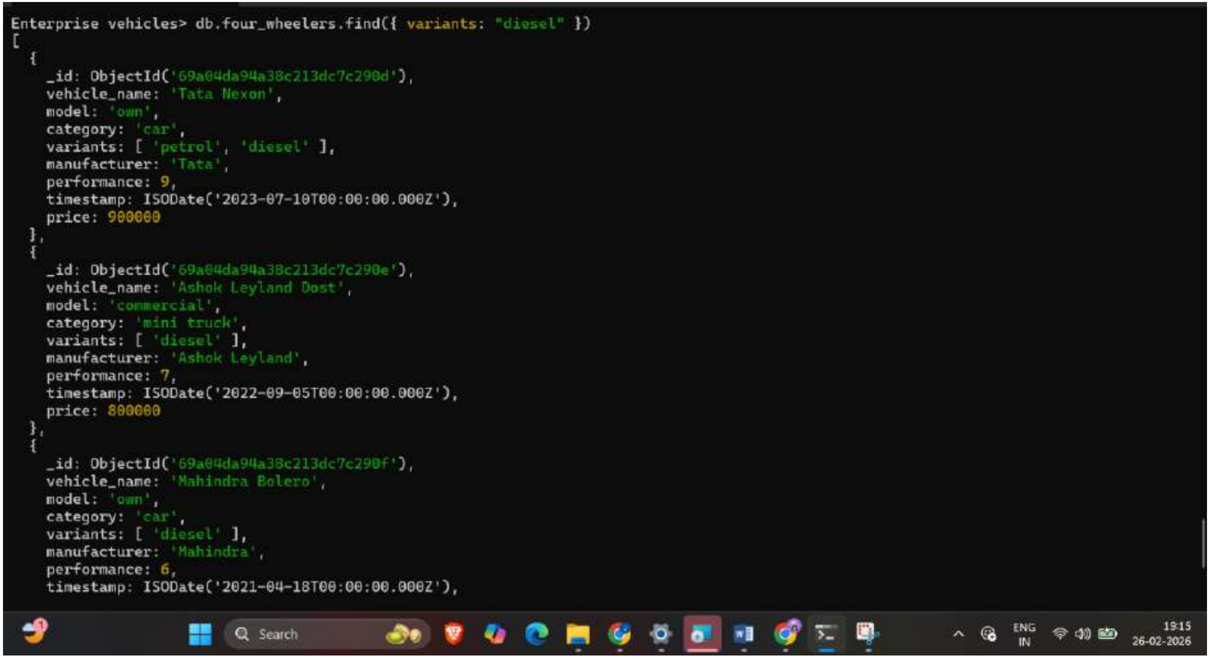
```
Enterprise vehicles> db.two_wheelers.find({}, { bike_name: 1, price: 1, _id: 0 })
[
  { bike_name: 'Hero Splendor', price: 75000 },
  { bike_name: 'Honda Activa 6G', price: 82000 },
  { bike_name: 'Bajaj Pulsar NS200', price: 140000 },
  { bike_name: 'TVS Apache RTR 160', price: 120000 },
  { bike_name: 'Suzuki Access 125', price: 90000 }
]
Enterprise vehicles> db.four_wheelers.find({}, { vehicle_name: 1, price: 1, _id: 0 })
[
  { vehicle_name: 'Maruti Suzuki Swift', price: 650000 },
  { vehicle_name: 'Tata Nexon', price: 900000 },
  { vehicle_name: 'Ashok Leyland Dost', price: 800000 },
  { vehicle_name: 'Mahindra Bolero', price: 950000 },
  { vehicle_name: 'Eicher Pro 2059', price: 1500000 }
]
Enterprise vehicles>
```

8. Write a MongoDB query to display two_wheelers from a particular company.



```
Enterprise vehicles> db.two_wheelers.find({ manufacturer: "Honda" })
[
  {
    _id: ObjectId('69a04bbb4a38c213dc7c2908'),
    bike_name: 'Honda Activa 6G',
    model: 'gearless',
    category: '125cc',
    colors_available: [ 'white', 'black', 'grey' ],
    manufacturer: 'Honda',
    performance: 8,
    timestamp: ISODate('2022-05-10T00:00:00.000Z'),
    price: 82000
  }
]
Enterprise vehicles>
```

9. Write a MongoDB query to display four_wheelers available in diesel variants.



```
Enterprise vehicles> db.four_wheelers.find({ variants: "diesel" })
[
  {
    _id: ObjectId('69a04da94a38c213dc7c290d'),
    vehicle_name: 'Tata Nexon',
    model: 'own',
    category: 'car',
    variants: [ 'petrol', 'diesel' ],
    manufacturer: 'Tata',
    performance: 9,
    timestamp: ISODate('2023-07-10T00:00:00.000Z'),
    price: 900000
  },
  {
    _id: ObjectId('69a04da94a38c213dc7c290e'),
    vehicle_name: 'Ashok Leyland Dost',
    model: 'commercial',
    category: 'mini truck',
    variants: [ 'diesel' ],
    manufacturer: 'Ashok Leyland',
    performance: 7,
    timestamp: ISODate('2022-09-05T00:00:00.000Z'),
    price: 800000
  },
  {
    _id: ObjectId('69a04da94a38c213dc7c290f'),
    vehicle_name: 'Mahindra Bolero',
    model: 'own',
    category: 'car',
    variants: [ 'diesel' ],
    manufacturer: 'Mahindra',
    performance: 6,
    timestamp: ISODate('2021-04-18T00:00:00.000Z'),
```

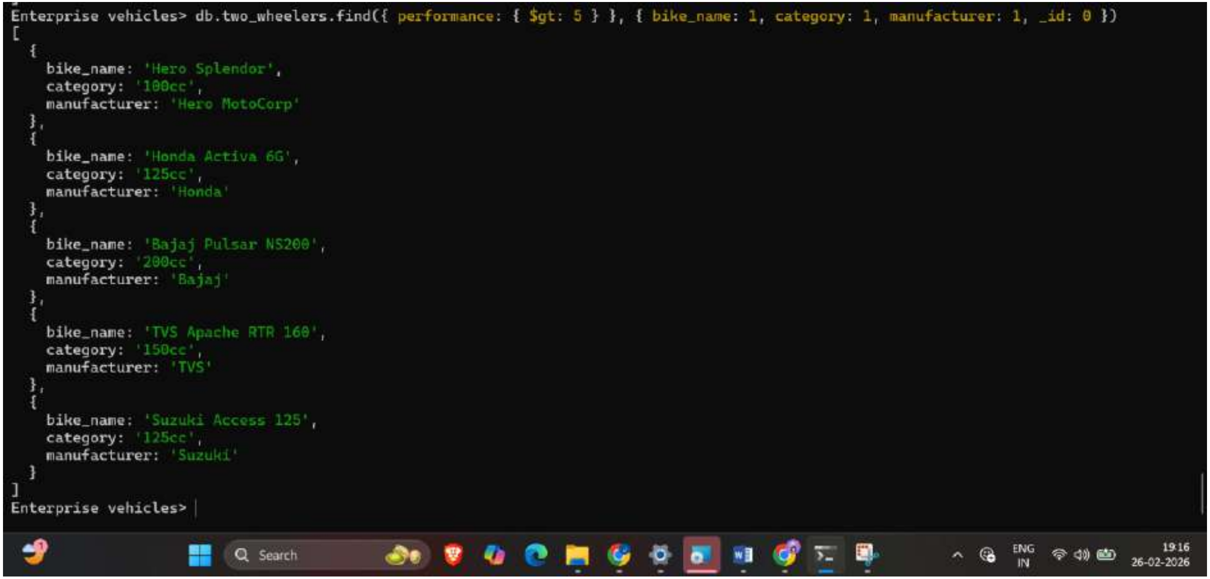
VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

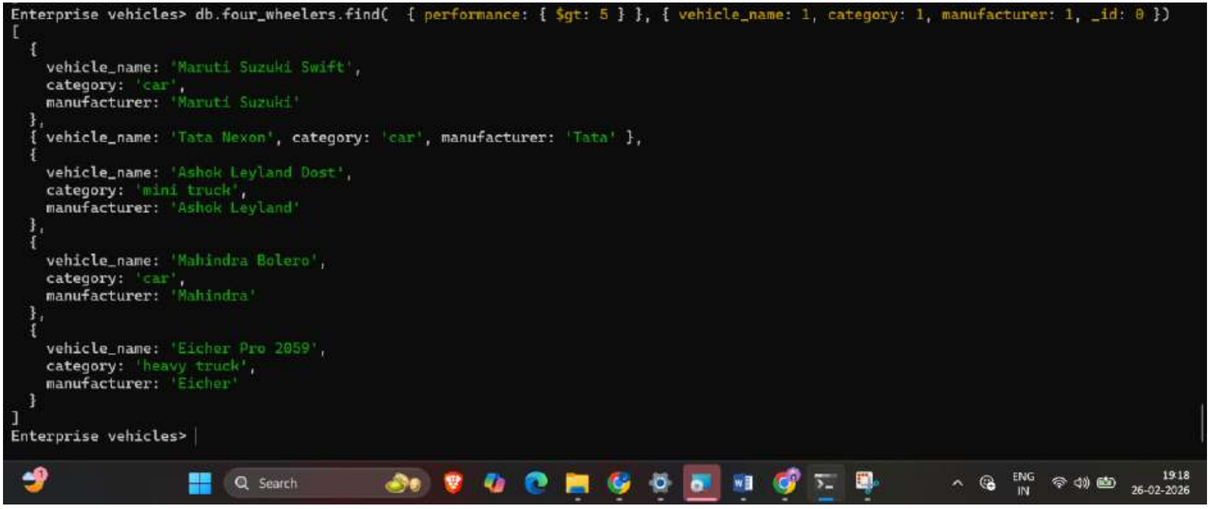
Branch/ Class: B.Tech/M.Tech **Date:** 26-02-2026
Faculty Name: Prof. S.Gopikrishnan **School:** SCOPE
Student name: Jahnavi Naidu **Reg. no.:** 23BCE8395

---

10. Write a MongoDB query to display vehicles name, category and manufacturer details whose rating is more than 5.



```
Enterprise vehicles> db.two_wheelers.find({ performance: { $gt: 5 } }, { bike_name: 1, category: 1, manufacturer: 1, _id: 0 })
[
  {
    bike_name: 'Hero Splendor',
    category: '100cc',
    manufacturer: 'Hero MotoCorp'
  },
  {
    bike_name: 'Honda Activa 6G',
    category: '125cc',
    manufacturer: 'Honda'
  },
  {
    bike_name: 'Bajaj Pulsar NS200',
    category: '200cc',
    manufacturer: 'Bajaj'
  },
  {
    bike_name: 'TVS Apache RTR 160',
    category: '150cc',
    manufacturer: 'TVS'
  },
  {
    bike_name: 'Suzuki Access 125',
    category: '125cc',
    manufacturer: 'Suzuki'
  }
]
Enterprise vehicles>
```



```
Enterprise vehicles> db.four_wheelers.find(  { performance: { $gt: 5 } }, { vehicle_name: 1, category: 1, manufacturer: 1, _id: 0 })
[
  {
    vehicle_name: 'Maruti Suzuki Swift',
    category: 'car',
    manufacturer: 'Maruti Suzuki'
  },
  { vehicle_name: 'Tata Nexon', category: 'car', manufacturer: 'Tata' },
  {
    vehicle_name: 'Ashok Leyland Dost',
    category: 'mini truck',
    manufacturer: 'Ashok Leyland'
  },
  {
    vehicle_name: 'Mahindra Bolero',
    category: 'car',
    manufacturer: 'Mahindra'
  },
  {
    vehicle_name: 'Eicher Pro 2059',
    category: 'heavy truck',
    manufacturer: 'Eicher'
  }
]
Enterprise vehicles>
```

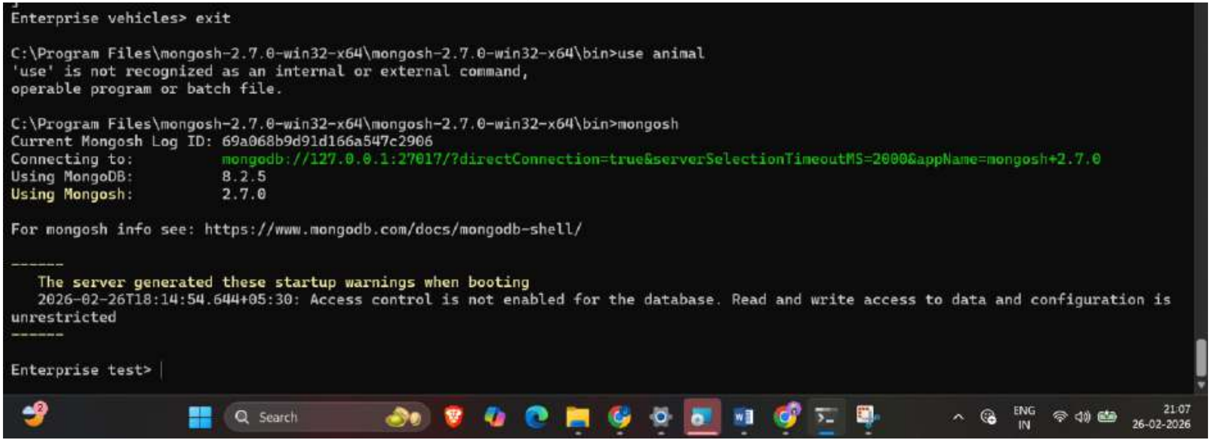
VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech **Date:** 26-02-2026
Faculty Name: Prof. S.Gopikrishnan **School:** SCOPE
Student name: Jahnavi Naidu **Reg. no.:** 23BCE8395

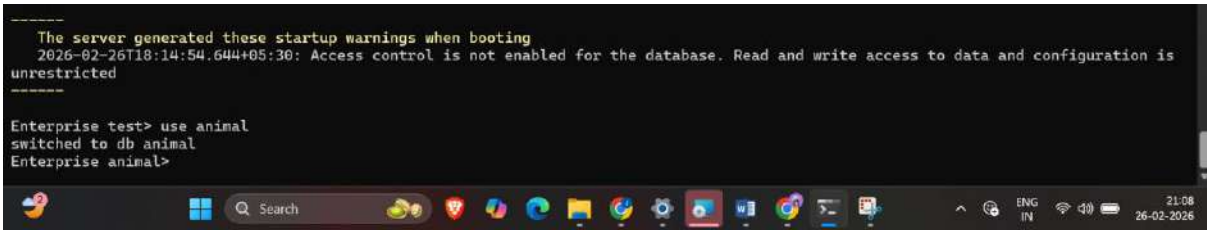
---

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



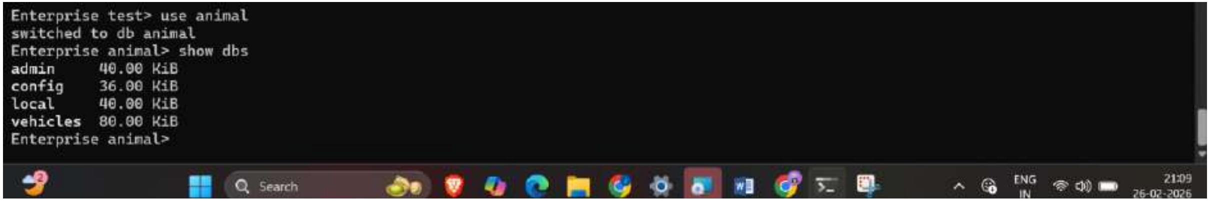
```
Enterprise vehicles> exit

C:\Program Files\mongosh-2.7.0-win32-x64\mongosh-2.7.0-win32-x64\bin>use animal
'use' is not recognized as an internal or external command,
operable program or batch file.

C:\Program Files\mongosh-2.7.0-win32-x64\mongosh-2.7.0-win32-x64\bin>mongosh
Current Mongosh Log ID: 69a068b9d91d166a547c2906
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
   2026-02-26T18:14:54.644+05:30: Access control is not enabled for the database. Read and write access to data and configuration is
unrestricted
------

Enterprise test>
```

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



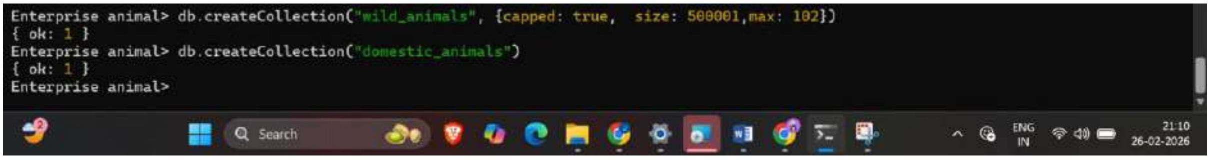
```
------
   The server generated these startup warnings when booting
   2026-02-26T18:14:54.644+05:30: Access control is not enabled for the database. Read and write access to data and configuration is
unrestricted
------

Enterprise test> use animal
switched to db animal
Enterprise animal>
```

2. Write a MongoDB query to display all the databases.



```
Enterprise test> use animal
switched to db animal
Enterprise animal> show dbs
admin     40.00 KiB
config    36.00 KiB
local     40.00 KiB
vehicles  80.00 KiB
Enterprise animal>
```

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



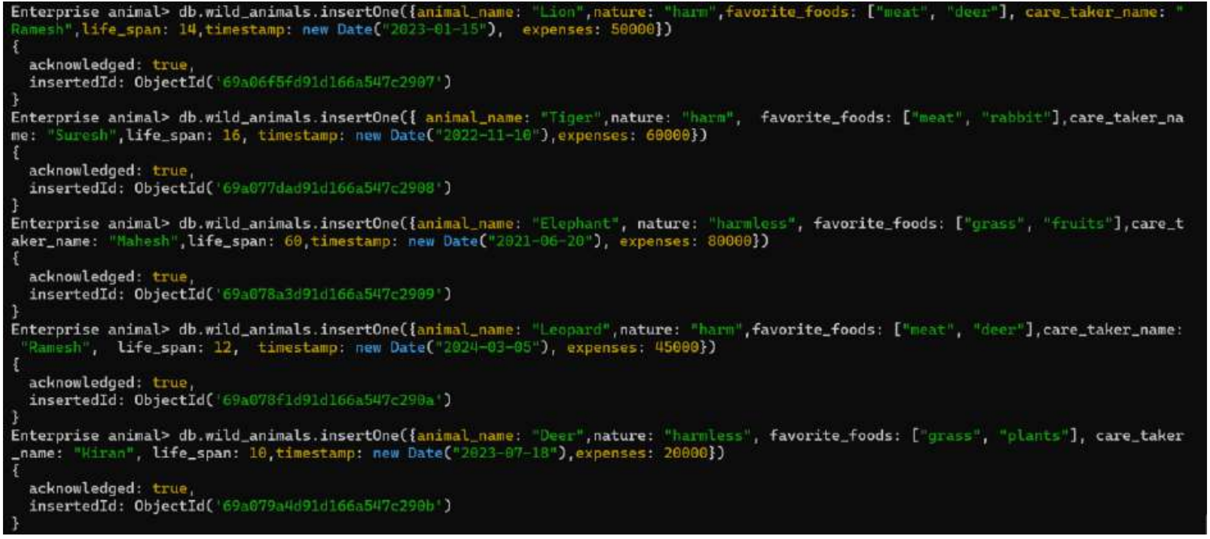
```
Enterprise animal> db.createCollection("wild_animals", {capped: true,  size: 500001,max: 102})
{ ok: 1 }
Enterprise animal> db.createCollection("domestic_animals")
{ ok: 1 }
Enterprise animal>
```

VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech	**Date:** 26-02-2026
Faculty Name: Prof. S.Gopikrishnan	**School:** SCOPE
Student name: Jahnavi Naidu	**Reg. no.:** 23BCE8395

4. 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.



```
Enterprise animal> db.wild_animals.insertOne({animal_name: "Lion",nature: "harm",favorite_foods: ["meat", "deer"], care_taker_name: "
Ramesh",life_span: 14,timestamp: new Date("2023-01-15"),  expenses: 50000})
{
  acknowledged: true,
  insertedId: ObjectId('69a06f5fd91d166a547c2907')
}
Enterprise animal> db.wild_animals.insertOne({ animal_name: "Tiger",nature: "harm",  favorite_foods: ["meat", "rabbit"],care_taker_na
me: "Suresh",life_span: 16, timestamp: new Date("2022-11-10"),expenses: 60000})
{
  acknowledged: true,
  insertedId: ObjectId('69a077dad91d166a547c2908')
}
Enterprise animal> db.wild_animals.insertOne({animal_name: "Elephant", nature: "harmless", favorite_foods: ["grass", "fruits"],care_t
aker_name: "Mahesh",life_span: 60,timestamp: new Date("2021-06-20"), expenses: 80000})
{
  acknowledged: true,
  insertedId: ObjectId('69a078a3d91d166a547c2909')
}
Enterprise animal> db.wild_animals.insertOne({animal_name: "Leopard",nature: "harm",favorite_foods: ["meat", "deer"],care_taker_name:
 "Ramesh",  life_span: 12,  timestamp: new Date("2024-03-05"), expenses: 45000})
{
  acknowledged: true,
  insertedId: ObjectId('69a078f1d91d166a547c290a')
}
Enterprise animal> db.wild_animals.insertOne({animal_name: "Deer",nature: "harmless", favorite_foods: ["grass", "plants"], care_taker
_name: "Kiran", life_span: 10,timestamp: new Date("2023-07-18"),expenses: 20000})
{
  acknowledged: true,
  insertedId: ObjectId('69a079a4d91d166a547c290b')
}
```

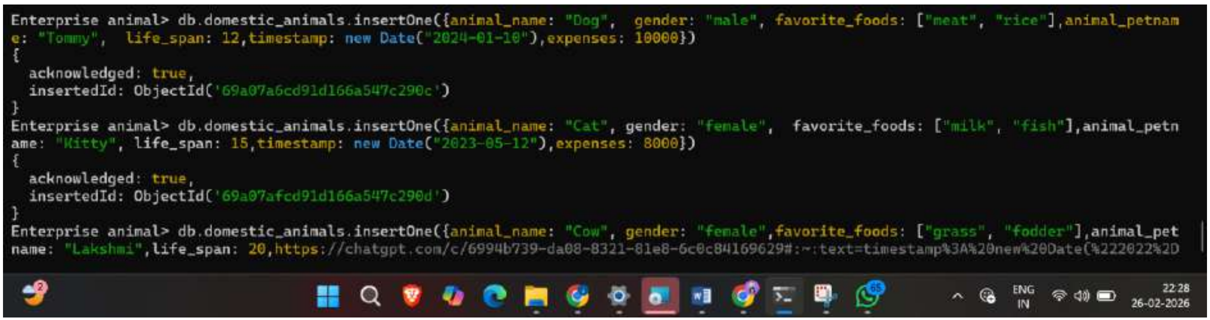
5. Add 5 domestic-animal details to the collection named 'domestic_animals'. Each document consists of following fields as animal_name, gender (male or female), favorite_foods (meat, rabbits, deer etc) as array, animal_petname, life span (in years), timestamp (when the animal registered at the Zoo) and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

VIT-AP UNIVERSITY, ANDHRA PRADESH

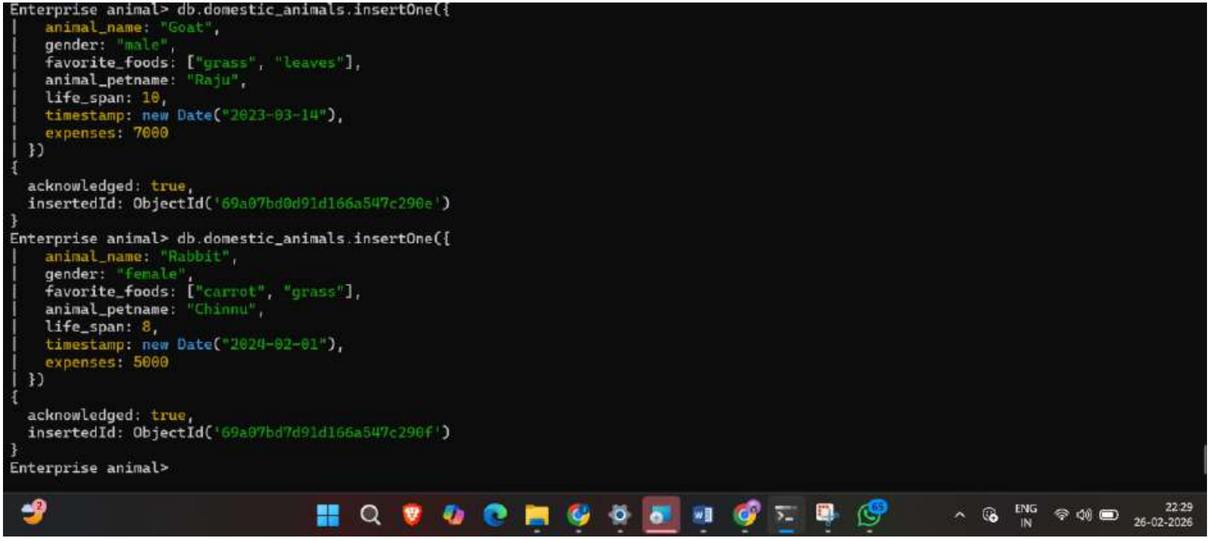
Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395

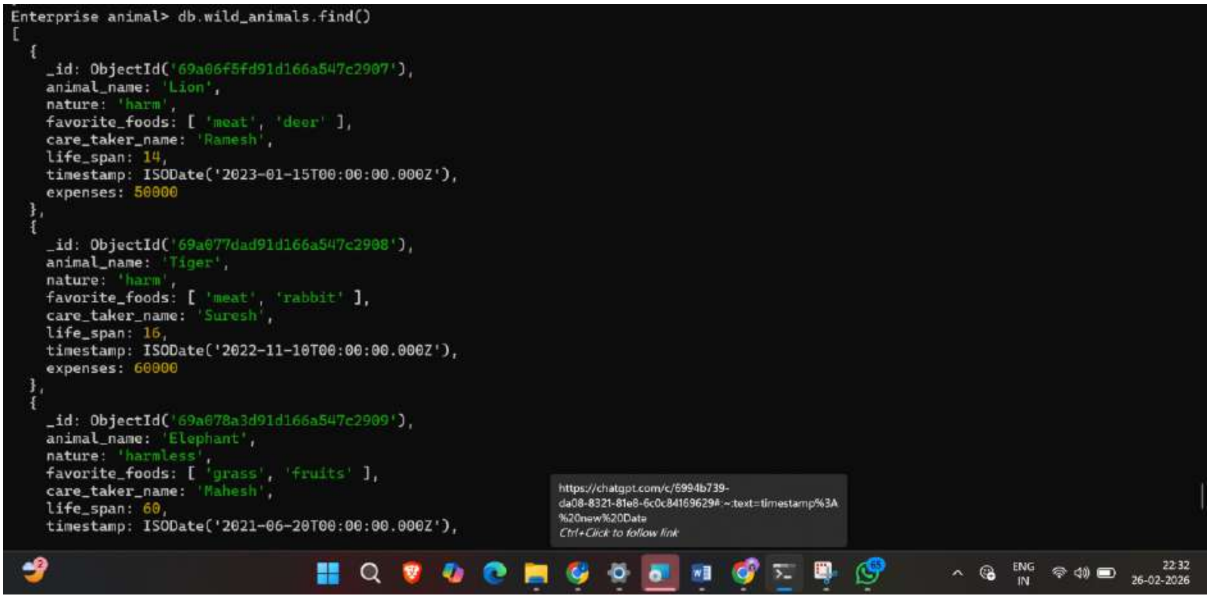


```
Enterprise animal> db.domestic_animals.insertOne({animal_name: "Dog",  gender: "male", favorite_foods: ["meat", "rice"],animal_petnam
e: "Tommy",  life_span: 12,timestamp: new Date("2024-01-10"),expenses: 10000})
{
  acknowledged: true,
  insertedId: ObjectId('69a07a6cd91d166a547c290c')
}
Enterprise animal> db.domestic_animals.insertOne({animal_name: "Cat", gender: "female",  favorite_foods: ["milk", "fish"],animal_petn
ame: "Kitty", life_span: 15,timestamp: new Date("2023-05-12"),expenses: 8000})
{
  acknowledged: true,
  insertedId: ObjectId('69a07afcd91d166a547c290d')
}
Enterprise animal> db.domestic_animals.insertOne({animal_name: "Cow", gender: "female",favorite_foods: ["grass", "fodder"],animal_pet
name: "Lakshmi",life_span: 20,https://chatgpt.com/c/6994b739-da08-8321-81e8-6c0c84169629#:~:text=timestamp%3A%20new%20Date(%222022%2D
```



```
Enterprise animal> db.domestic_animals.insertOne({
|   animal_name: "Goat",
|   gender: "male",
|   favorite_foods: ["grass", "leaves"],
|   animal_petname: "Raju",
|   life_span: 10,
|   timestamp: new Date("2023-03-14"),
|   expenses: 7000
| })
{
  acknowledged: true,
  insertedId: ObjectId('69a07bd0d91d166a547c290e')
}
Enterprise animal> db.domestic_animals.insertOne({
|   animal_name: "Rabbit",
|   gender: "female",
|   favorite_foods: ["carrot", "grass"],
|   animal_petname: "Chinnu",
|   life_span: 8,
|   timestamp: new Date("2024-02-01"),
|   expenses: 5000
| })
{
  acknowledged: true,
  insertedId: ObjectId('69a07bd7d91d166a547c290f')
}
Enterprise animal>
```

6. Write a MongoDB query to display all documents available in wild_animals and domastic_animals.



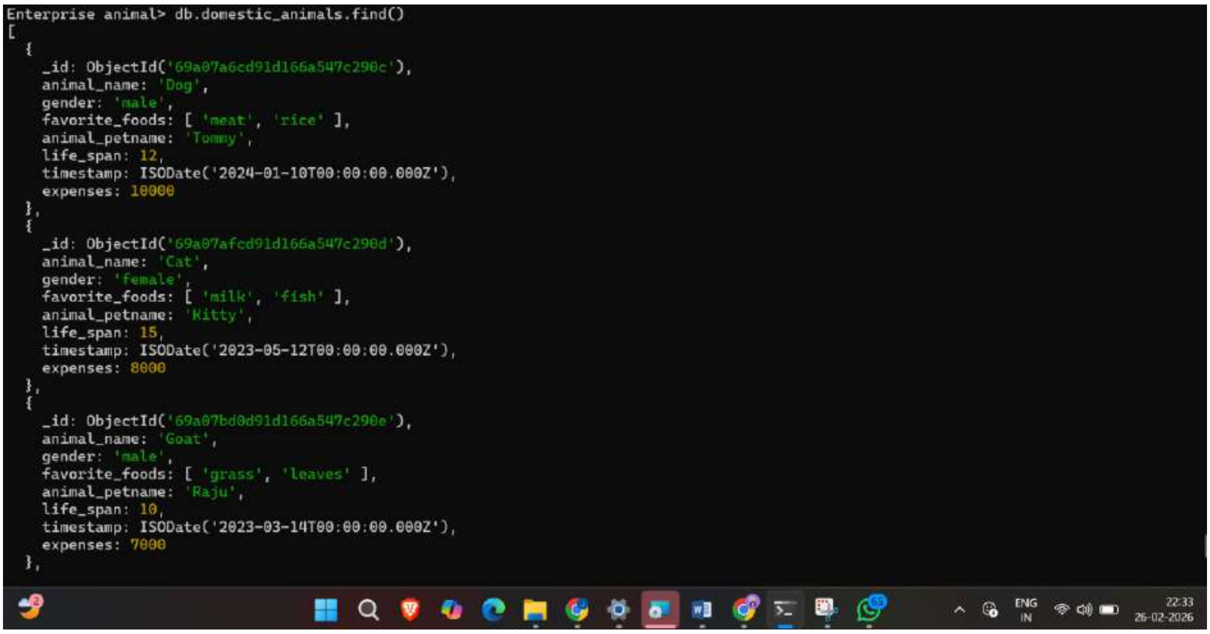
```
Enterprise animal> db.wild_animals.find()
[
  {
    _id: ObjectId('69a06f5fd91d166a547c2907'),
    animal_name: 'Lion',
    nature: 'harm',
    favorite_foods: [ 'meat', 'deer' ],
    care_taker_name: 'Ramesh',
    life_span: 14,
    timestamp: ISODate('2023-01-15T00:00:00.000Z'),
    expenses: 50000
  },
  {
    _id: ObjectId('69a077dad91d166a547c2908'),
    animal_name: 'Tiger',
    nature: 'harm',
    favorite_foods: [ 'meat', 'rabbit' ],
    care_taker_name: 'Suresh',
    life_span: 16,
    timestamp: ISODate('2022-11-10T00:00:00.000Z'),
    expenses: 60000
  },
  {
    _id: ObjectId('69a078a3d91d166a547c2909'),
    animal_name: 'Elephant',
    nature: 'harmless',
    favorite_foods: [ 'grass', 'fruits' ],
    care_taker_name: 'Mahesh',
    life_span: 60,
    timestamp: ISODate('2021-06-20T00:00:00.000Z'),
```

https://chatgpt.com/c/6994b739-da08-8321-81e8-6c0c84169629#:~:text=timestamp%3A%20new%20Date
Ctrl+Click to follow link

VIT-AP UNIVERSITY, ANDHRA PRADESH

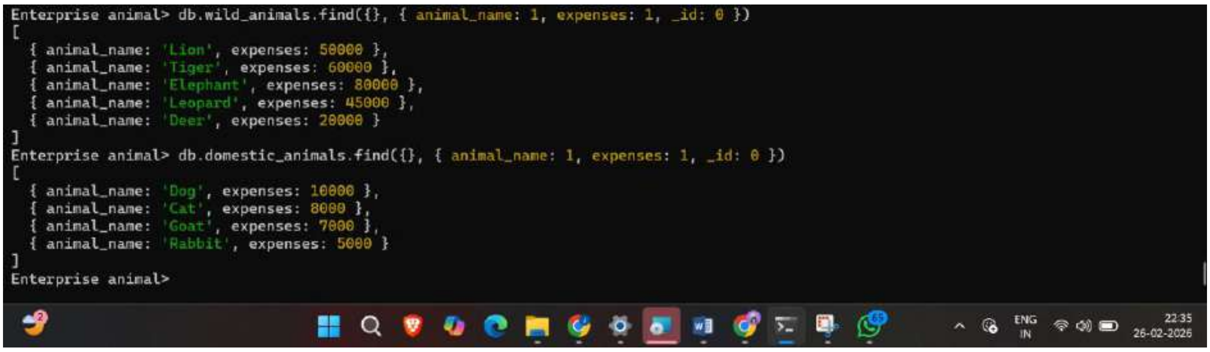
Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech **Date:** 26-02-2026
Faculty Name: Prof. S.Gopikrishnan **School:** SCOPE
Student name: Jahnavi Naidu **Reg. no.:** 23BCE8395



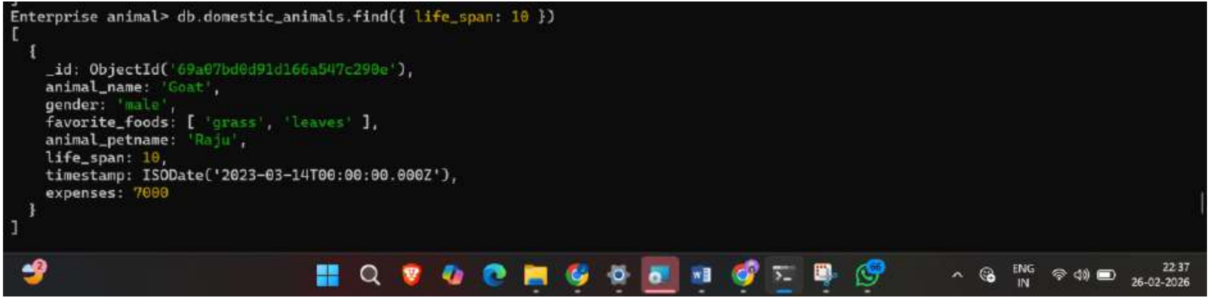
```
Enterprise animal> db.domestic_animals.find()
[
  {
    _id: ObjectId('69a07a6cd91d166a547c290c'),
    animal_name: 'Dog',
    gender: 'male',
    favorite_foods: [ 'meat', 'rice' ],
    animal_petname: 'Tommy',
    life_span: 12,
    timestamp: ISODate('2024-01-10T00:00:00.000Z'),
    expenses: 10000
  },
  {
    _id: ObjectId('69a07afcd91d166a547c290d'),
    animal_name: 'Cat',
    gender: 'female',
    favorite_foods: [ 'milk', 'fish' ],
    animal_petname: 'Kitty',
    life_span: 15,
    timestamp: ISODate('2023-05-12T00:00:00.000Z'),
    expenses: 8000
  },
  {
    _id: ObjectId('69a07bd0d91d166a547c290e'),
    animal_name: 'Goat',
    gender: 'male',
    favorite_foods: [ 'grass', 'leaves' ],
    animal_petname: 'Raju',
    life_span: 10,
    timestamp: ISODate('2023-03-14T00:00:00.000Z'),
    expenses: 7000
  },
```

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



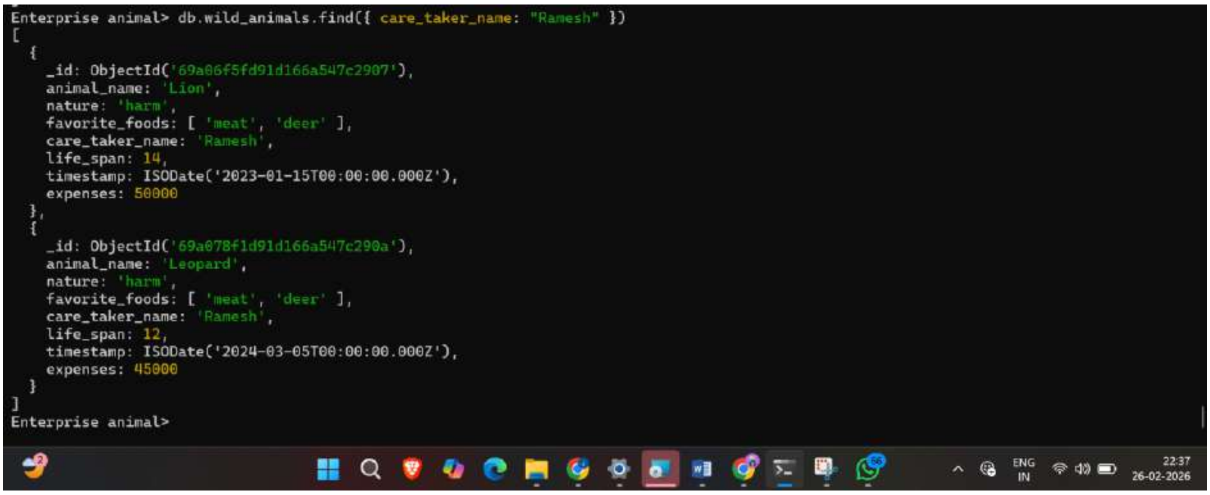
```
Enterprise animal> db.wild_animals.find({}, { animal_name: 1, expenses: 1, _id: 0 })
[
  { animal_name: 'Lion', expenses: 50000 },
  { animal_name: 'Tiger', expenses: 60000 },
  { animal_name: 'Elephant', expenses: 80000 },
  { animal_name: 'Leopard', expenses: 45000 },
  { animal_name: 'Deer', expenses: 20000 }
]
Enterprise animal> db.domestic_animals.find({}, { animal_name: 1, expenses: 1, _id: 0 })
[
  { animal_name: 'Dog', expenses: 10000 },
  { animal_name: 'Cat', expenses: 8000 },
  { animal_name: 'Goat', expenses: 7000 },
  { animal_name: 'Rabbit', expenses: 5000 }
]
Enterprise animal>
```

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



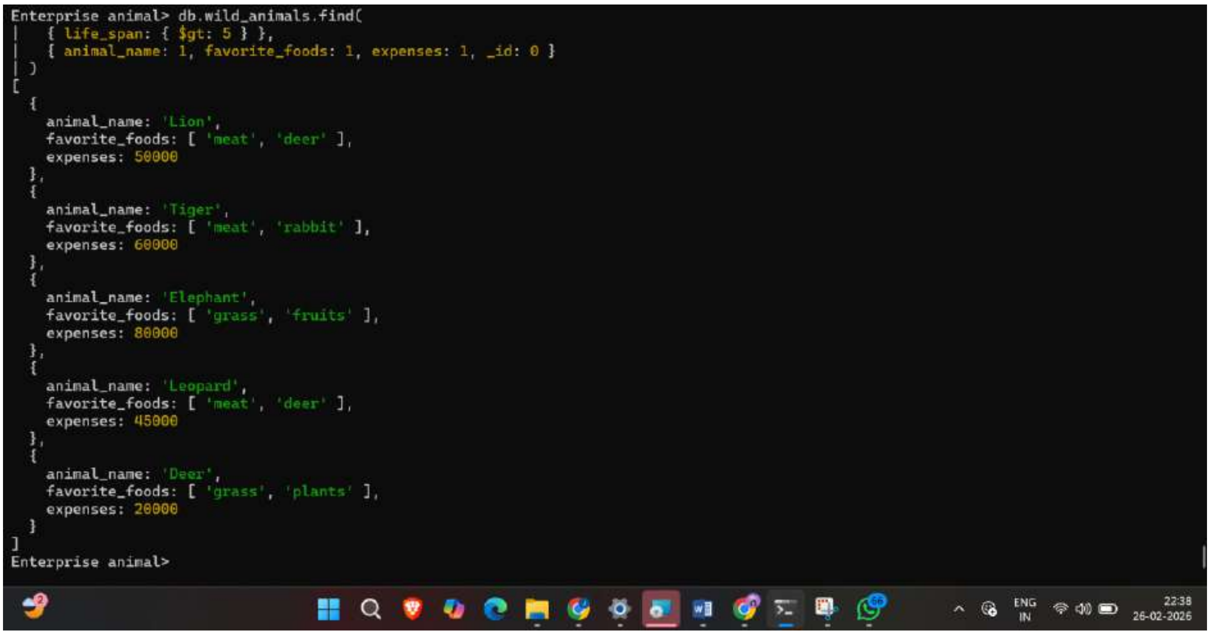
```
Enterprise animal> db.domestic_animals.find({ life_span: 10 })
[
  {
    _id: ObjectId('69a07bd0d91d166a547c290e'),
    animal_name: 'Goat',
    gender: 'male',
    favorite_foods: [ 'grass', 'leaves' ],
    animal_petname: 'Raju',
    life_span: 10,
    timestamp: ISODate('2023-03-14T00:00:00.000Z'),
    expenses: 7000
  }
]
```

9. Write a MongoDB query to display wild_animals available under a particular care_taker.



```
Enterprise animal> db.wild_animals.find({ care_taker_name: "Ramesh" })
[
  {
    _id: ObjectId('69a06f5fd91d166a547c2907'),
    animal_name: 'Lion',
    nature: 'harm',
    favorite_foods: [ 'meat', 'deer' ],
    care_taker_name: 'Ramesh',
    life_span: 14,
    timestamp: ISODate('2023-01-15T00:00:00.000Z'),
    expenses: 50000
  },
  {
    _id: ObjectId('69a078f1d91d166a547c290a'),
    animal_name: 'Leopard',
    nature: 'harm',
    favorite_foods: [ 'meat', 'deer' ],
    care_taker_name: 'Ramesh',
    life_span: 12,
    timestamp: ISODate('2024-03-05T00:00:00.000Z'),
    expenses: 45000
  }
]
Enterprise animal>
```

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



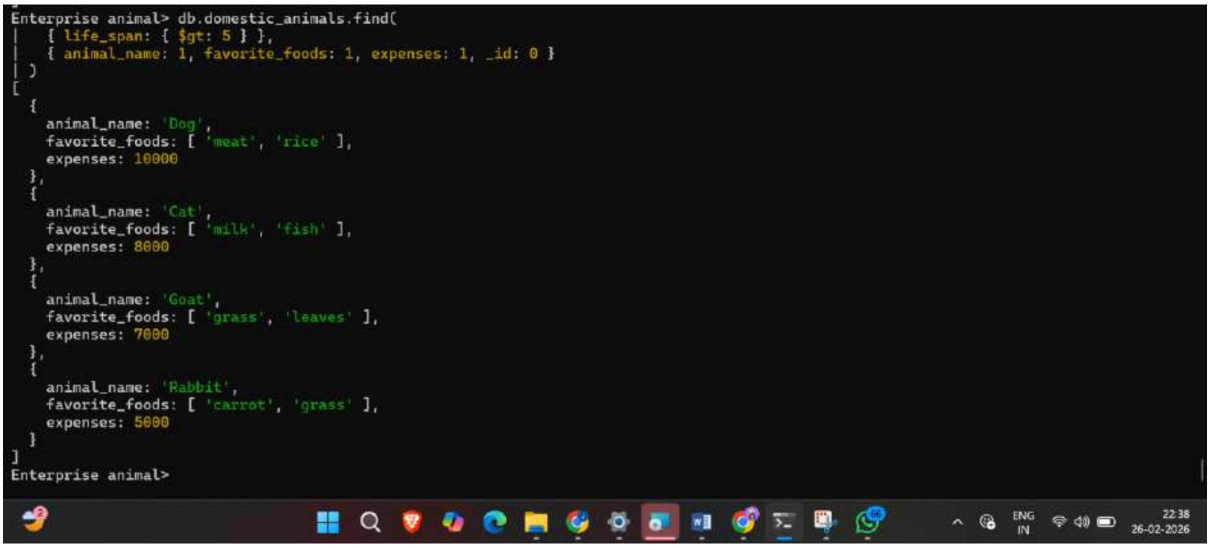
```
Enterprise animal> db.wild_animals.find(
|   { life_span: { $gt: 5 } },
|   { animal_name: 1, favorite_foods: 1, expenses: 1, _id: 0 }
| )
[
  {
    animal_name: 'Lion',
    favorite_foods: [ 'meat', 'deer' ],
    expenses: 50000
  },
  {
    animal_name: 'Tiger',
    favorite_foods: [ 'meat', 'rabbit' ],
    expenses: 60000
  },
  {
    animal_name: 'Elephant',
    favorite_foods: [ 'grass', 'fruits' ],
    expenses: 80000
  },
  {
    animal_name: 'Leopard',
    favorite_foods: [ 'meat', 'deer' ],
    expenses: 45000
  },
  {
    animal_name: 'Deer',
    favorite_foods: [ 'grass', 'plants' ],
    expenses: 20000
  }
]
Enterprise animal>
```

VIT-AP UNIVERSITY, ANDHRA PRADESH

Lab Sheet 6: MongoDB Basic commands

Branch/ Class: B.Tech/M.Tech
Faculty Name: Prof. S.Gopikrishnan
Student name: Jahnavi Naidu

Date: 26-02-2026
School: SCOPE
Reg. no.: 23BCE8395



```
Enterprise animal> db.domestic_animals.find(
|   { life_span: { $gt: 5 } },
|   { animal_name: 1, favorite_foods: 1, expenses: 1, _id: 0 }
| )
[
  {
    animal_name: 'Dog',
    favorite_foods: [ 'meat', 'rice' ],
    expenses: 10000
  },
  {
    animal_name: 'Cat',
    favorite_foods: [ 'milk', 'fish' ],
    expenses: 8000
  },
  {
    animal_name: 'Goat',
    favorite_foods: [ 'grass', 'leaves' ],
    expenses: 7000
  },
  {
    animal_name: 'Rabbit',
    favorite_foods: [ 'carrot', 'grass' ],
    expenses: 5000
  }
]
Enterprise animal>
```