

Curso de introducción al desarrollo web con Flask

Rodolfo Ferro

rodolfo.ferro@leon.gob.mx

LAB León



Módulo 5:

Bases de datos (MongoDB)

¿Qué es una base de datos?

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

Algunas características de los sistemas de base de datos:

- Acceso concurrente por parte de múltiples usuarios.
- Consultas complejas optimizadas.
- Seguridad de acceso.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación.



Relacional vs. NO Relacional

El principio de las bases de datos relacionales se basa en la organización de la información en trozos pequeños, que se relacionan entre ellos mediante la relación de identificadores.

Como su propio nombre indica, las bases de datos no relacionales son las que, a diferencia de las relacionales, no tienen un identificador que sirva de relación entre un conjunto de datos y otros. Como veremos, la información se organiza normalmente mediante documentos y es muy útil cuando **no tenemos un esquema exacto de lo que se va a almacenar**.

Relacional vs. NO Relacional

student_id	age	score
1	12	77
2	12	68
3	11	75



```
[  
  {  
    "student_id":1,  
    "age":12,  
    "score":77  
  },  
  {  
    "student_id":2,  
    "age":12,  
    "score":68  
  },  
  {  
    "student_id":3,  
    "age":11,  
    "score":75  
  }  
]
```

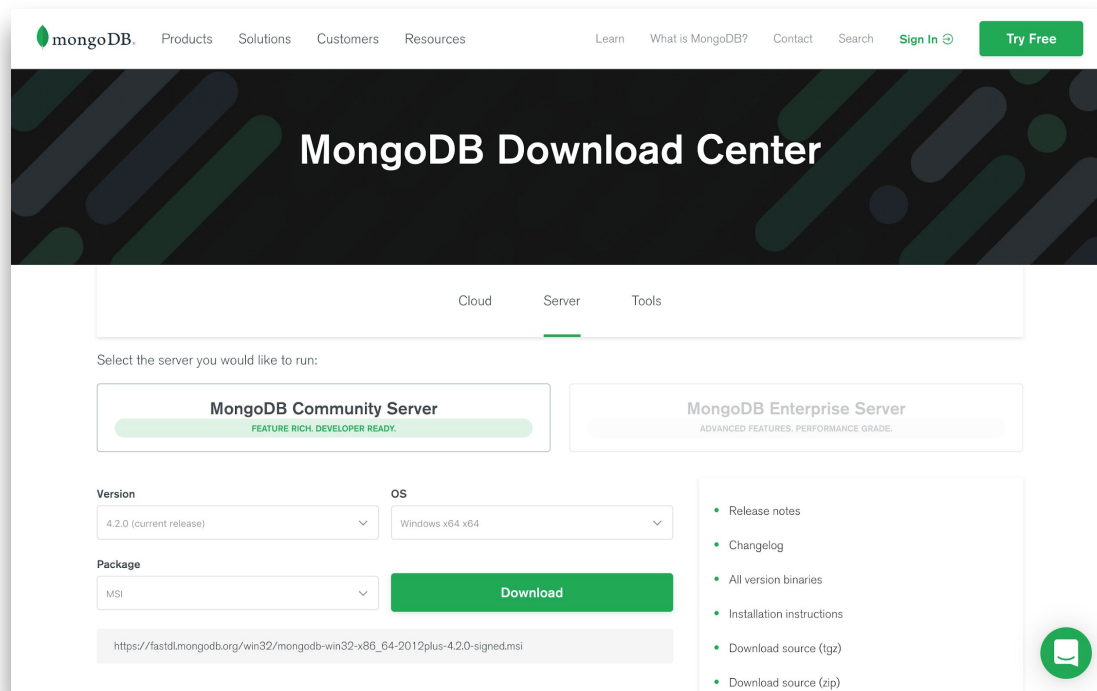
Setup del curso

Para el desarrollo de las prácticas contenidas en el taller, estaremos trabajando sobre scripts de Python, por lo que necesitaremos instalar algunos requerimientos.

La lista de requerimientos es la siguiente:

- Python y PIP
- MongoDB
- Flask y pymongo
- Atom / Sublime Text

Instalación de MongoDB



The screenshot shows the MongoDB Download Center interface. At the top, there is a navigation bar with links for Products, Solutions, Customers, Resources, Learn, What is MongoDB?, Contact, Search, Sign In, and a Try Free button. The main header area has a dark background with the text "MongoDB Download Center". Below this, there are tabs for Cloud, Server (which is selected), and Tools. A message says "Select the server you would like to run:". There are two main options: "MongoDB Community Server" (labeled "FEATURE RICH. DEVELOPER READY.") and "MongoDB Enterprise Server" (labeled "ADVANCED FEATURES. PERFORMANCE GRADE."). Under the Community Server option, there are dropdown menus for Version (4.2.0 (current release)) and OS (Windows x64 x64). There is also a Package dropdown menu (MSI) and a large green Download button. Below the Download button, a URL is provided: https://fastdl.mongodb.org/win32/mongodb-win32-x86_64-2012plus-4.2.0-signed.msi. On the right side, there is a list of links: Release notes, Changelog, All version binaries, Installation instructions, Download source (tgz), and Download source (zip). A green chat icon is visible in the bottom right corner.

mongoDB. Products Solutions Customers Resources Learn What is MongoDB? Contact Search Sign In Try Free

MongoDB Download Center

Cloud **Server** Tools

Select the server you would like to run:

MongoDB Community Server
FEATURE RICH. DEVELOPER READY.

MongoDB Enterprise Server
ADVANCED FEATURES. PERFORMANCE GRADE.

Version: 4.2.0 (current release) OS: Windows x64 x64

Package: MSI **Download**

https://fastdl.mongodb.org/win32/mongodb-win32-x86_64-2012plus-4.2.0-signed.msi

- Release notes
- Changelog
- All version binaries
- Installation instructions
- Download source (tgz)
- Download source (zip)

<https://www.mongodb.com/download-center/community>

Instalación de pymongo

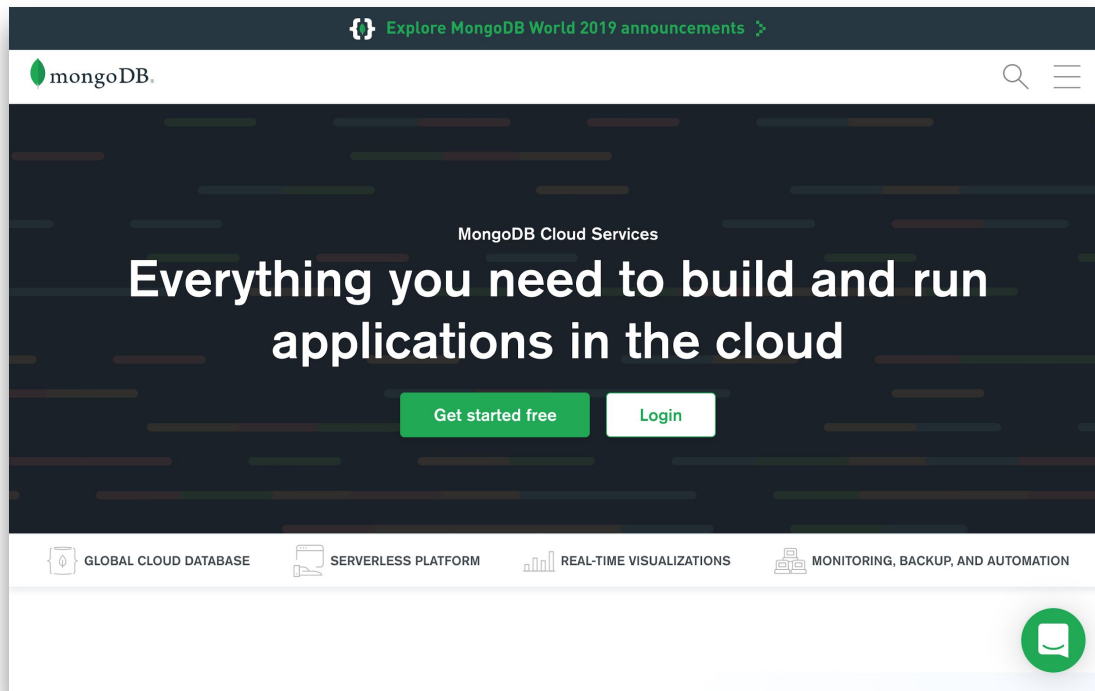
Para instalar `pymongo`, haremos uso de PIP, por lo que sólo hará falta instalarlo corriendo un comando en la terminal.

```
$ pip install pymongo
```

Para comprobar que `pymongo` ha sido correctamente instalado, deberíamos poder importarlo sin problemas:

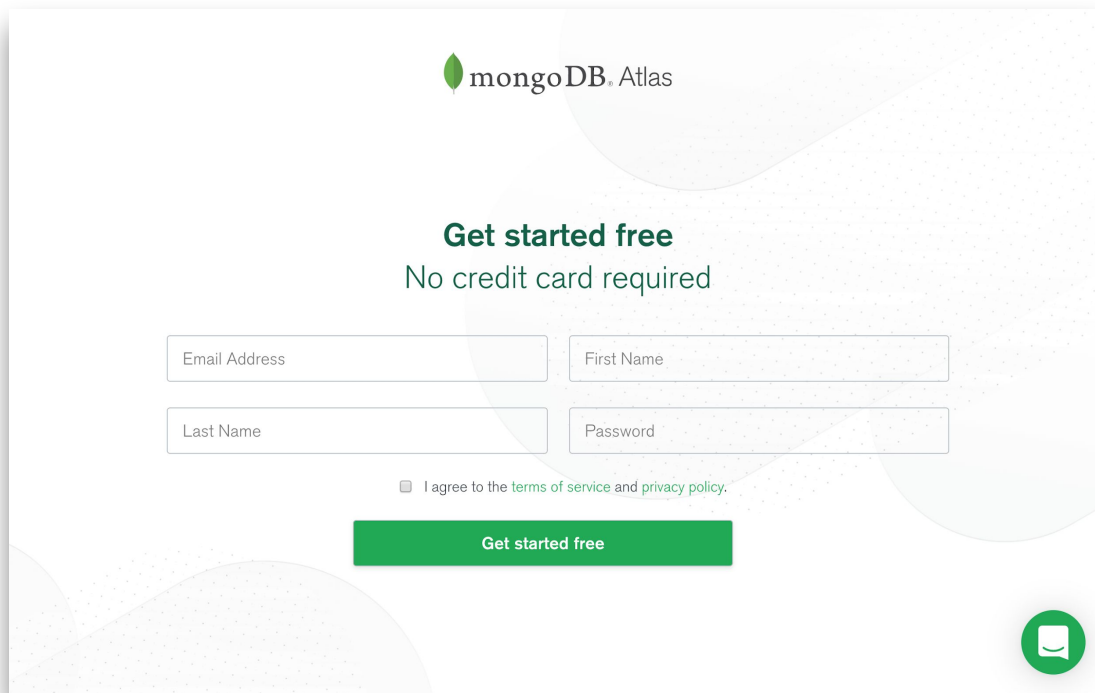
```
$ python
Python 3.7.3 (default, Mar 27 2019, 16:54:48)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import pymongo
```


Creación de cuenta en MongoDB Atlas



<https://www.mongodb.com/cloud>

Creación de cuenta en MongoDB Atlas

The image shows the MongoDB Atlas sign-up page. At the top, the MongoDB Atlas logo is displayed. Below it, the text "Get started free" is shown in a bold, dark green font, followed by "No credit card required" in a smaller, regular font. There are four input fields arranged in a 2x2 grid: "Email Address", "First Name", "Last Name", and "Password". Below these fields is a checkbox with the text "I agree to the terms of service and privacy policy." A large green button labeled "Get started free" is positioned below the checkbox. In the bottom right corner, there is a green circular icon with a white speech bubble inside.

mongoDB Atlas

Get started free
No credit card required

Email Address First Name

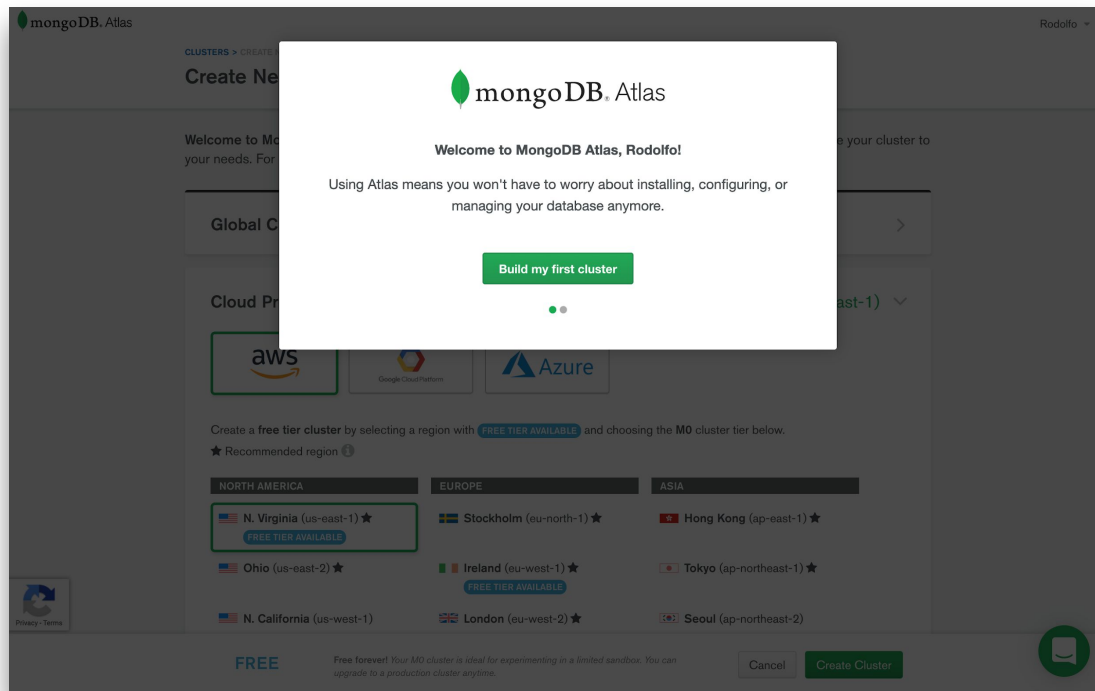
Last Name Password

☐ I agree to the [terms of service](#) and [privacy policy](#).

Get started free

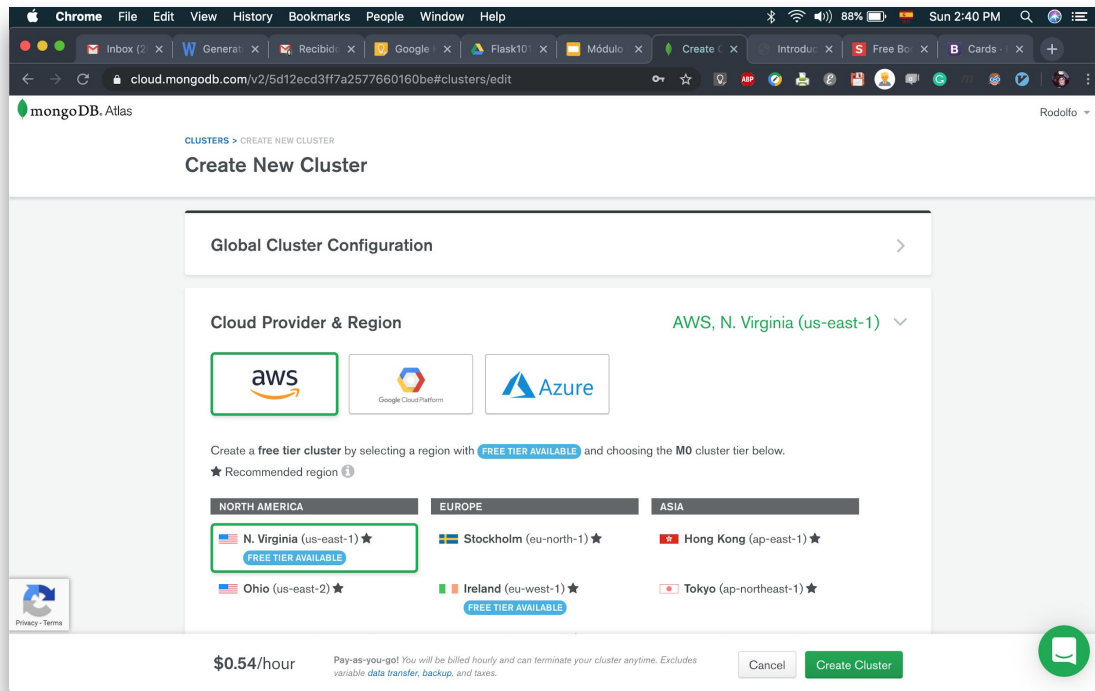
<https://www.mongodb.com/cloud>

Creación de cuenta en MongoDB Atlas



<https://www.mongodb.com/cloud>

Creación de clúster en MongoDB Atlas



<https://www.mongodb.com/cloud>

Accesos y usuarios en MongoDB Atlas

mongoDB, Atlas All Clusters

CONTEXT
pase-de-lista

ATLAS
Clusters
Data Lake BETA

SECURITY
Database Access
Network Access
Advanced

PROJECT
Access Management
Activity Feed
Alerts
Settings

SERVICES
Charts
Stitch
Triggers

HELP
Docs
Support

Add New User

SCRAM Authentication
SCRAM is MongoDB's default authentication method.

Enter username
e.g. new-user_31

Enter password SHOW

Autogenerate Secure Password

User Privileges

Atlas admin

Read and write to any database

Only read any database

Select Custom Role

Add Default Privileges

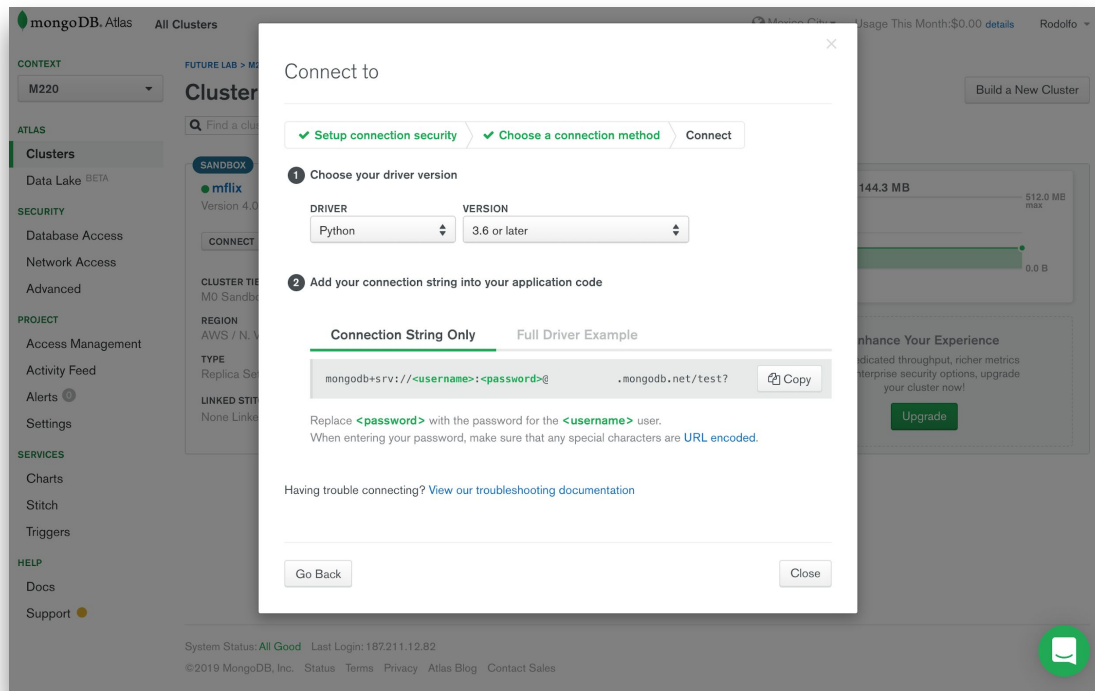
☐ Save as temporary user

Cancel Add User

System Status: All Good Last Login: 187.211.12.82
©2019 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

<https://www.mongodb.com/cloud>

Conexión a MongoDB Atlas



<https://www.mongodb.com/cloud>

Introducción a pymongo – Conexión a DB

Crearemos un script llamado `dbutils.py` donde trabajaremos todo lo relacionado a nuestra DB:

```
from pymongo import MongoClient

# Access to remote cluster:
MONGO_URI = "mongodb+srv://<user>:<pass>@<app-url>.mongodb.net/test"

# Connect to MongoClient:
client = MongoClient(MONGO_URI)
database = client['<database-key>']
collection = database['<collection-key>']
```

CRUD con pymongo – CREATE (insert)

En nuestro script llamado `dbutils.py` trabajaremos todo lo relacionado a nuestra DB:

```
# ...

# Once that we're connected and collection is loaded:
def db_insert_user(collection, user):
    """Function to insert a new user in collection."""

    return collection.insert(user)

# ...
```


CRUD con pymongo – READ (**find**)

En nuestro script llamado `dbutils.py` trabajaremos todo lo relacionado a nuestra DB:

```
# ...

# Once that we're connected and collection is loaded:
def db_fetch_all(collection, conditions={}):
    """Function to fetch content in collection."""

    cursor = collection.find(conditions)
    return cursor

# ...
```

CRUD con pymongo – UPDATE (update)

En nuestro script llamado `dbutils.py` trabajaremos todo lo relacionado a nuestra DB:

```
# ...

# Once that we're connected and collection is loaded:
def db_update_user(collection, user, data):
    """Function to update user content in collection."""

    collection.update_one(user, {'$set': data})
    return cursor

# ...
```

CRUD con pymongo – DELETE (delete)

En nuestro script llamado `dbutils.py` trabajaremos todo lo relacionado a nuestra DB:

```
# ...

# Once that we're connected and collection is loaded:
def db_clean(collection):
    """Function to delete all users in users collection."""

    return collection.delete_many({})

# ...
```

Tiempo de:

Integración a nuestra web app...

Siguiente sesión...

Módulo 6:

Deployment