

Python MongoDB:

Pymongo installation

pip install pymongo

pip install Flask-PyMongo

use admin

```
> db.createUser({
... user:"SocialMedia",
... pwd:"SocialMedia#2019@",
... roles:[{
... role:"userAdmin",
... db:"socialmedia"
... }]
... })
Successfully added user: {
  "user" : "SocialMedia",
  "roles" : [
    {
      "role" : "userAdmin",
      "db" : "socialmedia"
    }
  ]
}
```

use socialmedia

```

> use socialmedia
switched to db socialmedia
> db.createUser({
... user:"SocialMedia",
... pwd:"SocialMedia#2019",
... roles:[{
... role:"dbOwner",
... db:"socialmedia"
... }]
... })
Successfully added user: {
  "user" : "SocialMedia",
  "roles" : [
    {
      "role" : "dbOwner",
      "db" : "socialmedia"
    }
  ]
}
> db.getUsers()
[
  {
    "_id" : "socialmedia.SocialMedia",
    "user" : "SocialMedia",
    "db" : "socialmedia",
    "roles" : [
      {
        "role" : "dbOwner",
        "db" : "socialmedia"
      }
    ],
    "mechanisms" : [
      "SCRAM-SHA-1",
      "SCRAM-SHA-256"
    ]
  }
]

```

```
mycol = mongoconn.admin_users.find_one()
```

```
print(mycol)
```

```
#for x in mycol.find():
```

```
#    print(x)
```

```
#admin_users=mongoconn.admin_users.find()
```

Create User:

use digitalbox

```
db.createUser({
  user: 'digital',
  pwd: 'digital#2020@',
  roles: [{ role: 'userAdmin', db:'digitalbox'}]
})
```

//mongodump

cd C:\Program Files\MongoDB\Server\4.0\bin\

mongodump --db digitalbox

mongorestore --db digitalbox dump/digitalbox

Admin Credentials :

admin / 123456

pbkdf2_sha256\$150000\$IHAD92qoxQqY\$tdq/FoUB8DWwTvbS8UWZpV+q+FpU38XO4JsxXecePB1Y=

User Credentials:

Search Text:

```
db.lunchbox_Videos.createIndex( { title: "text", description: "text" } )
```

<https://stackoverflow.com/questions/48371016/pymongo-how-to-use-full-text-search>

```
def search_for_videos(search_text):
```

```
    collection.find({"$text": {"$search": search_text}}).limit(10)
```

```
db.lunchbox_Videos.createIndex( { title: "text", description: "text", keywords: "text", category:
"text", customkwd: "text" } )
```

You can use native Python types and Pymongo will automatically translate them into BSON - ISODate in this case

```
{  
  "$match": {  
    "Date": { "$gte": datetime.datetime(2016, 1, 1, 0, 0), "$lt": datetime.datetime(2019, 1, 1, 0, 0) }  
  }  
}
```