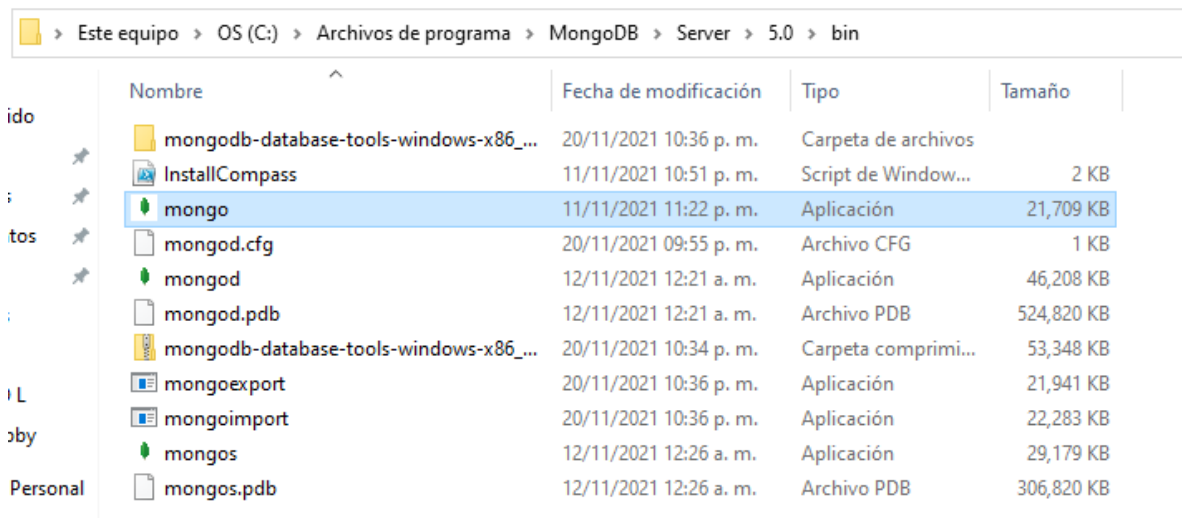


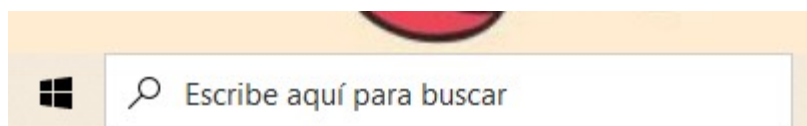
Una vez instalado MongoDB en Windows, para ejecutar los comandos de la práctica, debemos localizar el directorio donde se ha instalado mongod. En los sistemas Windows esto normalmente es en la siguiente ruta:

C:\Archivos de programa\MongoDB\Server\5.0\bin>

Es posible que, dependiendo de la versión de mongod que hayamos descargado e instalado, la ruta sea un poco diferente, pero a fin de cuentas estamos buscando la ruta donde se encuentra el archivo mongo. En la siguiente imagen lo podemos ver seleccionado y, en la parte superior podemos ver la ruta donde se encuentra.



Lo que sigue es abrir una terminal de cmd. Para ello, escribimos las letras cmd en la barra de búsqueda de Windows.



Una vez hecho lo anterior, se abrirá una ventana como la que se muestra a continuación.

```
C:\> Símbolo del sistema
Microsoft Windows [Versión 10.0.19043.1586]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jimen>
```

En esa ventana tendremos que teclear la instrucción **cd..** seguida de un **enter** hasta llegar al inicio de la ruta, es decir c:\

```
C:\> Símbolo del sistema
Microsoft Windows [Versión 10.0.19043.1586]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jimen>cd..

C:\Users>cd..

C:\>
```

Cuando lleguemos a la raíz de los directorios, comenzaremos a escribir la ruta donde se encuentra el ejecutable de mongo, y para ello utilizaremos de nuevo la instrucción **cd** seguida del nombre del directorio y un enter. En este caso será algo como lo que se muestra a continuación.

```
C:\>cd "Archivos de programa"

C:\Archivos de programa>cd MongoDB

C:\Archivos de programa\MongoDB>cd Server

C:\Archivos de programa\MongoDB\Server>cd 5.0

C:\Archivos de programa\MongoDB\Server\5.0>cd bin

C:\Archivos de programa\MongoDB\Server\5.0\bin>
```

Ahora sí, estamos listos para ejecutar las instrucciones de la práctica, y lo primero será iniciar el servidor de mongodb con la instrucción mongo seguida de un enter en la línea de comandos del cmd.

```
C:\Archivos de programa\MongoDB\Server\5.0\bin>mongo
MongoDB shell version v5.0.4
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("d3a8dbd5-fe88-40cc-88b4-439554ffd8d8") }
MongoDB server version: 5.0.4
=====
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility. The "mongo" shell has been deprecated and will be removed in
an upcoming release.
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
=====
---
The server generated these startup warnings when booting:
  2022-03-31T15:28:20.192-06:00: Access control is not enabled for the database. Read and write access to data and
configuration is unrestricted
---
---
  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()
---
>
```

1.- Mostrar las bases de datos

show databases

```
---
> show databases
admin          0.000GB
config         0.000GB
local          0.000GB
```

2.- Crear una nueva base de datos

use vehiculos

```
> use vehiculos
switched to db vehiculos
>
```

3.- Crear una colección dentro de la base de datos que acabamos de crear

db.createCollection("vehiculo", {capped:true, size:6142800, max:10000})

```
> db.createCollection("vehiculo", {capped:true, size:6142800, max:10000})
{ "ok" : 1 }
>
```

4.- Insertar un registro en la base de datos

db.vehiculos.insertOne({"Marca": "Volkswagen", "Modelo": 2018, "Color": "Rojo", "Cilindros": 4, "Combustible": "Gasolina"})

```
> db.vehiculos.insertOne({"Marca": "Volkswagen", "Modelo": 2018, "Color": "Rojo", "Cilindros": 4, "Combustible": "Gasolina"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("62576d2b057cda0554117bb7")
}
```

5.- Buscar un registro en la base de datos

db.vehiculos.find({"Marca":"Volkswagen"})

```
> db.vehiculos.find({"Marca":"Volkswagen"})
{ "_id" : ObjectId("62576d2b057cda0554117bb7"), "Marca" : "Volkswagen", "Modelo" : 2018, "Color" : "Rojo", "Cilindros" : 4, "Combustible" : "Gasolina" }
>
```

6.- Actualizar los valores de los registros en la base de datos. En este caso se está cambiando la marca del vehículo

db.vehiculos.update({"Marca":"Volkswagen"}, {\$set:{"Marca":"Audi"}})

Si usamos esta instrucción solo se modifica el primer valor que se encuentre en la base de datos que cumpla con esa condición. Para modificar todos los que cumplan con la condición deberemos escribir la instrucción de la siguiente manera

Db.vehiculos.update({"Marca":"Volkswagen"}, {\$set: {"Marca":"Audi"}}, {multi:true})

```
> db.vehiculos.update({"Marca":"Volkswagen"}, {$set:{"Marca":"Audi"}}, {multi:true})
WriteResult({ "nMatched" : 3, "nUpserted" : 0, "nModified" : 3 })
>
```

7.- Eliminar todos los registros de la base de datos que cumplan con una condición, por ejemplo, todos los de color rojo

db.vehiculos.remove({"Color":"Rojo"})

```
> db.vehiculos.remove({"Color":"Rojo"})
WriteResult({ "nRemoved" : 2 })
>
```

8.- Eliminar solamente un registro, el primero que encuentre, que cumpla con la condición establecida

```
db.vehiculos.remove({"Color":"Amarillo"}, 1)
```

```
> db.vehiculos.remove({"Color":"Amarillo"}, 1)
WriteResult({ "nRemoved" : 1 })
>
```

9.- Salir de la consola de MongoDB

exit

```
> exit
bye
C:\Archivos de programa\MongoDB\Server\5.0\bin>
```