



POSTMAN

Testing Postman CRUD MEVN 2

Agregar un objeto JSON a la base de datos

Creamos un nuevo “testing” y generamos una solicitud HTTP POST, de la forma:

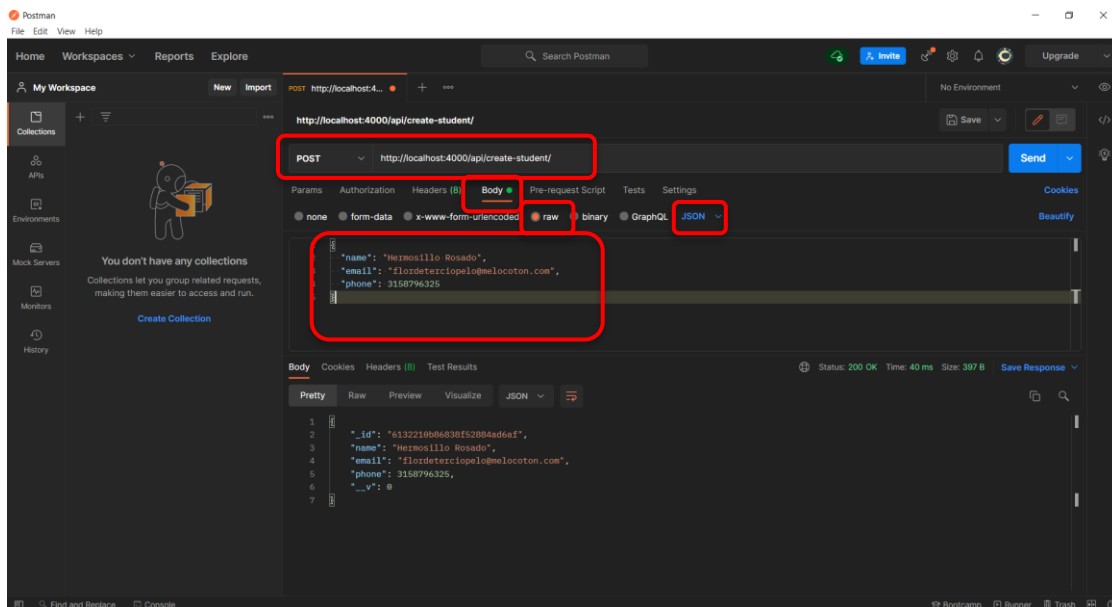
Solicitud: POST.

Dirección: `http://localhost:4000/api/create-student`

Ahora, creamos el siguiente registro para enviar la solicitud

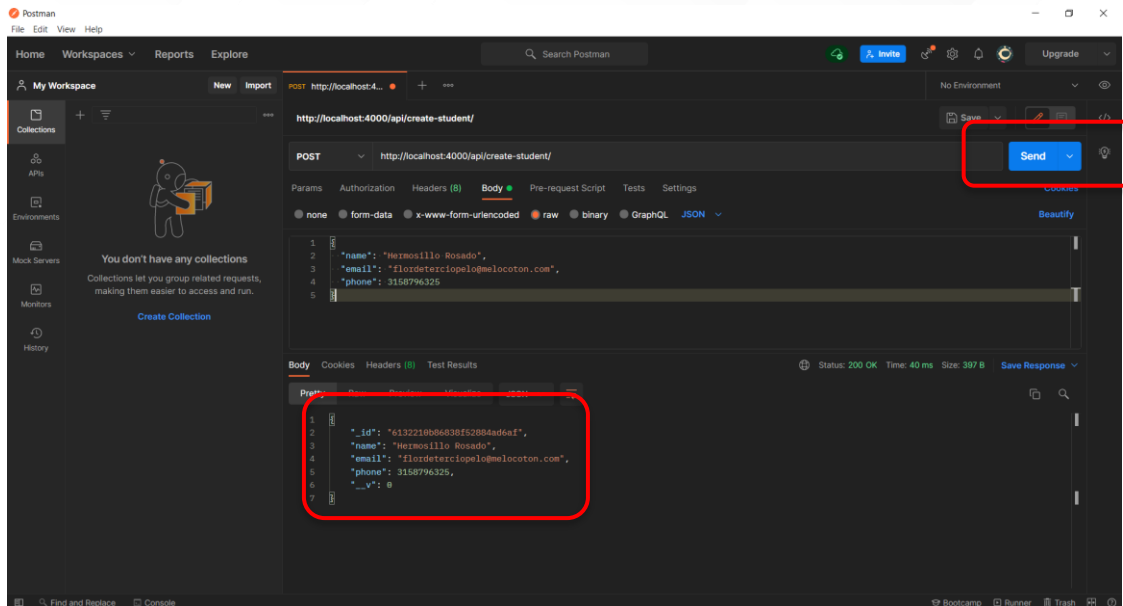
```
{  
  "name": "Hermosillo Rosado",  
  "email": "flordeterciopelo@melocoton.com",  
  "phone": 3158796325  
}
```

Tal como se aprecia en la imagen, registramos los datos de la siguiente forma:





Una vez agregado todo el registro, se presiona el botón **Send**



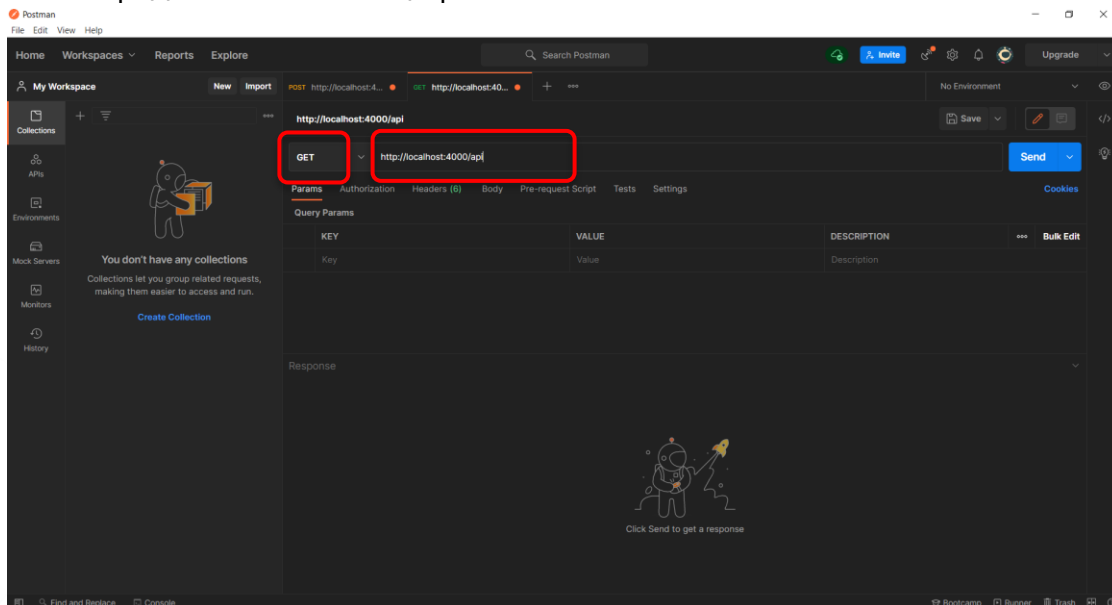
Como resultado, nos devuelve un objeto JSON registrado.

Recuperar la lista de registros

Después de haber agregado el primer objeto JSON a nuestra base de datos, podemos recuperar la lista de registros que esté disponibles enviando una solicitud HTTP GET a la siguiente ruta:

Solicitud: GET

Dirección: <http://localhost:4000/api>



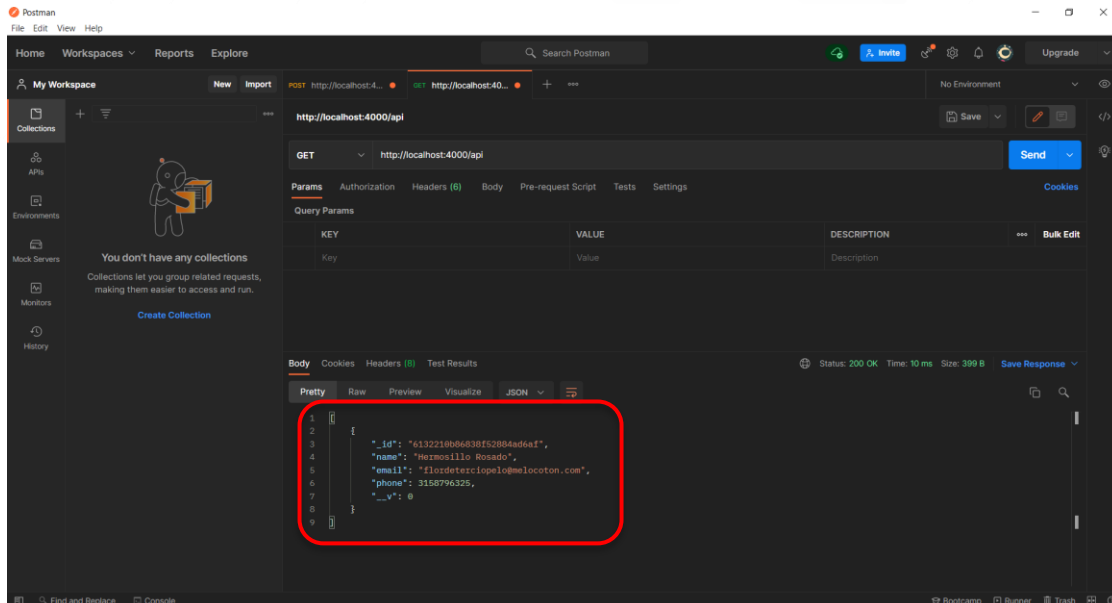
Se presiona el botón **Send**.



El futuro digital
es de todos

MinTIC

Como se puede apreciar, se recuperó el objeto JSON desde la bd. En la respuesta, puede ver que se ha asignado una identificación automáticamente a cada registro.

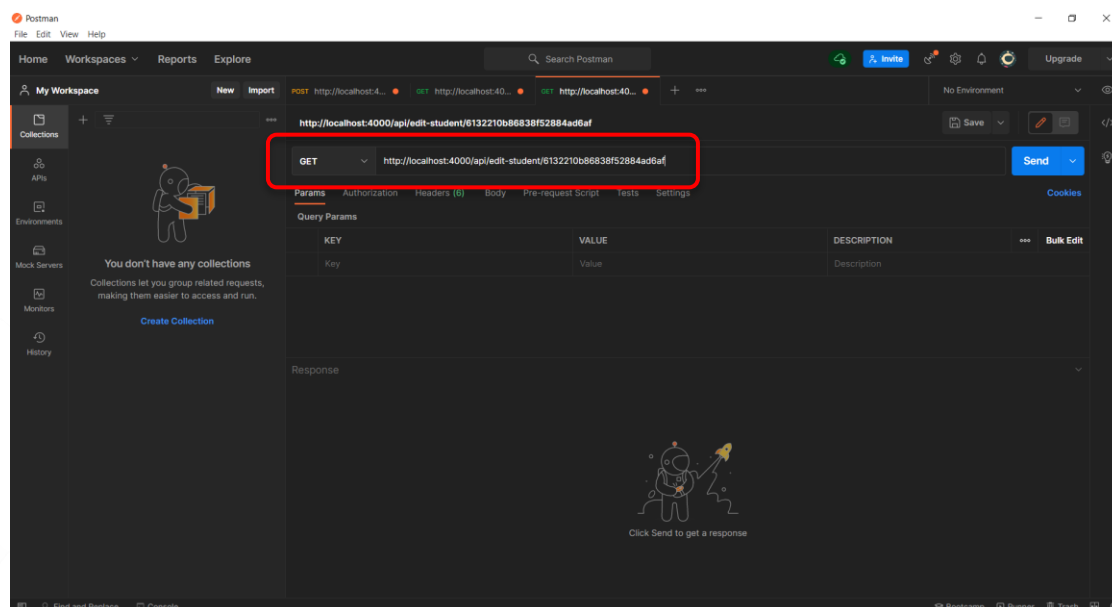


Recuperar un objeto JSON de la base de datos

A continuación, usemos la identificación para probar la ruta `/api/edit-student/` a fin de recuperar el objeto JSON de la base de datos, en función de su identificación:

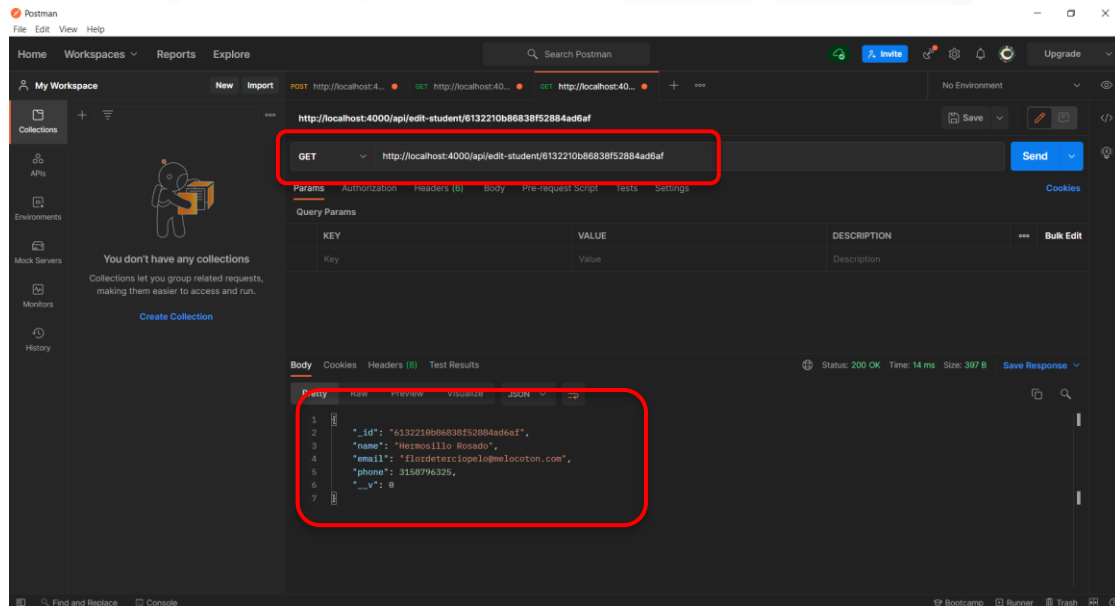
Solicitud: GET.

Dirección: `http://localhost:4000/api/edit-student/6132210b86838f52884ad6af`





Como se esperaba, se devuelve el mismo elemento de tarea pendiente en formato JSON.



Actualizar un objeto JSON en la base de datos:

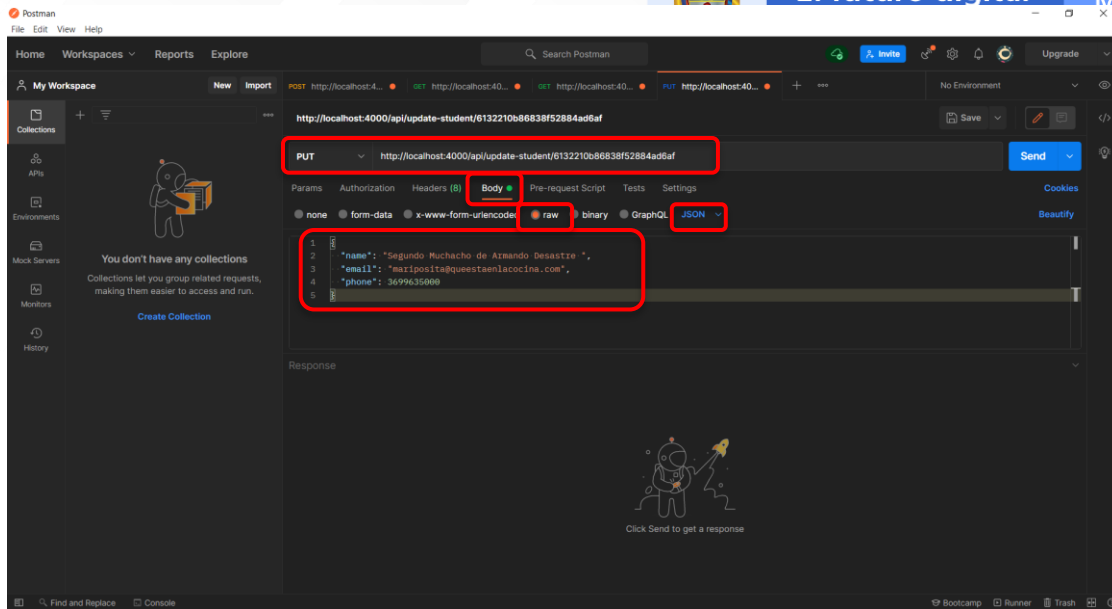
A continuación, intentemos actualizar el objeto JSON enviando una solicitud PUT a la ruta /api/update-student/

Solicitud: PUT

Dirección: <http://localhost:4000/api/update-student/6132210b86838f52884ad6af>

Ahora, tomaremos la siguiente actualización del objeto JSON:

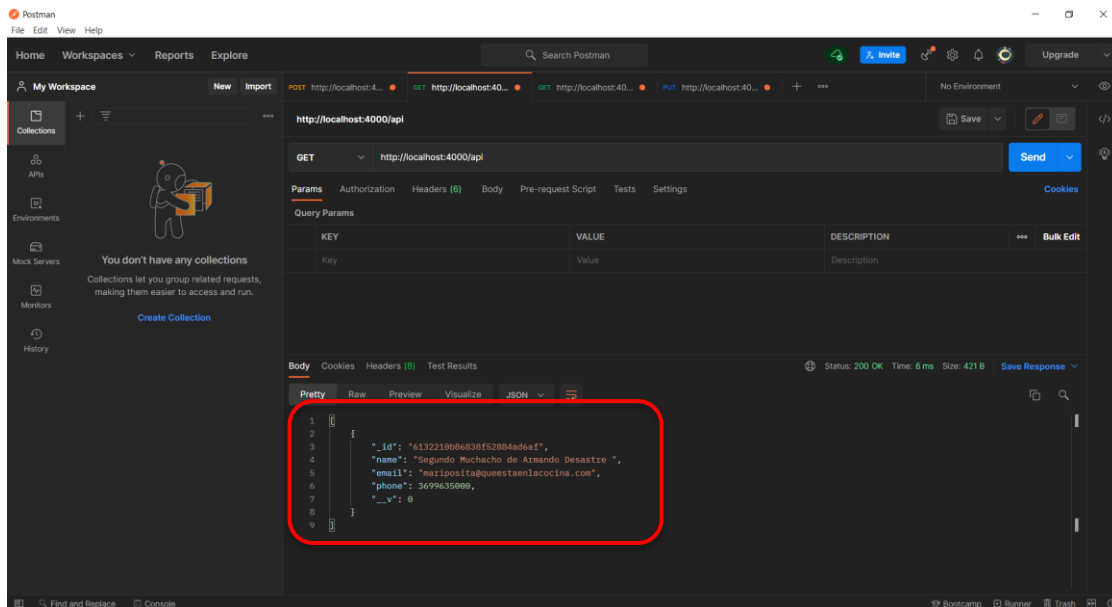
```
{  
  "name": "Segundo Muchacho de Armando Desastre ",  
  "email": "mariposita@queestaenlacocina.com",  
  "phone": 3699635000  
}
```



Ahora, recuperamos de nuevo la lista de registros para observar la actualización, de la forma:

Solicitud: GET

Dirección: <http://localhost:4000/api>



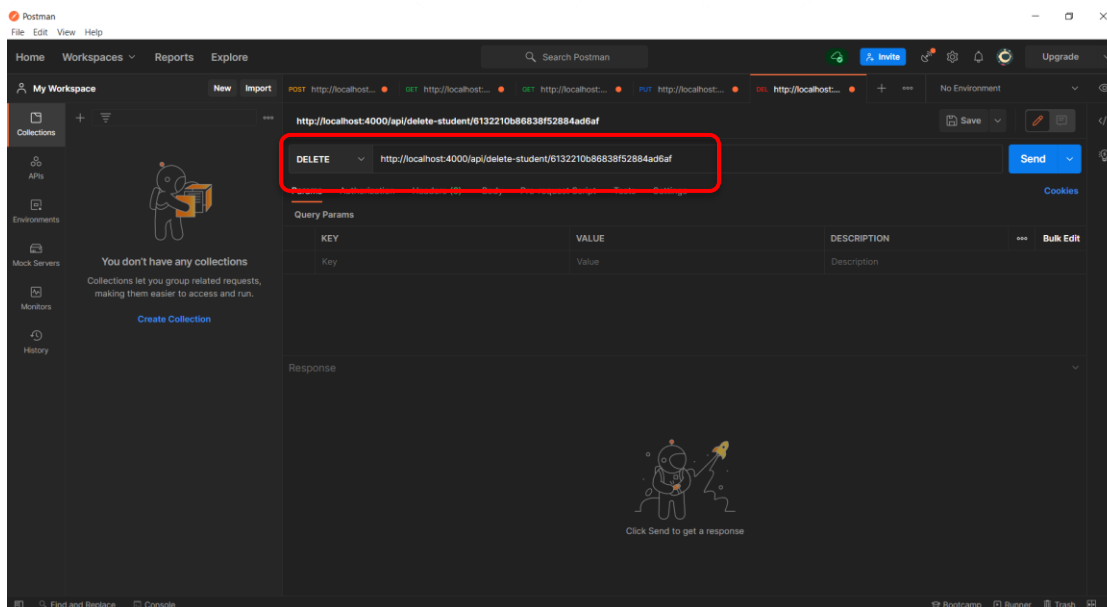


Borrado de un objeto JSON de la base de datos:

A continuación, evaluaremos la ruta `/api/delete-student/` para eliminar objetos JSON de la base de datos, de la forma:

Solicitud: DELETE

Dirección: `http://localhost:4000/api/delete-student/6132210b86838f52884ad6af`



Finalmente, recuperamos de nuevo la lista de registros para observar que el objeto JSON fue borrado satisfactoriamente de la base de datos, de la forma:

Solicitud: GET

Dirección: `http://localhost:4000/api`

