

06 / 06 / 2024



Sergio Motilla Mas

Bases de Dades – 1r DAM

Mini-projecte II:

Inscripció d'Alumnes

amb  MongoDB®

 **/sergioxrzz/miniprojecte-II**

# Índex

1	Introducció.....	3
2	Ferramentes & mètodes.....	3
2.1	Ferramentes.....	3
2.2	Mètodes.....	3
3	Perspectiva estàtica.....	5
3.1	Esquema Entitat/Relació (E/R).....	5
3.2	Pas a taules.....	5
3.3	DDL – Data Definition Language.....	6
3.4	DML – Data Manipulation Language.....	6
3.5	DQL – Data Query Language.....	7
4	Perspectiva Dinàmica.....	8
4.1	Diagrama.....	8
5	Conclusions.....	8
6	Bibliografia & webgrafia.....	9



# 1 Introducció

Benvinguts al micro-projecte de Base de Dades d'Inscripció d'Alumnes, de Sergio Motilla Mas per a 1<sup>r</sup> de DAM.

En l'era digital actual, la gestió eficient de la informació és fonamental per a qualsevol institució educativa. Aquest projecte naix de la necessitat de crear una solució informàtica que facilite la inscripció i el seguiment dels alumnes en diversos cursos i programes acadèmics.

A aquesta segona part del projecte ens trobarem amb un programeta que ens permetrà inserir alguns paràmetres bàsics per a l'inscripció d'un alumne/a i gestionar-les amb Python i MongoDB.

## 2 Ferramentes & Mètodes

### 2.1 Ferramentes

Per a fer aquest projecte s'ha fet ús de l'IDE 'Visual Studio Code' per fer el codi en Python, 'MongoDB (recomanable instal·lar 'MongoDB Compass' si voleu fer ús de una IU com a 'MySQL Workbench')' i 'Oracle VM VirtualBox'

S'ha utilitzat MongoDB 4.4.8 en una MV Ubuntu 20, ja que versions posteriors ( $\geq 5.0$ ) donen error a les MV al executar el servei de mongod (CORE-DUMP) i aquestes versions no es troben disponibles a les versions posteriors a Ubuntu 20.

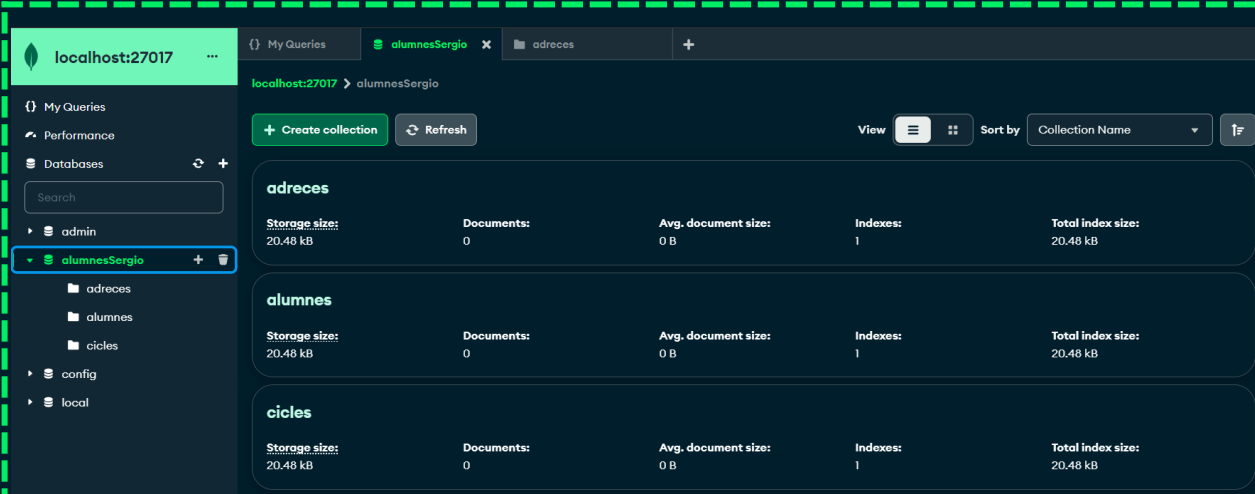
### 2.2 Mètodes

Per a poder fer el projecte s'ha fet ús de *Python 3.12.3*, al qual se li han importat les llibreries de 'tkinter' per poder crear una interfície gràfica, i de 'pymongo' per permetre una connexió entre el codi de Python i la base de dades i tot el que la envolta.

En el host, s'instal·larà el MongoDB on es farà tot el relacionat en la base de dades, per després fer una connexió a la MV (on també es deurà tindre instal·lat MongoDB).

Una vegada instal·lat tot, es crea la base de dades (HOST) i el script de Python que farà que es connecte a aquesta, mentre que en la MV s'instal·la la versió que li corresponga (recursos al final del document).

A l'arxiu de configuració de MongoDB del host s'haurà de posar en l'apartat de bindIp: 'bindIp: 127.0.0.1, IpDelHost', després s'haurà de comprovar que la connexió es possible entre les 2 màquines (si alguna es Windows, pot haver-hi conflicte per el Firewall, per tant seria recomanable crear una regla [i evitar desactivar el Firewall] per permetre l'accés des del port [per defecte 27017]), reiniciar el servei de mongod i seguidament fer des de la màquina 'client' «mongosh -host [IP del host]»



The screenshot shows the MongoDB Compass web interface. On the left, a sidebar lists databases: 'admin', 'alumnosSergio', 'config', and 'local'. The 'alumnosSergio' database is selected. The main panel shows three collections: 'adreces', 'alumnos', and 'cicles'. Each collection has a table with the following columns: 'Storage size', 'Documents', 'Avg. document size', 'Indexes', and 'Total index size'. For 'adreces', the values are 20.48 kB, 0, 0 B, 1, and 20.48 kB respectively. The same values are shown for 'alumnos' and 'cicles'.

Màquina HOST, base de dades amb les "collections" 'alumnos', 'adreces' i 'cicles' - [MongoDB Compass].



The terminal screenshot shows the following commands and output:

```
sergiom@ubuntu20:~$ mongosh --host 192.168.1.40
Current Mongosh Log ID: 665f9d4b9030be5c42a26a12
Connecting to:      mongodb://192.168.1.40:27017/?directConnection=true&appName=mongosh+2.2.6
Using MongoDB:      7.0.11
Using Mongosh:      2.2.6

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

-----
  The server generated these startup warnings when booting
  2024-06-05T00:50:22.360+02:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

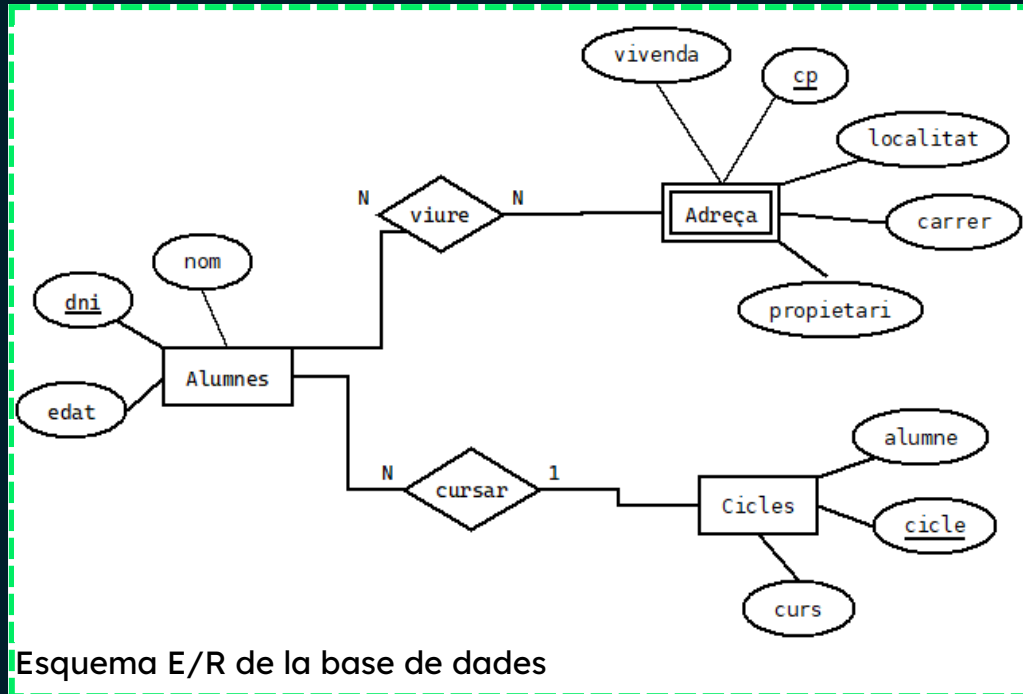
test> show dbs
admin          40.00 KiB
alumnosSergio 144.00 KiB
config         72.00 KiB
local          72.00 KiB
test> use alumnosSergio
switched to db alumnosSergio
alumnosSergio> show collections
adreces
alumnos
cicles
```

Màquina CLIENT(VM), el qual es connecta mitjançant 'mongosh' a la base de dades 'alumnosSergio' del host.



## 3 Perspectiva Estàtica

### 3.1 Esquema Entitat/Relació (E/R)



### 3.2 Pas a taules

(E) **Alumnes**: dni + nom + edat

- (E) **Adreça**: cp + localitat + carrer + vivenda + propietari

C.Ali: propietari → Alumnes(dni)

- (E) **Cicles**

C.Ali: alumne → Alumnes(dni)

- (R) **Viure**: dni + cp

C.Ali: dni → Alumnes(dni)

cp → Adreça(cp)

- (R) **Cursar**: dni + cicle

C.Ali: dni → Alumnes(dni)

cicle → Cicles(cicle)





### 3.3 DDL – Data Definition Language

Inclós en el DML (vore següent apartat).

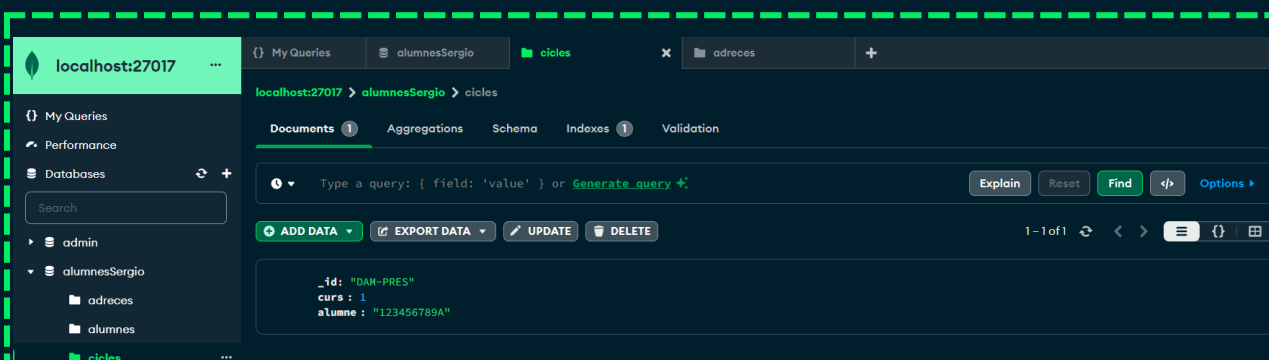
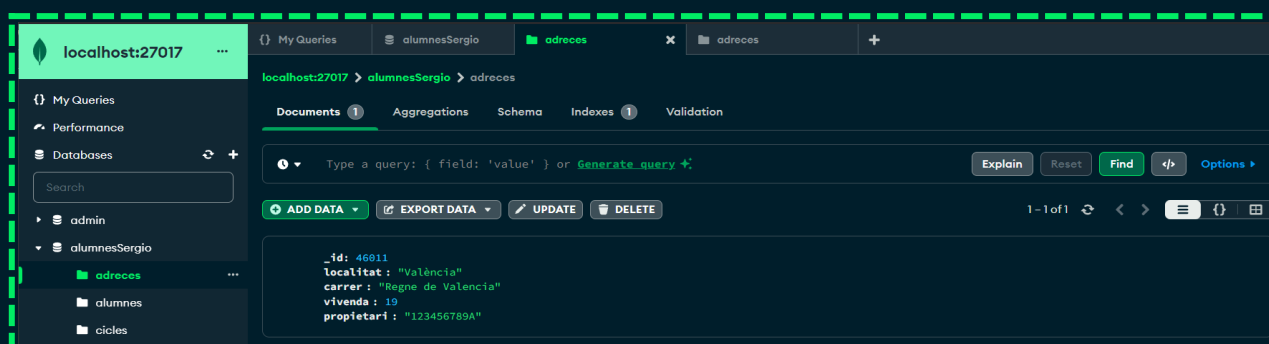
### 3.4 DML – Data Manipulation Language

Ja que les colleccions es creen al mateix temps que se li insereixen les dades, el DDL i el DML van un poc de la mà.

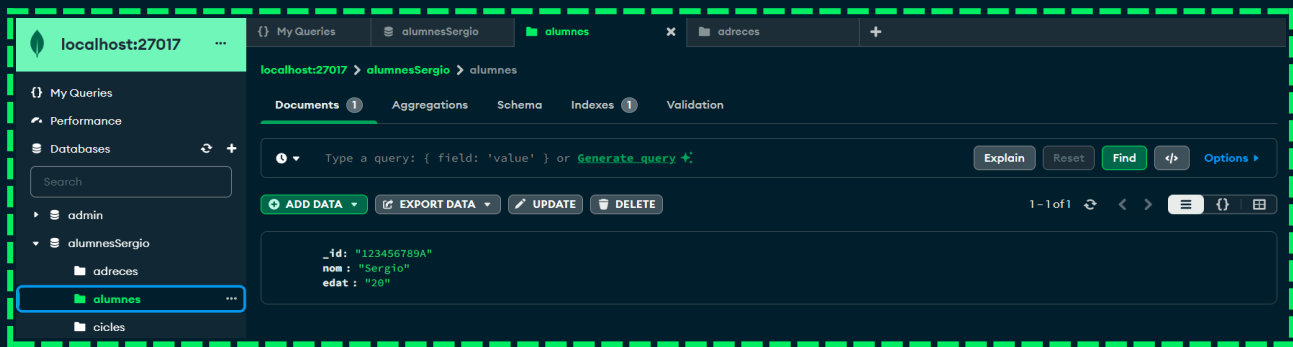
```
alumnSergio> show collections
adreces
alumnSergio
cicles
alumnSergio> db.alumnSergio.insertOne({_id:"123456789A",nom:"Sergio",edat:"20"})
alumnSergio> db.adreces.insertOne({_id:46011,localitat:"València",carrer:"Regne de Valencia",vivenda:19,propietari:"123456789A"})
{ acknowledged: true, insertedId: 46011 }
alumnSergio> db.c
db.constructor          db.createUser          db.changeUserPassword  db.createCollection
db.createEncryptedCollection db.createView          db.createRole           db.currentOp
db.commandHelp          db.cicles

alumnSergio> db.cicles.insertOne({_id:"DAM-PRES",curs:1,alumne:"123456789A"})
{ acknowledged: true, insertedId: 'DAM-PRES' }
alumnSergio>
```

Client, creant dades en les colleccions de la BD - [mongosh]



Comprovació al host com a que s'han inserit correctament, en 'MongoDB Compass' per fer-ho més visual.



## 3.5 DQL – Data Query Language

```
alumnosSergio> db.adreces.findOne({propietari:"123456789A"})
{
  _id: 46011,
  localitat: 'València',
  carrer: 'Regne de Valencia',
  vivenda: 19,
  propietari: '123456789A'
}
alumnosSergio> db.alumnos.findOne({nom:"Sergio"})
{ _id: '123456789A', nom: 'Sergio', edat: '20' }
alumnosSergio>
```

## 4 Perspectiva Dinàmica



### 4.1 Diagrama



## 5 Conclusions

Es una base de dades bastant senzilla, que no conté moltes entitats ni massa atributs, però que compleix amb algunes necessitats essencials a l'hora de poder inscriure a alumnes.

A futur es planteja afegir professors, les notes de les avaluacions dels alumnes o qualsevol tipus d'estadística útil.



## 6 Bibliografia & Webgrafia

GitHub: [github.com/sergioxrzz/mini-projecte-II-mongoDB.com](https://github.com/sergioxrzz/mini-projecte-II-mongoDB.com)

MongoDB: [mongodb.com](https://mongodb.com)

Python 3.12: [python.org/es](https://python.org/es)

pymongo (llibreria de Python per a MongoDB): [pypi.org/project/pymongo](https://pypi.org/project/pymongo)

Instal·lació paquets **.deb** (per a les VM):

[server](#) | [shell](#) | [mongos](#) |

mongosh → 'apt install mongodb-mongosh' [terminal]

Explicació de l'error a les MV amb MongoDB (>=5.0): >> [link](#) <<

Estils del document amb 'estil MongoDB': >> [MongoDB Press Kit](#) <<