

# ESCUELA POLITÉCNICA NACIONAL ESCUELA DE FORMACIÓN DE TECNÓLOGOS



## ANÁLISIS DE DATOS

**(TDSD353)** 

ASIGNATURA: Análisis de datos

PROFESOR: Ing. Juan Pablo Zaldumbide Proaño.

PERÍODO ACADÉMICO: 2021-B

## DEBER 1

## MongoDB



Nombre del estudiante: Farinango Eduardo

#### **INSTRUCCIONES**

```
1.- Generar un script que ingrese "n" ciudadanos en MongoDB, conformados por {
        "nombres_completos":"<conformado por dos elementos al azar de la lista nombres>",
        "apellidos_completos":"<conformado por dos elementos al azar de la lista apellidos>",
        "edad":"<valor al azar entero, mayor de edad y menor que 80 años>",
        "cedula":"<string al azar formado de 10 dígitos>"
}
```

2.- Crear una función que muestre los elementos ingresados en la base de datos. Debe ejecutar el ejercicio localmente y en la nube.

Entregables: Capturas de pantalla con los pasos seguidos y código fuente (.ipynb o .py) en un repositorio de github

#### Desarrollo:

Para podernos conectar a Atlas MongoDB debemos primeramente instalar en driver respectivo ue creara la conexión.

```
Anaconda Prompt (anaconda3)

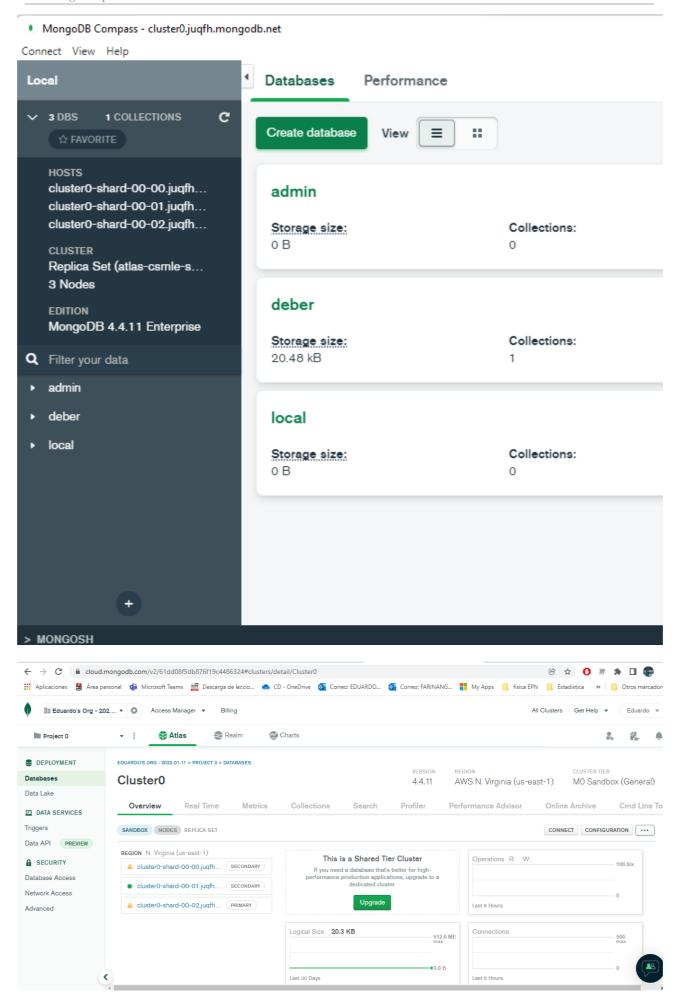
(base) D:\AD\Jupyter>pip install pymongo
WARNING: Ignoring invalid distribution -ymongo (c:\users\eduardo\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ymongo (c:\users\eduardo\anaconda3\lib\site-packages)
Requirement already satisfied: pymongo in c:\users\eduardo\anaconda3\lib\site-packages (4.0.1)
WARNING: Ignoring invalid distribution -ymongo (c:\users\eduardo\anaconda3\lib\site-packages)
```

Una vez instalado en Jupyter importamos MongoCLient de nuestra librería pymongo.

```
#Farinango Eduardo
import couchdb

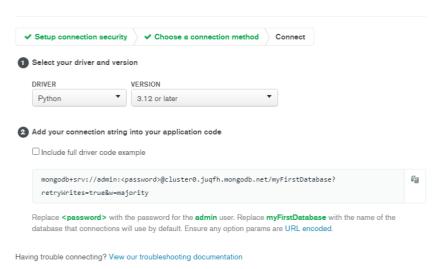
import random
import string
import numpy as np
import pymongo
from pymongo import MongoClient
```

Por el lado del servidor en la nube creamos una nueva conexión utilizando MongoDB Compass localmente utilizando las credenciales ingresadas al momento de la instalación.



Una vez creado el cluster0 deberemos crear la conexión de Python a Atlas MongoDB, este nos brinda una opción para crear un cliente y el código necesario con los parámetros aceptados.

#### Connect to ClusterO



#### Creamos nuestro cliente

```
try:
    client = pymongo.MongoClient('mongodb+srv://admin:casa@cluster0.juqfh.mongodb.net/myFirstDatabase?retryWrites=true&w=
    client.server_info()
    print("Conexion exitosa")
    client.close
    except pymongo.errors.ServerSelectionTimeoutError as errorTiempo:
    print("Conexión rechazada")

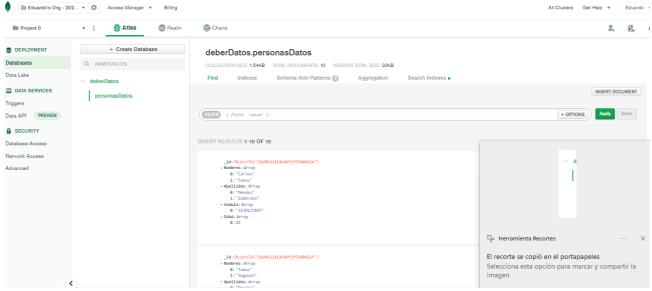
Conexion exitosa
]: dbm = client['deberDatos']
    col = dbm['personasDatos']
```

Creamos nuestro script que nos permita generar las n personas con sus datos randomicos.

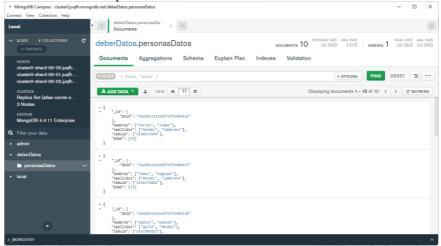
```
cantidad = int(input("Ingrese la cantidad de personas"))
#cantidad = 20
tamanio = 10
age = np.arange(18,80)
edad =age.tolist()
for x in range(cantidad):
    aux = (''.join(random.choice(string.digits) for _ in range(tamanio)))
    cedula.append(aux)
for i in range (cantidad):
    col.insert_one({"Nombres":[random.choice(nombre),random.choice(nombre)],
           "Apellidos":[random.choice(apellido),random.choice(apellido)],
"Cedula":[random.choice(cedula)],
           "Edad":[random.choice(edad)]})
for i in range (cantidad):
    doc={"Nombres":[random.choice(nombre), random.choice(nombre)],
    "Apellidos":[random.choice(apellido), random.choice(apellido)],
           "Cedula":[random.choice(cedula)],
          "Edad":[random.choice(edad)]}
    db.save(doc)
```

Ingrese la cantidad de personas 10

### Procedemos a revisar del lado del servidor.



Y del lado de Compass.



Para poder realizar el ejercicio en CouchDB iniciamos con nuestras credenciales, creamos la base de datos a la que vamos a apuntar, utilizando los drivers necesarios y realizando la conexión con el cliente procedemos a ingresar los datos de las personas.

```
couch=couchdb.Server('http://admin:casa@127.0.0.1:5984')
```

#### |: db=couch['datos\_personas']

