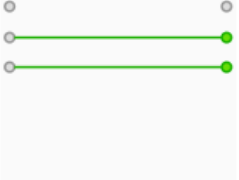


request.body request  
message.body prepare\_request  
message.body prepare\_response  
message.body response

The screenshot displays the IBM API Connect interface. The top section shows a REST client configuration for a POST request to the endpoint `https://api.us-south.apiconnect.appdomain.cloud/bank1julioparicom-dev/sb/demo-request-response-call/reqres-register`. The request body is a JSON object: `{ "name": "juliopari20", "job": "architect20" }`. Below this, the response body is shown in JSON format: `{ "name": "juliopari20", "job": "architect20", "id": "6883", "createdAt": "2021-07-13T13:42:49.319Z" }`. The status of the request is `201 Created`.

The bottom section shows a flow mapping diagram. A `switch` node branches into three paths based on the request method: `POST /register`, `POST /reqres-register`, and `POST /generar-ca`. Each path contains a sequence of `map`, `invoke reqres` (or `invoke bc`), and `map` nodes. The `POST /reqres-register` path is highlighted with a red box.

**map** 🗑️ <> + ✕

Input	Map	Output
<pre>- request {   name string   job string   add property... }</pre>		<pre>- prepare_request {   name string   job string   add property... }</pre>

**map** 🗑️ <> + ✕

Context variable: request.body | Name: request

Content type: application/json | Definition: Inline schema

**+ input** + parameters for operation... Done

**map** 🗑️ <> + ✕

Context variable: message.body | Name: prepare\_request

Content type: application/json | Definition: Inline schema

**+ output** + outputs for operation... Done

The screenshot displays the configuration interface for an API endpoint in IBM API Connect. The top section is for the endpoint configuration, and the bottom section is for the mapping logic.

**Endpoint Configuration:**

- URL:** `https://reqres.in/api/users` (highlighted with a red box)
- Access URL through Secure Gateway
- TECH PREVIEW:** Access the URL through a Secure Gateway. The gateway is set when the API is moved from staged state to published state. Note: This feature is experimental.
- TLS Profile:** (Empty field)
- Timeout:** 60 (Time to wait before a reply back from the endpoint. Default is 60 seconds.)
- Username:** (Empty field)
- Password:** (Empty field)
- HTTP Method:** POST (highlighted with a red box)
- Compression
- Cache Type:** Protocol

**Mapping Diagram:**

The mapping diagram shows a **map** operation. It has an **Input** section and an **Output** section.

**Input:**

```
- prepare_response {  
  name string  
  job string  
  id string  
  createdAt string  
  add property...  
}
```






**Map:**

Four green lines connect the input fields to the output fields, indicating a 1:1 mapping.

**Output:**

```
- response {  
  name string  
  job string  
  id string  
  createdAt string  
  add property...  
}
```



---

 **map**    

---






Context variable `message.body` Name `prepare_response`

---

Content type `application/json` Definition \* `Inline schema`  

**+ input** + parameters for operation... **Done**



---

 **map**    

---

Context variable `message.body` Name `response`

---

Content type `application/json` Definition \* `Inline schema`  

**+ output** + outputs for operation... **Done**

---

## Julio Pari (IT Architect IBM)



Si te ha interesado este artículo y deseas un apoyo o asesoría en algún requerimiento, envíame un mensaje a: (info@juliopari.com) o sino a través de LinkedIn: <https://www.linkedin.com/in/juliopari/>

---