

Date: 28.04.2025

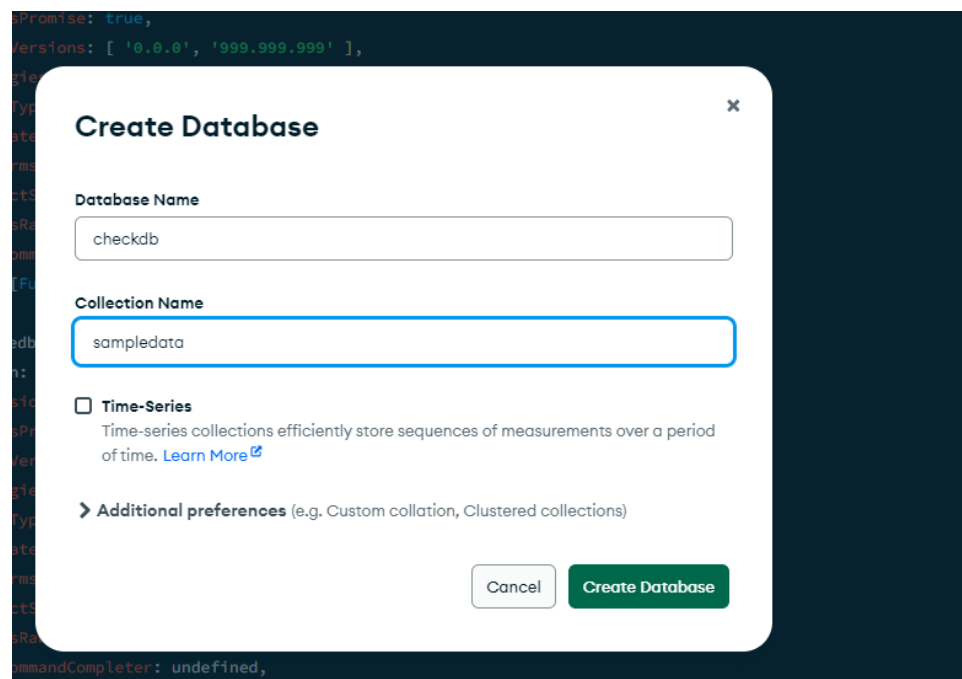
Database drop, delete collections, update documents

create database as sampledbs, create collection sample

```
> db.dropDatabase()  
< { ok: 1, dropped: 'sampledb' }  
sampledb> |
```

drop database

create database checkdb and create collection as sampledata



The screenshot shows a 'Create Database' dialog box with the following fields and options:

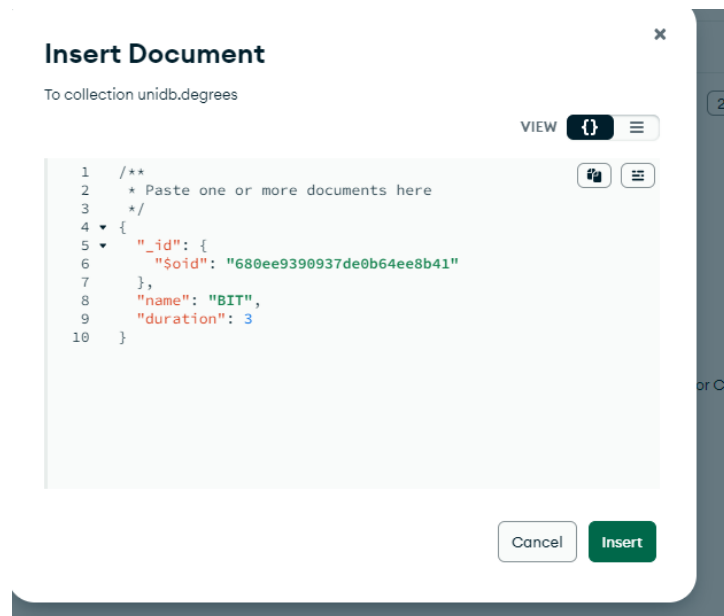
- Database Name:** checkdb
- Collection Name:** sampledata
- ☐ **Time-Series**
Time-series collections efficiently store sequences of measurements over a period of time. [Learn More](#)
- > Additional preferences** (e.g. Custom collation, Clustered collections)
- Buttons:** Cancel, Create Database

Drop collection

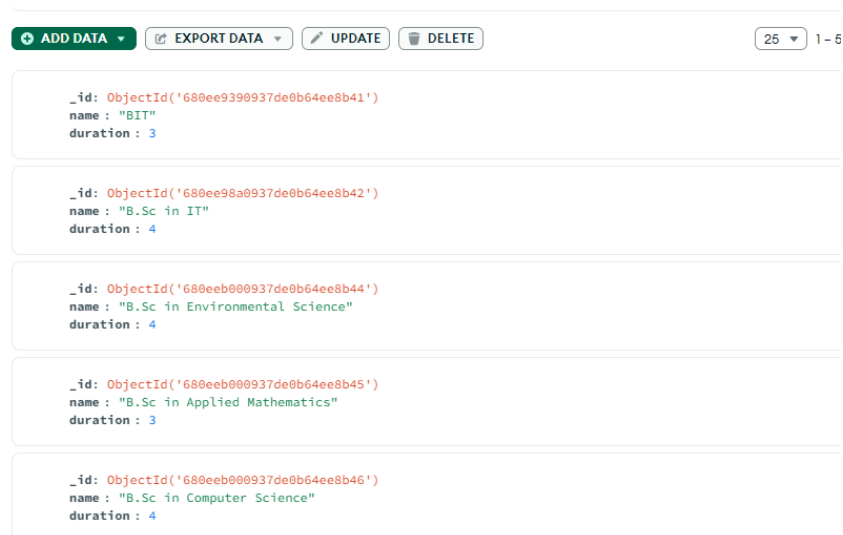
```
> db.sampledata.drop()  
< true
```

In undb database create collection as degrees.

And insert documents



Insert multiple documents



Delete particular object id

```
db.degrees.deleteOne({_id:ObjectId('680ee9390937de0b64ee8b41')})
```

```
> db.degrees.deleteOne({_id:ObjectId('680ee9390937de0b64ee8b41')})
< {
  acknowledged: true,
  deletedCount: 1
}
unidb> |
```

Check whether its deleted

```
db.degrees.find().projection(['name','duration'])
```

```
}
> db.degrees.find().projection(['name','duration'])
< {
  _id: ObjectId('680ee98a0937de0b64ee8b42'),
  name: 'B.Sc in IT',
  duration: 4
}
{
  _id: ObjectId('680eeb000937de0b64ee8b44'),
  name: 'B.Sc in Environmental Science',
  duration: 4
}
{
  _id: ObjectId('680eeb000937de0b64ee8b45'),
  name: 'B.Sc in Applied Mathematics',
  duration: 3
}
{
  _id: ObjectId('680eeb000937de0b64ee8b46'),
  name: 'B.Sc in Computer Science',
  duration: 4
}
```

Select certain data (name only)

```
db.degrees.find({}, {name:1, _id:0}).pretty()
```

```
> db.degrees.find({}, {name:1, _id:0}).pretty()
< {
  name: 'B.Sc in IT'
}
{
  name: 'B.Sc in Environmental Science'
}
{
  name: 'B.Sc in Applied Mathematics'
}
{
  name: 'B.Sc in Computer Science'
}
```

```
db.degrees.deleteMany({duration:{$lt:4}})
```

```
> db.degrees.find({}, {name:1, _id:0}).pretty()
< {
  name: 'B.Sc in IT'
}
{
  name: 'B.Sc in Environmental Science'
}
{
  name: 'B.Sc in Computer Science'
}
```

Type a query: { field: 'value' } or [Generate query](#)

Explain

Reset

Find

Options

+ ADD DATA

EXPORT DATA

UPDATE

DELETE

251 - 4 of 4

<>⌵☰⌲⌳⌴

_id: ObjectId('680ee98a9937de0b64ee8b42')
name: "B.Sc in IT"
duration: 4

_id: ObjectId('680eeb009937de0b64ee8b44')
name: "B.Sc in Environmental Science"
duration: 4

_id: ObjectId('680eeb009937de0b64ee8b46')
name: "B.Sc in Computer Science"
duration: 4

▶_id: ObjectId('680eef370937de0b64ee8b53')
name: "fsgt"
duration: 4

Edit document

↶↷📄🗑️

```
1 _id: ObjectId('680eef370937de0b64ee8b53')
2 name: 'fbb'
duration: 0
```

ObjectId
String
Int32

Document modified.

CANCEL UPDATE

Update contain 2 operation, what document we want to update, what the value we want to update/set

```
db.degrees.updateOne({_id:ObjectId('680eef370937de0b64ee8b53')},{ $set:{name:'BIT',duration:3}})
```

```
> db.degrees.updateOne({_id:ObjectId('680eef370937de0b64ee8b53')},{ $set:{name:'BIT',duration:3}})
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Find update data

```
> db.degrees.findOne({_id:ObjectId('680eef370937de0b64ee8b53')})
< {
  _id: ObjectId('680eef370937de0b64ee8b53'),
  name: 'BIT',
  duration: 3
}
unidb>
```

Increase duration by one which duration is greater than 4

```
db.degrees.updateMany({duration:{$gte:4}},{ $inc:{duration:1}})
```

```
> db.degrees.updateMany({duration:{$gte:4}},{ $inc:{duration:1}})
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
```

Add new certain skill to the students

Push – add elements only

```
db.students.updateOne({regno:'2021IT001'},{$push:{skills:'PHP'}})
```

```
> db.students.updateOne({regno:'2021IT001'},{$push:{skills:'PHP'}})
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
unmdb>
```

```
db.students.findOne({regno:'2021IT001'})
```

```
> db.students.findOne({regno:'2021IT001'})
< {
  _id: ObjectId('6809be213a66cfb0140e2c8c'),
  regno: '2021IT001',
  name: 'Kolin',
  age: 25,
  degree: 'IT',
  gpa: 3.2,
  skills: [
    'Js',
    'Mongodb',
    'Node',
    'PHP'
  ]
}
```