

PRUEBAS UNITARIAS TodoCLI.py

En este documento se reflejan las pruebas unitarias realizadas sobre la aplicación de tipo CLI "TodoCLI.py".

En la siguiente tabla mostraremos las distintas pruebas que se van a realizar, junto con el proceso que debe seguir la aplicación y el resultado final.

Nº	Prueba	Proceso a seguir	Resultado final esperado
1	> python TodoCLI.py	--	Function not covered Consult the manual
2	> python TodoCLI.py -POST	- app : Solicita el Todo - user : aporta un Todo alfanúmero	Todo almacenado en la base de Datos
3	> python TodoCLI.py -POST	- app : Solicita el Todo - user : aporta un Todo null	Todo almacenado en la base de Datos
4	> python TodoCLI.py -GET	--	Muestra los datos almacenados en formato tabla
5	> python TodoCLI.py -GET txt	--	Muestra los datos almacenados en formato txt
6	> python TodoCLI.py -GET json	--	Muestra los datos almacenados en formato json
7	> python TodoCLI.py -GET table	--	Muestra los datos almacenados en formato tabla
8	> python TodoCLI.py -GET gfhf	--	Unknown format, applying default format → table Muestra los datos almacenados en formato tabla

Nº	Prueba	Proceso a seguir	Resultado final esperado
9	> python TodoCLI.py -DELETE	<ul style="list-style-type: none"> - app: Muestra los datos almacenados en formato txt - app: Solicita el content del Todo a eliminar - user: introduce un content existente 	El Todo seleccionado es eliminado de la Base de Datos
10	> python TodoCLI.py -DELETE	<ul style="list-style-type: none"> - app: Muestra los datos almacenados en formato txt - app: Solicita el content del Todo a eliminar - user: Introduce un content no existente 	La Base de Datos no se alterada
11	> python TodoCLI.py -TEST	<ul style="list-style-type: none"> - app: genera un número entre 10 y 100 aleatorio de Todos - app: muestra los Todos almacenado en formato txt - app: muestra los Todos almacenado en formato json - app: muestra los Todos almacenado en formato json - app: elimina todos los Todos existentes en la base de datos 	Todos los datos de la colección de la base de datos han sido eliminados

A continuación analizaremos cada una de las pruebas realizadas

PRUEBA Nº 1

> python TodoCLI.py	--	Function not covered Consult the manual
---------------------	----	--

```
PS C:\Aarón\node-todo-master> python TodoCLI.py
Function not covered
Consult the manual
```

PRUEBA N° 2

```
> python TodoCLI.py -POST
```

- **app**: Solicita el Todo
- **user**: aporta un Todo alfanúmero

Todo almacenado en la base de Datos

The screenshot shows a terminal window on the left with the command `python TodoCLI.py -POST` and its output: `Give me the Todo you want to POST`, `Hola Mundo123#`, and a prompt. On the right, the MongoDB Compass interface shows the 'Documents' tab for 'TodoDB.TodoS'. It displays one document with the following structure:

```
{
  "_id": ObjectId('640b9c1eda67be31f0497c7d'),
  "content": "Hola Mundo123#"
}
```

PRUEBA N° 3

```
> python TodoCLI.py -POST
```

- **app**: Solicita el Todo
- **user**: aporta un Todo null

Todo almacenado en la base de Datos

The screenshot shows a terminal window on the left with the command `python TodoCLI.py -POST` and its output: `Give me the Todo you want to POST` and a prompt. On the right, the MongoDB Compass interface shows the 'Documents' tab for 'TodoDB.TodoS'. It displays two documents:

```
{
  "_id": ObjectId('640b9c1eda67be31f0497c7d'),
  "content": "Hola Mundo123#"
},
{
  "_id": ObjectId('640b9ca3137215008e1d7f1e'),
  "content": ""
}
```

PRUEBA N° 4

```
> python TodoCLI.py -GET
```

--

Muestra los datos almacenados en formato tabla

```
PS C:\Aarón\node-todo-master> python TodoCLI.py -GET
_id                                     | content
-----|-----
640b9c1eda67be31f0497c7d | Hola Mundo123#
640b9ca3137215008e1d7f1e |
```

PRUEBA Nº 5

> python TodoCLI.py -GET txt	--	Muestra los datos almacenados en formato txt
------------------------------	----	--

```
PS C:\Aarón\node-todo-master> python TodoCLI.py -GET txt
{'_id': ObjectId('640b9c1eda67be31f0497c7d'), 'content': 'Hola Mundo123#'}
{'_id': ObjectId('640b9ca3137215008e1d7f1e'), 'content': ''}
```

PRUEBA Nº 6

> python TodoCLI.py -GET json	--	Muestra los datos almacenados en formato json
-------------------------------	----	---

```
PS C:\Aarón\node-todo-master> python TodoCLI.py -GET json
[
  {
    "_id": {
      "$oid": "640b9c1eda67be31f0497c7d"
    },
    "content": "Hola Mundo123#"
  },
  {
    "_id": {
      "$oid": "640b9ca3137215008e1d7f1e"
    },
    "content": ""
  }
]
```

PRUEBA Nº 7

> python TodoCLI.py -GET table	--	Muestra los datos almacenados en formato tabla
--------------------------------	----	--

```
PS C:\Aarón\node-todo-master> python TodoCLI.py -GET table
_id | content
-----|-----
640b9c1eda67be31f0497c7d | Hola Mundo123#
640b9ca3137215008e1d7f1e |
```

PRUEBA N° 8

<pre>> python TodoCLI.py -GET gfhf</pre>	<pre>--</pre>	<p>Unknown format, applying default format → table</p> <p>Muestra los datos almacenados en formato tabla</p>
---	---------------	--

```
PS C:\Aarón\node-todo-master> python TodoCLI.py -GET gfhf

Unknown format, applying default format --> table

_id                                     | content
-----|-----
640b9c1eda67be31f0497c7d             | Hola Mundo123#
640b9ca3137215008e1d7f1e             |
```

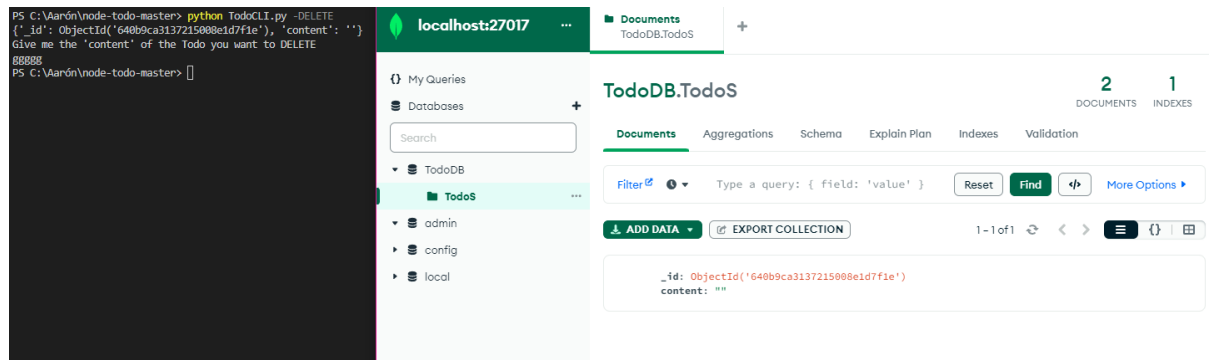
PRUEBA N° 9

<pre>> python TodoCLI.py -DELETE</pre>	<p>- app: Muestra los datos almacenados en formato txt</p> <p>- app: Solicita el content del Todo a eliminar</p> <p>- user: introduce un content existente</p>	<p>El Todo seleccionado es eliminado de la Base de Datos</p>
---	---	--

The screenshot shows a terminal window on the left and a web browser on the right. The terminal displays the command `python TodoCLI.py -DELETE` and the output of the application, which prompts the user to provide the content of the todo to delete. The user enters 'Hola Mundo123#'. The web browser shows the 'TodoDB.TODOs' interface with a list of documents. The document with `_id: ObjectId('640b9ca3137215008e1d7f1e')` and `content: ''` is highlighted, indicating it has been selected for deletion.

PRUEBA N°10

<pre>> python TodoCLI.py -DELETE</pre>	<p>- app: Muestra los datos almacenados en formato txt</p> <p>- app: Solicita el content del Todo a eliminar</p> <p>- user: Introduce un content no existente</p>	<p>La Base de Datos no se alterada</p>
---	--	--



PRUEBA N° 11

<p>> python TodoCLI.py -TEST</p>	<ul style="list-style-type: none"> - app: genera un número entre 10 y 100 aleatorio de Todos - app: muestra los Todos almacenado en formato txt - app: muestra los Todos almacenado en formato json - app: muestra los Todos almacenado en formato json - app: elimina todos los Todos existentes en la base de datos 	<p>Todos los datos de la colección de la base de datos han sido eliminados</p>
-------------------------------------	---	--

The screenshot shows a terminal window on the left with the command `python TodoCLI.py -TEST` and its output, which lists 20 JSON documents. The MongoDB web interface on the right shows the 'Todos' collection in the 'TodoDB' database, with 2 documents and 1 index. The collection is empty, displaying 'This collection has no data'.