

Please sign in

**Sign in**

- Para seguridad temporal:
  - `spring.security.user.name=admin`
  - `spring.security.user.password=admin123`

The screenshot shows the Postman interface with a collection named "crud". A GET request is made to `http://localhost:8080/api/v1/products`. The response status is 200 OK, with a response time of 189 ms and a body size of 166 B. The response body is empty, indicated by the message "This request does not have a body".

The screenshot shows the Postman interface with a collection named "crud". A PUT request is made to `http://localhost:8080/api/v1/products`. The response status is 401 Unauthorized, with a response time of 16 ms and a body size of 342 B. The response body is empty.

The screenshot shows the Postman interface with a collection named "crud". A POST request is made to `http://localhost:8080/api/v1/products?continue`. The response status is 400 Bad Request, with a response time of 8 ms and a body size of 7.5 KB. The response body contains the following JSON:

```
timestamp: 2025-11-10T05:55:03.185+00:00
status: 400
error: Bad Request
org.springframework.http.converter.HttpMessageNotReadableException: JSON parse error: Cannot deserialize value of type `com.example.demo.product.Product` from ArrayToken.START_ARRAY at org.springframework.http.converter.json.AbstractJackson2HttpMessageConverter.readJavaType(AbstractJackson2HttpMessageConverter.java:356) at org.springframework.http.converter.json.AbstractJackson2HttpMessageConverter.read(AbstractJackson2HttpMessageConverter.java:356) at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerMapping.handleInternal(RequestMappingHandlerMapping.java:200) at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:172) at org.springframework.web.servlet.handler.HandlerMethodArgumentResolverComposite.resolveArgument(HandlerMethodArgumentResolverComposite.java:122) at
```

The screenshot shows the Postman application interface. In the top navigation bar, there are tabs for Home, Workspaces (with a dropdown menu showing 'JOSUE DAVID BARRERA SANTOS's...'), API Network, and a search bar. On the right side of the header, there are buttons for Invite, Upgrade, and Share.

The left sidebar contains sections for Collections (with a trash icon), Environments (with a gear icon), Flows (with a flowchart icon), and History (with a clock icon). Below these are buttons for New and Import.

The main workspace is titled 'crud / Eliminar' and shows a DELETE request. The URL is set to `http://localhost:8080/api/v1/products/1`. The Headers tab shows a count of 7. The Body tab is selected and has the 'raw' option checked. The response area below says 'Click Send to get a response'.

At the bottom of the interface, there are buttons for Online, Find and replace, Console, Runner, Start Proxy, Cookies, Vault, Trash, and Help.

- GET /api/v1/products
  - Devuelve lista de todos los productos.
  - Respuesta 200:  
[{"id":1,"name":"...","price":0.0,"description":"..."}]
- POST /api/v1/products
  - Crea producto.
  - Request body:  
{ "name":"Pizza", "price":55.0, "description":"..." }
  - Respuesta 200: vacío o mensaje creado.
  - 409 o 400 si nombre repetido.
- PUT /api/v1/products/{id}
  - Actualiza producto por id.
  - Request body igual al de POST.
  - Respuesta 200: vacío o actualizado.
- DELETE /api/v1/products/{id}
  - Elimina producto.
  - Respuesta 200 si eliminado, 404 si no existe.