Laboratorio 2

Ejecutar un contenedor que corre MongoDB y con el cual nos conectaremos por medio de Python

Ultimar versión de mongo: docker pull mongo:latest

```
Status: Downloaded newer image for mongo:latest
docker.io/library/mongo:latest
ubuntu $
```

```
latest: Pulling from library/mongo
10ac4908093d: Pull complete
685504455d09: Pull complete
ebd36404f329: Pull complete
3abd9b25affb: Pull complete
2d7fde532eae: Pull complete
24fc70e4c7d7: Pull complete
ffc2353072f7: Pull complete
560de8e3a6c7: Pull complete
0748cd1d792c: Pull complete
Digest: sha256:2374c2525c598566cc4e62145ba65aecfe1bd3bf090cccce1ca44f3e2b60f861
Status: Downloaded newer image for mongo:latest
```

1. Iniciar el container de MongoDB utilizando el comando docker run -d -p 27017:27017 --name m1 mongo

```
ubuntu $ docker run -d -p 27017:27017 --name m1 mongo
94ed2692f4cde53ba69ac238abb0f48a4d336f063c2d455493b92a92535fbfdb
ubuntu $
```

Puedes conectarte al contenedor de Mongo con docker exec -it m1 /bin/bash y luego conectarte a MongoDBpor medio del comando mongosh

docker exec -it m1 /bin/bash

```
ubuntu $ docker exec -it m1 /bin/bash
root@b5a6b43d8126:/#
```

mongosh

```
root@94ed2692f4cd:/# mongosh
Current Mongosh Log ID: 63fac0135d648e9d49180474
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.6.2
Using MongoDB:          6.0.4
Using Mongosh:          1.6.2

For mongosh info see: https://docs.mongodb.com/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2023-02-26T02:10:18.298+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/p
rodnotes-filesystem
   2023-02-26T02:10:19.131+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2023-02-26T02:10:19.131+00:00: vm.max_map_count is too low
------

------
   Enable MongoDB's free cloud-based monitoring service, which will then receive and display
   metrics about your deployment (disk utilization, CPU, operation statistics, etc).

   The monitoring data will be available on a MongoDB website with a unique URL accessible to you
   and anyone you share the URL with. MongoDB may use this information to make product
   improvements and to suggest MongoDB products and deployment options to you.

   To enable free monitoring, run the following command: db.enableFreeMonitoring()
   To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
------
```

Listar las bases de datos

```
test> show databases;
admin   40.00 KiB
config  12.00 KiB
local   40.00 KiB
test>
```

Ejemplo de trabajar con una tabla

```
mi-db> db.Employee.insert(
...    {
...      "Employeename" : "Chris",
...      "EmployeeDepartment" : "Sales"
...    }
... )
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("63fac25273c538d7fc7c65c4") }
}
mi-db>
```

Verificar tablas creada

```
already on db mi-db
mi-db> show tables;
Employee
mi-db>
```

Salir de mongo

```
mi-db> exit
```

```
root@94ed2692f4cd:/#
exit
ubuntu $
```

2. Utilizaremos los scripts de Python existentes en la carpeta para la colección de mongo, utilizando la librería https://api.mongodb.com/python/current/tutorial.html

i. Instalar pip install pymongo

```
ubuntu $ pip install pymongo
Collecting pymongo
  Downloading pymongo-4.3.3-cp38-cp38-manylinux2014_x86_64.whl (519 kB)
     |████████████████████████████████| 519 kB 12.7 MB/s
Collecting dnspython<3.0.0,>=1.16.0
  Downloading dnspython-2.3.0-py3-none-any.whl (283 kB)
     |████████████████████████████████| 283 kB 25.3 MB/s
Installing collected packages: dnspython, pymongo
Successfully installed dnspython-2.3.0 pymongo-4.3.3
ubuntu $
```

ii. Ejecuta los scripts con python populate.py ypython find.py

Creación de los archivos populate.py y find.py

```
ubuntu $ nano populate.py
ubuntu $ nano populate.py
ubuntu $ nano find.py
ubuntu $ ls -la
total 44
drwx------  5 root root 4096 Feb 26 02:29 .
drwxr-xr-x 19 root root 4096 Feb 23 12:40 ..
-rw-------  1 root root   20 Nov 13 17:27 .bash_history
-rw-r--r--  1 root root 3208 Feb 23 12:40 .bashrc
drwxr-xr-x  3 root root 4096 Feb 26 02:27 .local
-rw-r--r--  1 root root  161 Dec  5  2019 .profile
drwx------  2 root root 4096 Feb 23 12:37 .ssh
drwxr-xr-x  6 root root 4096 Feb 26 02:09 .theia
-rw-r--r--  1 root root  109 Feb 26 02:04 .vimrc
lrwxrwxrwx  1 root root    1 Feb 23 12:40 filesystem -> /
-rw-r--r--  1 root root  282 Feb 26 02:29 find.py
-rw-r--r--  1 root root  682 Feb 26 02:28 populate.py
ubuntu $
```

Ejecutar python populate.py

```
ubuntu $ python populate.py
Nombre de la DB:  mi-db
<pymongo.results.InsertManyResult object at 0x7f61b1268f70>
```

Ejecutar python find.py

```
ubuntu $ python find.py
Imprime un registro
 {'_id': ObjectId('63fac4abdd02b08fc9cd19d3'), 'name': 'firulais', 'owner': 'jahir', 'specie': 'perro'}

Imprime todos los registros
{'_id': ObjectId('63fac4abdd02b08fc9cd19d3'), 'name': 'firulais', 'owner': 'jahir', 'specie': 'perro'}
{'_id': ObjectId('63fac4abdd02b08fc9cd19d4'), 'name': 'taco', 'owner': 'jonathan', 'specie': 'perro'}
{'_id': ObjectId('63fac4abdd02b08fc9cd19d5'), 'name': 'garfield', 'owner': 'erick', 'specie': 'gato'}
{'_id': ObjectId('63fac4abdd02b08fc9cd19d6'), 'name': 'charlotte', 'owner': 'juan daniel', 'specie': 'araña'}
{'_id': ObjectId('63fac4abdd02b08fc9cd19d7'), 'name': 'solovino', 'owner': 'jorge', 'specie': 'cuyo'}
ubuntu $
```

iii. Revisa los registros por medio del CLI de mongo o de tu DBMS favorito

```
ubuntu $ docker exec -it m1 /bin/bash
root@94ed2692f4cd:/# mongosh
Current Mongosh Log ID: 63fac6d3be25537bdc51a089
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.6.2
Using MongoDB:          6.0.4
Using Mongosh:          1.6.2

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------
   The server generated these startup warnings when booting
   2023-02-26T02:10:18.298+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/p
rodnotes-filesystem
   2023-02-26T02:10:19.131+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2023-02-26T02:10:19.131+00:00: vm.max_map_count is too low
------

------
   Enable MongoDB's free cloud-based monitoring service, which will then receive and display
   metrics about your deployment (disk utilization, CPU, operation statistics, etc).

   The monitoring data will be available on a MongoDB website with a unique URL accessible to you
   and anyone you share the URL with. MongoDB may use this information to make product
   improvements and to suggest MongoDB products and deployment options to you.

   To enable free monitoring, run the following command: db.enableFreeMonitoring()
   To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
------

test>
```

Seleccionar la base de datos y la collection
show databases;
show collections;

```
test> show databases;
admin   40.00 KiB
config  72.00 KiB
local   40.00 KiB
mi-db   80.00 KiB
test> use mi-db
switched to db mi-db
mi-db> show collections
show collections

mi-db> show collections;
Employee
pet
```

Consultar contenido de la tabla pet

```
mi-db> db.pet.find()
[
  {
    _id: ObjectId("63fac4abdd02b08fc9cd19d3"),
    name: 'firulais',
    owner: 'jahir',
    specie: 'perro'
  },
  {
    _id: ObjectId("63fac4abdd02b08fc9cd19d4"),
    name: 'taco',
    owner: 'jonathan',
    specie: 'perro'
  },
  {
    _id: ObjectId("63fac4abdd02b08fc9cd19d5"),
    name: 'garfield',
    owner: 'erick',
    specie: 'gato'
  },
  {
    _id: ObjectId("63fac4abdd02b08fc9cd19d6"),
    name: 'charlotte',
    owner: 'juan daniel',
    specie: 'araña'
  },
  {
    _id: ObjectId("63fac4abdd02b08fc9cd19d7"),
    name: 'solovino',
    owner: 'jorge',
    specie: 'cuyo'
  }
]
mi-db>
```

3. Una vez que hayas terminado de jugar con MongoDB y los scripts de Python, asegúrate de detener y remover el contenedor de MongoDB en ejecución utilizando docker stop m1; docker rm m1

```
mi-db> exit
root@94ed2692f4cd:/#
exit
ubuntu $
```

```
ubuntu $ docker stop m1
m1
ubuntu $ docker rm m1
m1
ubuntu $ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
ubuntu $
```