

<https://www.halvorsen.blog>



# MongoDB Cloud and Python

Hans-Petter Halvorsen

# Contents

- MongoDB
  - MongoDB Atlas – The Cloud version of MongoDB
  - MongoDB Compass
- Python and PyMongo - Python Driver/Library for MongoDB
- Basic Python Examples
  - Write Data
  - Read Data
- Datalogging Example
- Data Monitoring Example



# MongoDB

# MongoDB

- MongoDB is a cross-platform document-oriented database program.
- MongoDB is a NoSQL database program
- MongoDB uses JSON-like documents
- Home Page: <https://www.mongodb.com/>

## Software:

- **MongoDB Community Server** – Free version of the MongoDB Server which can be installed locally on your computer or a server
- MongoDB Atlas – Premade MongoDB ready to use in the Cloud
- **MongoDB Compass** – GUI for connecting to and manipulating your MongoDB database
- **PyMongo** – MongoDB Driver for Python

# Python and MongoDB

- I have already made a Tutorial and Video using MongoDB Community Server, MongoDB Compass, and Python Examples using PyMongo

Python and MongoDB:

- <https://youtu.be/ilbgHPuKWQI>
- <https://www.halvorsen.blog/documents/programming/python>

=> This Tutorial focuses on **MongoDB Atlas** - The Cloud version of MongoDB



# MongoDB Atlas

# MongoDB Atlas

- Premade MongoDB ready to use in the Cloud
- You can use a Shared Clusters for free
- Purpose: Learning MongoDB or developing small applications

<https://www.mongodb.com/cloud/atlas>

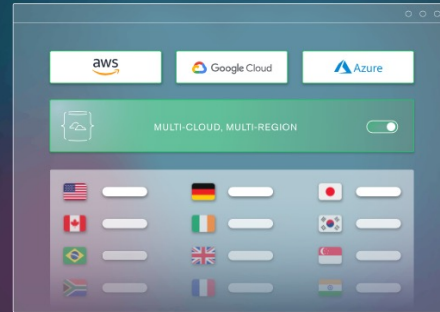
# MongoDB Atlas

[Products](#)[Solutions](#)[Resources](#)[Company](#)[Pricing](#)[Sign In](#)[Try Free](#)

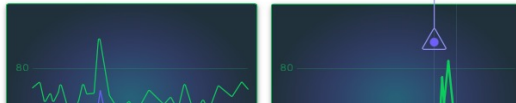
MONGODB ATLAS

## Global multi-cloud database

The most advanced cloud database service on the market, with unmatched data distribution and mobility across AWS, Azure, and Google Cloud, built-in automation for resource and workload optimization, and so much more.

[Get started now](#)

AUTO-SCALE



## Intelligent performance optimization

MongoDB Atlas offers a hands-free approach to performance optimization with auto-scaling, actionable recommendations tailored to your specific workloads, and more, so you can easily adapt to changing requirements.





# Create Database

Hans-Petter's Org - ... Access Manager Billing All Clusters Get Help Hans-Petter

Project 0 Atlas Realm Charts

DEPLOYMENT

Databases

Explore Data NEW

Triggers

Data Lake

SECURITY

Database Access


Network Access

Advanced

HANS-PETTER'S ORG - 2021-10-13 > PROJECT 0

## Database Deployments

Find a database deployment...




### Create a database

Choose your cloud provider, region, and specs.

[Build a Database](#)

Once your database is up and running, live migrate an existing MongoDB database into Atlas with our [Live Migration Service](#).

FREE

 Shared

For learning and exploring MongoDB in a cloud environment. Basic configuration options.

- ✓ No credit card required to start
- ✓ Explore with sample datasets
- ✓ Upgrade to dedicated clusters for full functionality

[Create](#)

Starting at **FREE**

System Status: All Good

©2021 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Get Started

# Create Cluster

CLUSTERS > CREATE A SHARED CLUSTER

## Create a Shared Cluster

Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our [documentation](#).

PREVIEW Serverless

Dedicated

FREE Shared

For learning and exploring MongoDB in a sandbox environment. Basic configuration controls.

No credit card required to start. Upgrade to dedicated clusters for full functionality. Explore with sample datasets. Limit of one free cluster per project.

### Cloud Provider & Region

AWS, N. Virginia (us-east-1) ▾



★ Recommended region ⓘ

NORTH AMERICA	EUROPE	ASIA
N. Virginia (us-east-1) ★	Ireland (eu-west-1) ★	Singapore (ap-southeast-1) ★
Oregon (us-west-2) ★	Frankfurt (eu-central-1) ★	Mumbai (ap-south-1)
AUSTRALIA		
Sydney (ap-southeast-2) ★		

FREE

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

[Back](#)

Create Cluster

# Network Access

- DEPLOYMENT
  - Databases
  - Triggers
  - Data Lake
- SECURITY
  - Database Access
  - Network Access**
  - Advanced

HPH > MEASUREMENTSYSTEM

## Network Access

IP Access List Peering Private Endpoint

+ ADD IP ADDRESS

You will only be able to connect to your cluster from the following list of IP Addresses:

IP Address	Comment	Status	Actions
██████████ (includes your current IP address)		● Active	<a href="#">EDIT</a> <a href="#">DELETE</a>
██████████		● Active	<a href="#">EDIT</a> <a href="#">DELETE</a>

System Status: All Good



# Connect to the Cluster

HPH > MEASUREMENTSYSTEM

## Database Deployments

Find a database deployment...

**MongoDBCluster** Connect View Monitoring Brow

● R 0.003  
● W 0  
Last 2 hours  
0.02/s

● Connections 11.0  
Last 2 hours  
24.0

VERSION	REGION	CLUSTER TIER
4.4.9	Azure / Netherlands (westeurope)	M0 Sandbox (Gen

System Status: **All Good**

©2021 MongoDB, Inc. [Status](#) [Terms](#) [Privacy](#) [Atlas Blog](#) [Contact Sales](#)

## Connect to MongoDBCluster

✓ **Setup connection security**

Choose a connection method

**Connect**

### Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.



#### Connect with the MongoDB Shell

Interact with your cluster using MongoDB's interactive Javascript interface



#### Connect your application

Connect your application to your cluster using MongoDB's native drivers



#### Connect using MongoDB Compass

Explore, modify, and visualize your data with MongoDB's GUI



Go Back

Close

# Browse Collections

HPH Access Manager Billing All Clusters Get Help Hans-Petter

MeasurementSystem Atlas Realm Charts

DEPLOYMENT

Databases

Triggers

Data Lake

SECURITY

Database Access

Network Access

Advanced

HPH > MEASUREMENTSYSTEM

## Database Deployments

Find a database deployment...

+ Create

MongoDBCluster Connect View Monitoring Browse Collections ...

FREE SHARED

R 0.003 W 0 Last 2 hours 0.02/s

Connections 19.0 Last 2 hours 26.0

In 50.9 B/s Out 847.3 B/s Last 2 hours 1.3 KB/s

Data Size 21.3 KB Last 44 minutes 512.0 MB

Enhance Your Experience  
For production throughput and richer metrics, upgrade to a dedicated cluster now!

Upgrade

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED REALM APP	ATLAS SEARCH
4.4.9	Azure / Netherlands (westeurope)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	<a href="#">Create Index</a>

System Status: All Good

©2021 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

# Create Database, View Data, etc

The screenshot displays the MongoDB Atlas web interface. At the top, there's a navigation bar with 'HPH' and 'Access Manager' dropdowns, and 'Billing' text. On the right, it shows 'All Clusters', 'Get Help', and a user profile 'Hans-Petter'. Below this, a secondary navigation bar includes 'MeasurementSystem', 'Atlas', 'Realm', and 'Charts'. A left sidebar contains 'DEPLOYMENT' (with 'Databases' selected) and 'SECURITY' sections. The main content area shows the 'MongoDBCluster' configuration with tabs for 'Overview', 'Real Time', 'Metrics', 'Collections', 'Search', 'Profiler', 'Performance Advisor', 'Online Archive', and 'Command Line Tools'. The 'Collections' tab is active, displaying 'Library.Book' with a filter bar and query results. The filter bar contains the text '{ field: 'value' }'. The query results show 3 documents, with the first two visible:

```
{ "_id": ObjectId("6166cec4d28b4ac432552476"), "Title": "C# Programming", "Author": "Knut Hamsun" }
```

```
{ "_id": ObjectId("6166d3be2bc8a3fad760569b"), "Title": "LabVIEW Programming", "Author": "Trond Hamsun" }
```



# MongoDB Compass

# MongoDB Compass

- MongoDB Compass is the official Graphical User Interface (GUI) for MongoDB
- With MongoDB Compass you can explore and manipulate your MongoDB data
- To use Compass, you must connect to an existing MongoDB database. You can connect to:
  - A MongoDB server that you have installed locally, or
  - A MongoDB Atlas cluster.



# MongoDB Compass

## From MongoDB Atlas

Connect to MongoDBCluster

✔ Setup connection security > ✔ Choose a connection method > Connect

I do not have MongoDB Compass

I have MongoDB Compass

### 1 Choose your version of Compass:

1.12 or later

See your Compass version in "About Compass"

### 2 Copy the connection string, then open MongoDB Compass.

mongodb+srv:// :<password>@mongodbcluster.znkt.mongodb.net/test

You will be prompted for the password for the **MongoDBUserHPH** user's (Database User) username. When entering your password, make sure that any special characters are **URL encoded**.

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

×

ss - Connect

ction Help

on

erznkzt.mongr ...

7

7

7

7

7

7

7

7

7

7

7

7

## New Connection

☆ FAVORITE

Fill in connection fields individually

Paste your connection string (SRV or Standard ⓘ)

mongodb+srv://MongoDBUserHPH:<password>@mongodb

You have unsaved changes. [\[discard\]](#)

Connect

### New to Compass and don't have a cluster?

If you don't already have a cluster, you can create one for free using [MongoDB Atlas](#).

[CREATE FREE CLUSTER](#)

### How do I find my connection string in Atlas?

If you have an Atlas cluster, go to the Cluster view. Click the 'Connect' button for the cluster to which you wish to connect.

[See example](#)

### How do I format my connection string?

[See example](#)

# MongoDB Compass

MongoDB Compass - mongodcluster.znkt.mongodb.net/MeasurementSystem.MeasurementData

Connect View Collection Help

Local

4 DBS 9 COLLECTIONS

☆ FAVORITE

HOSTS

- mongodcluster-shard-00-...
- mongodcluster-shard-00-...
- mongodcluster-shard-00-...

CLUSTER

Replica Set (atlas-i4g5c5-...

3 Nodes

EDITION

MongoDB 4.4.9 Enterprise

Filter your data

Library

MeasurementSystem

- MeasurementData
- admin
- local

+> \_MONGOSH

MeasurementSystem.Me... Documents

## MeasurementSystem.MeasurementData

DOCUMENTS 10 TOTAL SIZE 890B AVG. SIZE 89B INDEXES 1 TOTAL SIZE 36.0KB AVG. SIZE 36.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' }

ADD DATA

VIEW

Displaying documents 1 - 10 of 10

```
{ "_id": ObjectId("6166daffc52c9c655d041a82"), "MeasurementValue": 21, "MeasurementDateTime": "2021-10-13 15:11:27" }
{ "_id": ObjectId("6166db0bc52c9c655d041a83"), "MeasurementValue": 22, "MeasurementDateTime": "2021-10-13 15:11:39" }
{ "_id": ObjectId("6166db15c52c9c655d041a84"), "MeasurementValue": 21, "MeasurementDateTime": "2021-10-13 15:11:49" }
{ "_id": ObjectId("6166db1fc52c9c655d041a85"), "MeasurementValue": 20, "MeasurementDateTime": "2021-10-13 15:11:59" }
```



# Python

Hans-Petter Halvorsen

[Table of Contents](#)

# PyMongo

- The PyMongo package contains tools for interacting with MongoDB database from Python
- The PyMongo package is a native Python driver for MongoDB
- Install using PIP: `pip install pymongo`
- <https://pypi.org/project/pymongo/>



# Basic Python Examples

Hans-Petter Halvorsen

[Table of Contents](#)

# Basic Example – Write Data

```
import pymongo

client = pymongo.MongoClient("mongodb+srv://<user>:<password>@xxx.mongodb.net/")

database = client["Library"]
collection = database["Book"]

document = { "Title": "Database Programming", "Author": "Elvis Presly" }

x = collection.insert_one(document)
```

# Basic Example – Write Data

```
import pymongo
import database

connectionString = database.GetConnectionString()
client = pymongo.MongoClient(connectionString)
database = client["Library"]
collection = database["Book"]

document = { "Title": "Database Programming", "Author": "Elvis Presly" }

x = collection.insert_one(document)
```

```
def GetConnectionString()

    connectionString = "mongodb+srv://<user>:<password>@xxx.mongodb.net/"

    return connectionString
```

database.py

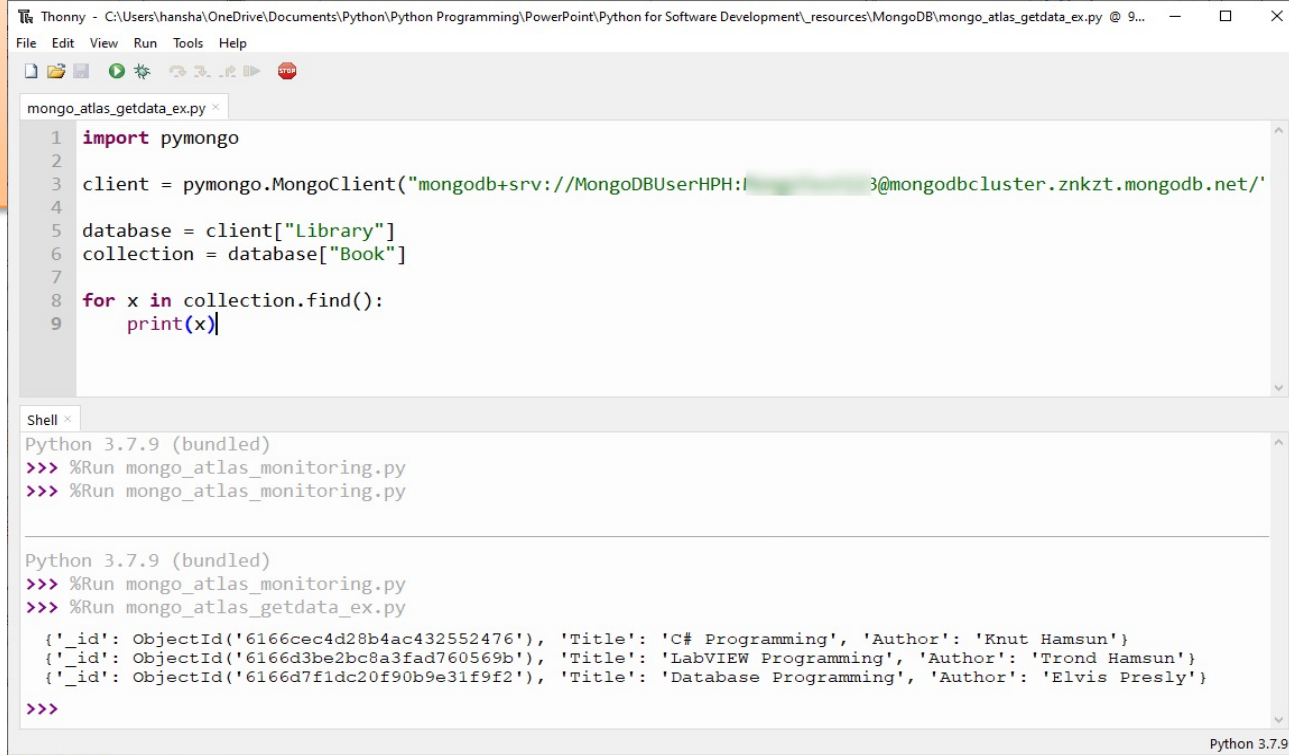
# Basic Example – Read Data

```
import pymongo
```

```
client = pymongo.MongoClient("mongodb+srv://<user>:<password>@xxx.mongodb.net/")
```

```
database = client["Library"]  
collection = database["Book"]
```

```
for x in collection.find():  
    print(x)
```



The screenshot shows a Python IDE window titled 'Thonny - C:\Users\hansha\OneDrive\Documents\Python\Python Programming\PowerPoint\Python for Software Development\\_resources\MongoDB\mongo\_atlas\_getdata\_ex.py @ 9...'. The code in the editor is as follows:

```
1 import pymongo  
2  
3 client = pymongo.MongoClient("mongodb+srv://MongoDBUserHPH:123456789@mongodbcluster.znkzt.mongodb.net/")  
4  
5 database = client["Library"]  
6 collection = database["Book"]  
7  
8 for x in collection.find():  
9     print(x)
```

The Shell window shows the execution of the code:

```
Python 3.7.9 (bundled)  
>>> %Run mongo_atlas_monitoring.py  
>>> %Run mongo_atlas_monitoring.py  
  
Python 3.7.9 (bundled)  
>>> %Run mongo_atlas_monitoring.py  
>>> %Run mongo_atlas_getdata_ex.py  
  
{ '_id': ObjectId('6166cec4d28b4ac432552476'), 'Title': 'C# Programming', 'Author': 'Knut Hamsun'}  
{ '_id': ObjectId('6166d3be2bc8a3fad760569b'), 'Title': 'LabVIEW Programming', 'Author': 'Trond Hamsun'}  
{ '_id': ObjectId('6166d7f1dc20f90b9e31f9f2'), 'Title': 'Database Programming', 'Author': 'Elvis Presly'}  
>>>
```

Python 3.7.9



# Basic Example – Read Data

```
import pymongo
import database

connectionString = database.GetConnectionString()

client = pymongo.MongoClient(connectionString)

database = client["Library"]
collection = database["Book"]

for x in collection.find():
    print(x)
```

```
def GetConnectionString()

    connectionString = "mongodb+srv://<user>:<password>@xxx.mongodb.net/"

    return connectionString
```

database.py



# Datalogging Example

# Datalogging Example

- We can log data from a DAQ device, Raspberry Pi or similar
- We start by creating a simple Random Generator that simulates a Temperature Sensor and log these data to the MongoDB database
- Then we will in another script read the data from the database and plot them.

```
import pymongo
import random
import time
from datetime import datetime
import database

connectionString = database.GetConnectionString()
client = pymongo.MongoClient(connectionString)
database = client["MeasurementSystem"]
collection = database["MeasurementData"]

Ts = 10 # Sampling Time
N = 10
for k in range(N):
    # Generate Random Data
    LowLimit = 20
    UpperLimit = 25
    MeasurementValue = random.randint(LowLimit, UpperLimit)

    #Find Date and Time
    now = datetime.now()
    datetimeformat = "%Y-%m-%d %H:%M:%S"
    MeasurementDateTime = now.strftime(datetimeformat)

    # Insert Data into Database
    document = { "MeasurementValue": MeasurementValue, "MeasurementDateTime": MeasurementDateTime }
    x = collection.insert_one(document)

    # Wait
    time.sleep(Ts)
```

# View Logged Data MongoDB Compass

The screenshot displays the MongoDB Compass interface. On the left sidebar, the 'Local' database is selected, showing a collection named 'MeasurementSystem.MeasurementData'. The main panel shows a list of 8 documents. The filter bar contains the query `{ field: 'value' }`. The document list shows the following data:

Document ID	MeasurementValue	MeasurementDateTime
<code>ObjectId("6166daffc52c9c655d041a82")</code>	21	"2021-10-13 15:11:27"
<code>ObjectId("6166db0bc52c9c655d041a83")</code>	22	"2021-10-13 15:11:39"
<code>ObjectId("6166db15c52c9c655d041a84")</code>	21	"2021-10-13 15:11:49"
<code>ObjectId("6166db1fc52c9c655d041a85")</code>	20	"2021-10-13 15:11:59"
<code>ObjectId("6166db2ac52c9c655d041a86")</code>	20	"2021-10-13 15:12:10"
<code>ObjectId("6166db34c52c9c655d041a87")</code>	23	"2021-10-13 15:12:20"

At the bottom left, the terminal shows the prompt `> _MONGOSH`.



# Data Monitoring Example

# Plotted Data

Thonny - C:\Users\hansha\OneDrive\Documents\Python\Python Programming\PowerPoint\Python for Software Development\\_resources\MongoDB\mongo\_atlas\_monitoring.py @ 11 : 108

File Edit View Run Tools Help

mongo\_atlas\_datalogging.py x mongo\_atlas\_monitoring.py x

```
1 import pymongo
2 import matplotlib.pyplot as plt
3 from datetime import datetime
4
5 # Connect to Database
6 import pymongo
7 import matplotlib.pyplot as plt
8 from datetime import datetime
9
10 # Connect to Database
11 client = pymongo.MongoClient("mongodb+srv://
12 database = client["MeasurementSystem"]
13 collection = database["MeasurementData"]
14
15 t = []
16 data = []
17
18 # Retrieving and Formatting Data
19 for document in collection.find():
20     MeasurementValue = document["MeasurementValue"]
21     MeasurementDateTime = document["MeasurementDateTime"]
22
23     timeformat = "%Y-%m-%d %H:%M:%S"
24     MeasurementDateTime = datetime.strptime(
25
26     data.append(MeasurementValue)
27     t.append(MeasurementDateTime)
28
29 # Plotting
30 plt.plot(t, data, 'o-')
```

Shell x

Python 3.7.9 (bundled)

```
>>> %Run mongo_atlas_monitoring.py
>>> %Run mongo_atlas_monitoring.py
```

Figure 1

Temperature

t [s]	Temp [degC]
15:11:30	21.0
15:11:45	22.0
15:12:00	21.0
15:12:15	20.0
15:12:30	23.0
15:12:45	22.0
15:12:55	24.0
15:13:00	25.0

x=15:12:58 y=22.57

Python 3.7.9

```
import pymongo
import matplotlib.pyplot as plt
from datetime import datetime
import database

# Connect to Database
connectionString = database.GetConnectionString()
client = pymongo.MongoClient(connectionString)
database = client["MeasurementSystem"]
collection = database["MeasurementData"]

t = []
data = []

# Retrieving and Formatting Data
for document in collection.find():
    MeasurementValue = document["MeasurementValue"]
    MeasurementDateTime = document["MeasurementDateTime"]

    timeformat = "%Y-%m-%d %H:%M:%S"
    MeasurementDateTime = datetime.strptime(MeasurementDateTime, timeformat)

    data.append(MeasurementValue)
    t.append(MeasurementDateTime)

# Plotting
plt.plot(t, data, 'o-')
plt.title('Temperature')
plt.xlabel('t [s]')
plt.ylabel('Temp [degC]')
plt.grid()
plt.show()
```



# Hans-Petter Halvorsen

University of South-Eastern Norway

[www.usn.no](http://www.usn.no)

E-mail: [hans.p.halvorsen@usn.no](mailto:hans.p.halvorsen@usn.no)

Web: <https://www.halvorsen.blog>

