MongoDb with python: Basics

Download and setup mongoDb in Windows with python

```
>> from the mongodb website download the executable and install the mongo d
b
>> from cmd run pip install pymongo to install the module for interacting w
ith mongoDB with python
```

Download and setup mongoDB in Ubuntu with python

```
>> Import the MongoDB repository (sudo apt-key adv --keyserver hkp://keyser
ver.ubuntu.com:80 --recv 7F0CEB10)
>> Create a source list file for MongoDB (echo "deb http://repo.mongodb.or
g/apt/ubuntu xenial/mongodb-org/3.4 multiverse" | sudo tee /etc/apt/source
s.list.d/mongodb-org-3.4.list)
>> sudo apt-get install -y mongodb
>> install pymongo module with following command "pip install pymongo"
```

steps to intializing a connectio with mongo db

1. import the MongoClient from pymongo module "from pymongo import MongoClient"

2. create an instance of MongoClient with 2 parameter 'localhost' and the port number - "myclient = MongoClient('localhost',27017)"

3. Check the available database "myclient.list_database_names()"

In [1]:

```
from pymongo import MongoClient
myclient = MongoClient('localhost',27017)
myclient.list_database_names()
```

Out[1]:

```
['admin', 'config', 'demo', 'local']
```

1.Let us learn, how to create a database, a table (called collection in mongodb) and insert values within it.

1. follow above 2 steps - import MongoClient from pymongo, create instance of MongoClient with localhost and port number.

2. Create an instance of your DB (if not present already pymongo will create a new db with your provided name) - mydb = myclient['demo'] # db name

3. Create an instance of your collection (table) (if not exist pymongo will create a table with same name.) - mycoll=mydb['dbtable']

4. Create a list of values to be insterted in your collection (table).

5. execute the command collection.insert_many()

6. Check for the db and the table

In [16]:

```
mydb = myclient['demo'] # db name
mycoll=mydb['dbtable'] # collection name which is same as table name for sql db


mylist = [
    {'id': 1, 'name': 'Rahul Singh', 'mobile' : '8097846982'},
    {'id': 2, 'name': 'Pallavi Hankare', 'mobile' : '8693881508'},
    {'id': 3, 'name': 'Aashish Gupta', 'mobile' : '8082245459'},
    {'id': 4, 'name': 'Aashu Singh', 'mobile' : '87846982'},
    {'id': 5, 'name': 'Vivek Tivari', 'mobile' : '8097846982'},
    ]
# Doc string for inster_many() Insert an iterable of documents.
x = mycoll.insert_many(mylist) # insert query for more than 1 row. for one row 'ins

#Insert a document(s) into this collection. -  but this methode is deprecated
#mycoll.insert()
#Docstring:     A Mongo collection.
#mycoll.inster_one()

#---------------checking for operation results---------#
print(f'available dbs are: {myclient.list_database_names()}')

# Docstring:Get a list of all the collection names in this database.
print(f'avaible collections in our DB demo are {mydb.list_collection_names()}')
```

```
available dbs are: ['admin', 'config', 'demo', 'local']
avaible collections in our DB demo are ['dbtable']
```

2. learn about find_one() and find() which is equal to select statement

SO, once we have created database, created table and inserted a value in table (collection) the next step is to retrive the data from collection.

To do so, we need to perform 4steps

```
#### > import MongoClient from pymongo
#### > create an instance of Mongoclient with localhost and the port numbe
r.
#### > Get refference to the database and then to the collection.
#### > Run find_one and find() methodes on collection object.
```

In [47]:

```
#Docstring: Get a single document from the database.
print(f'result of find_one methode: {mycoll.find_one()}')
print('result of find() methode :')
for d in mycoll.find():
    #print(d)
    pass

#let us use the limit methode to get only certain number of records
for d in mycoll.find().limit(5):
    print(d)
```

```
result of find_one methode: {'_id': ObjectId('5c9894c282769d17df2f20b
2'), 'id': 2, 'name': 'Pallavi Hankare', 'mobile': '8693881508'}
result of find() methode :
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
```

In [29]:

```
mycoll.distinct('name')
```

Out[29]:

```
['Rahul Singh',
 'Pallavi Hankare',
 'Aashish Gupta',
 'Aashu Singh',
 'Vivek Tivari']
```

2.1 Let us learn little bit more about fquerying the data:

find methode always takes dictionary as parameter where key will be the field name the value will be th value of field

In [37]:

```
#get the document or record where the mobile number is 8097846982
docs = mycoll.find({'mobile':'8097846982'})
for d in docs:
    #print(d)
    pass

#query to get records where name is Vivek Tivari
docs = mycoll.find({'name':'Vivek Tivari'})
for d in docs:
    #print(d)
    pass

#use regex to find records where name value starts with A
docs = mycoll.find({'name':{'$regex':'^A'}})
for d in docs:
    print(d)
    pass
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
```

2.3 So we have learned on retriving the data, let us learn about sorting the data.

In [40]:

```
docs = mycoll.find().sort('id',1) #sorting in
for d in docs:
    #print(d)
    pass

docs = mycoll.find().sort('id',-1) #sorting in
for d in docs:
    print(d)
type(docs)
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98984a82769d18dad0a817'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81c'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a821'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989acc82769d18dad0a826'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98999b82769d18dad0a819'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989ab382769d18dad0a81e'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989acc82769d18dad0a823'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b1'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c98984a82769d18dad0a813'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a818'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81d'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
```

```
{'_id': ObjectId('5c989acc82769d18dad0a822'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
```

```
Out[40]:
```

```
pymongo.cursor.Cursor
```

3. let us learn about delete operations

methods for deleting the data are delete_one() and the delete_many()

In [42]:

```
# so the delete one will delete the first record whic matches the query criteria.
#for example the table had 5 rows with value "Rahul Singh", when we ran below comma
#the delete_one methode deleted the only first rown where the name was "Rahul Singh
docs = mycoll.delete_one({'name':'Rahul Singh'})

for d in mycoll.find():
    print(d)
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98984a82769d18dad0a813'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a817'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a818'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a819'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81c'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81d'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81e'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a821'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989acc82769d18dad0a822'), 'id': 1, 'name': 'Rahul
Singh', 'mobile': '8097846982'}
{'_id': ObjectId('5c989acc82769d18dad0a823'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a826'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
```

In [43]:

```
#let us illustrate the working of delete_many()
#as expacted the delete_many methode deleted all the rows which met with query crit
docs =  mycoll.delete_many({'name':'Rahul Singh'})
for d in mycoll.find():
    print(d)
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a817'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a819'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81c'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81e'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a821'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989acc82769d18dad0a823'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a826'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
```

4. The last basic functionality is update methode. Let us learn on how to update the data in mongoDB

again there are two methodes for updating the data in mongodb : update_one() and update_many()

In [45]:

```
query = {'name':'Vivek Tivari'}
newValue = {'$set':{'mobile':'Lost!'}}
#mycoll.update_one(filter, update)
# Docstring: Update a single document matching the filter.
mycoll.update_one(query,newValue)

#check the updated values
for d in mycoll.find():
            print(d)
#as expacted update_one finds the first record matching with filter or the query ar
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a817'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c98999b82769d18dad0a819'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81c'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989ab382769d18dad0a81e'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a821'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
{'_id': ObjectId('5c989acc82769d18dad0a823'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a826'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': '8097846982'}
```

In [46]:

```
#let us illustrate the update_many methode.
query = {'name':'Vivek Tivari'}
newValue = {'$set':{'mobile':'Lost!'}}

mycoll.update_many(query,newValue)

#check the updated values
for d in mycoll.find():
            print(d)
```

```
{'_id': ObjectId('5c9894c282769d17df2f20b2'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c9894c282769d17df2f20b3'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c9894c282769d17df2f20b4'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c9894c282769d17df2f20b5'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c98984a82769d18dad0a814'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98984a82769d18dad0a815'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98984a82769d18dad0a816'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98984a82769d18dad0a817'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c98999b82769d18dad0a819'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c98999b82769d18dad0a81a'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c98999b82769d18dad0a81b'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c98999b82769d18dad0a81c'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c989ab382769d18dad0a81e'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989ab382769d18dad0a81f'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989ab382769d18dad0a820'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989ab382769d18dad0a821'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
{'_id': ObjectId('5c989acc82769d18dad0a823'), 'id': 2, 'name': 'Pallav
i Hankare', 'mobile': '8693881508'}
{'_id': ObjectId('5c989acc82769d18dad0a824'), 'id': 3, 'name': 'Aashis
h Gupta', 'mobile': '8082245459'}
{'_id': ObjectId('5c989acc82769d18dad0a825'), 'id': 4, 'name': 'Aashu
Singh', 'mobile': '87846982'}
{'_id': ObjectId('5c989acc82769d18dad0a826'), 'id': 5, 'name': 'Vivek
Tivari', 'mobile': 'Lost!'}
```

5. Last thing to do is, deleting the table and deleting the database.

To delete any db or tables object in mongodb, simply use the drop() methode.

In [50]:

```
#before deleting let us get the list of collections or the tables availble in our d
print(mydb.list_collection_names())
#let us drop the collection 'dbtables' from demo db and verify
print(mycoll.drop())
print(f'list of availble db tables or collections after dropping dbtable {mydb.list
```

```
['dbtable']
None
list of availble db tables or collections after dropping dbtable []
```

In [51]:

```
#let us drop the demo db from mongo db repository
print(f'list of availble dbs before deleting the demo db : {myclient.list_database_
myclient.drop_database('demo')
print(f'list of availble dbs before deleting the demo db : {myclient.list_database_
```

```
list of availble dbs before deleting the demo db : ['admin', 'config',
'local']
list of availble dbs before deleting the demo db : ['admin', 'config',
'local']
```