Task Databases Part 3

Check status mongodb service:

```
root@docker:~# systemctl status mongodb
● mongodb.service - An object/document-oriented database
   Loaded: loaded (/lib/systemd/system/mongodb.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-10-23 15:56:58 UTC; 39s ago
     Docs: man:mongod(1)
 Main PID: 26244 (mongod)
    Tasks: 23 (limit: 528)
   CGroup: /system.slice/mongodb.service
           └─26244 /usr/bin/mongod --unixSocketPrefix=/run/mongodb --config /etc/mongodb.conf

Oct 23 15:56:58 docker systemd[1]: Started An object/document-oriented database.
root@docker:~# mongo --eval 'db.runCommand({ connectionStatus: 1 })'
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.6.3
{
        "authInfo" : {
                "authenticatedUsers" : [ ],
                "authenticatedUserRoles" : [ ]
        },
        "ok" : 1
}
```

Mongo – command in Linux for editing in Mongodb console.

Use cars – command for editing database cars (if there is no database cars – mongodb create it).

After editing database we create collection:

db.createCollection(“manufacturer”):

```
> use cars
switched to db cars
> db.createCollection("manufacturer")
{ "ok" : 1 }
> show dbs
admin   0.000GB
cars    0.000GB
config  0.000GB
local   0.000GB
> show collections
manufacturer
> db.manufacturer.insertOne({"name": "Ford"})
{
        "acknowledged" : true,
        "insertedId" : ObjectId("63556b9da6c29b2575e06bd3")
}
> db.manufacturer.insertOne({"name": "Mersedes"})
{
        "acknowledged" : true,
        "insertedId" : ObjectId("63556ba8a6c29b2575e06bd4")
}
> db.manufacturer.insertOne({"name": "BMW"})
{
        "acknowledged" : true,
        "insertedId" : ObjectId("63556bafa6c29b2575e06bd5")
}
> db.manufacturer.insertMany([{"name": "Skoda"}, {"name": "Renault"}, {"name":"Peugeot"}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("63556c69a6c29b2575e06bd6"),
                ObjectId("63556c69a6c29b2575e06bd7"),
                ObjectId("63556c69a6c29b2575e06bd8")
        ]
}
```

```
> db.createCollection("models")
{ "ok" : 1 }
> db.models.insertMany([{"name": "Superb", manufacturer:"Skoda"}, {"name": "Fabia", manufacturer:"Skoda"}, {"name":"Octavia", manufacturer:"Skoda"}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("63556e06a6c29b2575e06bd9"),
                ObjectId("63556e06a6c29b2575e06bda"),
                ObjectId("63556e06a6c29b2575e06bdb")
        ]
}
```

- Insert data to document using function insertMany()

```
> db.models.insertMany([{"name": "Mustang", manufacturer:"Ford"}, {"name": "Focus", manufacturer:"Ford"}, {"name":"Transit", manufacturer:"Ford"}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("63556f09a6c29b2575e06bdc"),
                ObjectId("63556f09a6c29b2575e06bdd"),
                ObjectId("63556f09a6c29b2575e06bde")
        ]
}
```

Find some information from documents in collection with function Find(), sort(), limit(), skip():

```
>
> db.models.find({manufacturer:"Ford"})
{ "_id" : ObjectId("63556f09a6c29b2575e06bdc"), "name" : "Mustang", "manufacturer" : "Ford" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bdd"), "name" : "Focus", "manufacturer" : "Ford" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bde"), "name" : "Transit", "manufacturer" : "Ford" }
>
>
>
>
>
> db.models.find({manufacturer:"Ford", name:"Mustang"})
{ "_id" : ObjectId("63556f09a6c29b2575e06bdc"), "name" : "Mustang", "manufacturer" : "Ford" }
>
```

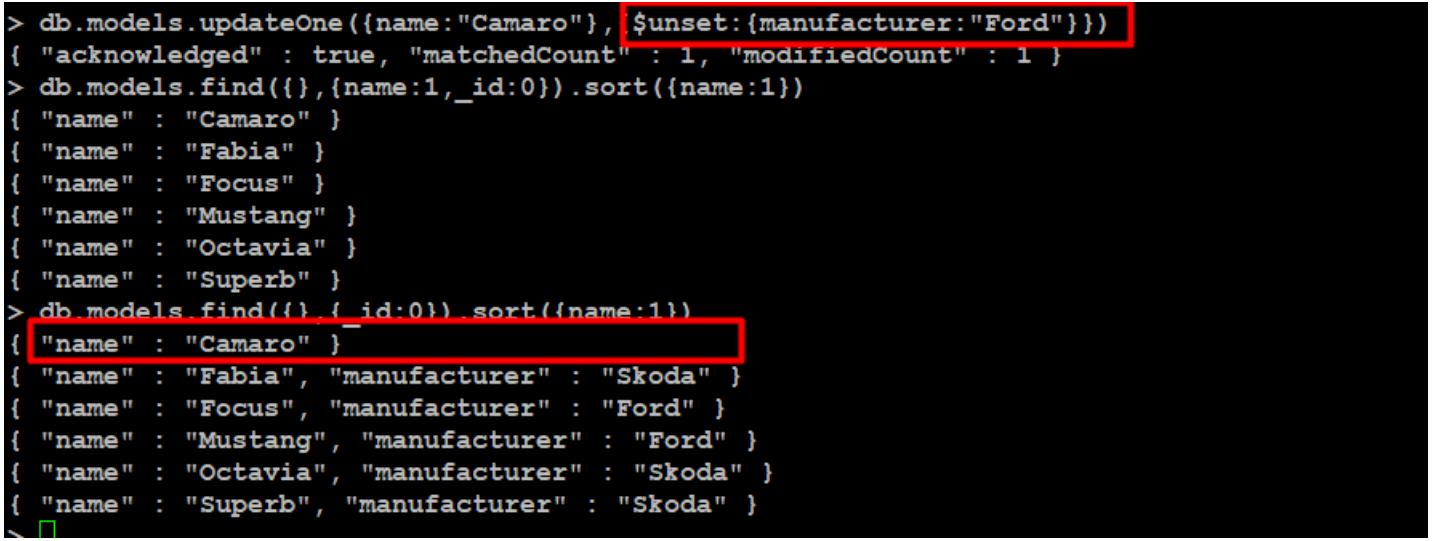
```
> db.models.find({manufacturer:"Ford"})
{ "_id" : ObjectId("63556f09a6c29b2575e06bdc"), "name" : "Mustang", "manufacturer" : "Ford" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bdd"), "name" : "Focus", "manufacturer" : "Ford" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bde"), "name" : "Transit", "manufacturer" : "Ford" }
>
>
>
> db.models.find({manufacturer:"Ford"},{name:1})
{ "_id" : ObjectId("63556f09a6c29b2575e06bdc"), "name" : "Mustang" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bdd"), "name" : "Focus" }
{ "_id" : ObjectId("63556f09a6c29b2575e06bde"), "name" : "Transit" }
>
```

```
> db.models.find({},{name:1,_id:0}).sort({name:1})
{ "name" : "Fabia" }
{ "name" : "Focus" }
{ "name" : "Mustang" }
{ "name" : "Octavia" }
{ "name" : "Superb" }
{ "name" : "Transit" }
```

```
>
> db.models.find({},{name:1,_id:0}).sort({name:1})
{ "name" : "Fabia" }
{ "name" : "Focus" }
{ "name" : "Mustang" }
{ "name" : "Octavia" }
{ "name" : "Superb" }
{ "name" : "Transit" }
>
>
>
>
> db.models.find({},{name:1,_id:0}).sort({name:1}).limit(3)
{ "name" : "Fabia" }
{ "name" : "Focus" }
{ "name" : "Mustang" }
>
>
> db.models.find({},{name:1,_id:0}).sort({name:1}).skip(3).limit(3)
{ "name" : "Octavia" }
{ "name" : "Superb" }
{ "name" : "Transit" }
```

- Update documents using function updateOne() with operators $set and $unset:

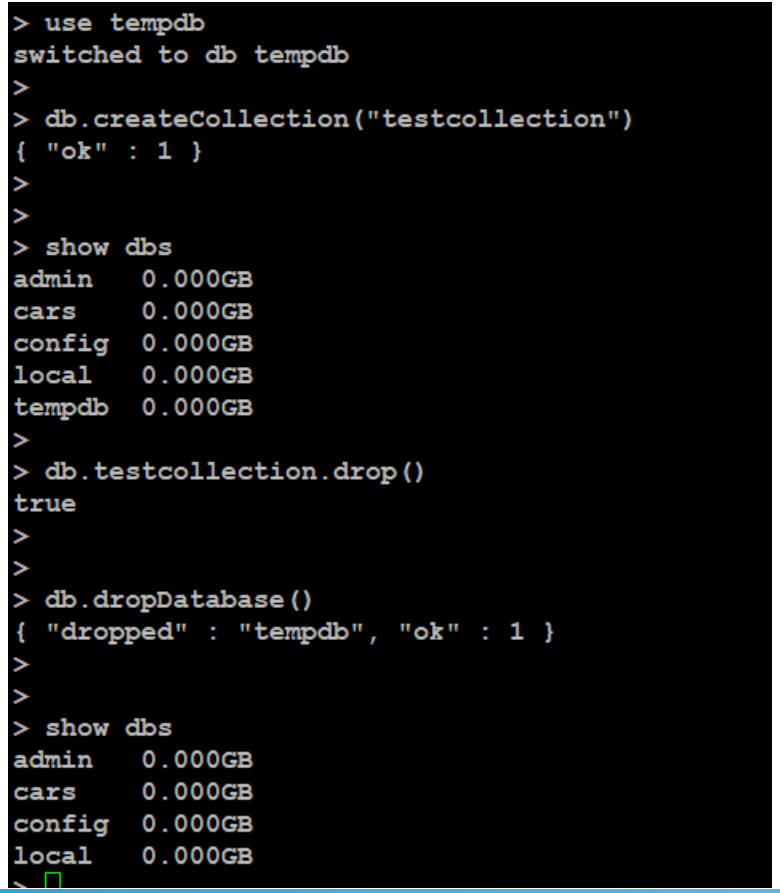
```
> db.models.updateOne({name:"Mondeo"},{$set:{name: "Camaro"}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
>
>
>
>
> db.models.find({},{name:1,_id:0}).sort({name:1})
{ "name" : "Camaro" }
{ "name" : "Fabia" }
{ "name" : "Focus" }
{ "name" : "Mustang" }
{ "name" : "Octavia" }
{ "name" : "Superb" }
```


> db.models.updateOne({name:"Camaro"},{$unset:{manufacturer:"Ford"}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.models.find({},{name:1,_id:0}).sort({name:1})
{ "name" : "Camaro" }
{ "name" : "Fabia" }
{ "name" : "Focus" }
{ "name" : "Mustang" }
{ "name" : "Octavia" }
{ "name" : "Superb" }
> db.models.find({},{_id:0}).sort({name:1})
{ "name" : "Camaro" }
{ "name" : "Fabia", "manufacturer" : "Skoda" }
{ "name" : "Focus", "manufacturer" : "Ford" }
{ "name" : "Mustang", "manufacturer" : "Ford" }
{ "name" : "Octavia", "manufacturer" : "Skoda" }
{ "name" : "Superb", "manufacturer" : "Skoda" }

- Delete document using function deleteOne():

```
> db.models.deleteOne({name:"Camaro"})
{ "acknowledged" : true, "deletedCount" : 1 }
>
>
> db.models.find({},{_id:0}).sort({name:1})
{ "name" : "Fabia", "manufacturer" : "Skoda" }
{ "name" : "Focus", "manufacturer" : "Ford" }
{ "name" : "Mustang", "manufacturer" : "Ford" }
{ "name" : "Octavia", "manufacturer" : "Skoda" }
{ "name" : "Superb", "manufacturer" : "Skoda" }
```

- Create database, create collection and after delete collection using function drop() and drop database using function dropDatabase():


> use tempdb
switched to db tempdb
>
> db.createCollection("testcollection")
{ "ok" : 1 }
>
>
> show dbs
admin   0.000GB
cars   0.000GB
config 0.000GB
local   0.000GB
tempdb 0.000GB
>
> db.testcollection.drop()
true
>
>
> db.dropDatabase()
{ "dropped" : "tempdb", "ok" : 1 }
>
>
> show dbs
admin   0.000GB
cars   0.000GB
config 0.000GB
local   0.000GB