

DOCUMENTACIÓN DE PRUEBAS DE SOLICITUDES:

ID solicitud:	004
URL:	https://jsonplaceholder.typicode.com/comments?postId=5
Método:	GET
Objetivo:	Verificar que la API devuelva correctamente todas las fotos correspondientes al álbum con ID 2, incluyendo los campos <code>title</code> y <code>url</code> de cada foto.
Detalle del procedimiento:	Se envía una solicitud GET al endpoint <code>/albums/2/photos</code> . La API responde con un arreglo de objetos JSON, donde cada objeto contiene detalles de una foto, incluyendo el título y la url correspondientes al álbum 2.
Conclusión:	La API responde exitosamente con un código 200 OK y devuelve correctamente las fotos del álbum 2, permitiendo visualizar cada foto junto con su título correspondiente.
Captura:	Se expone debajo a continuación:

The screenshot displays the Postman application interface. The top navigation bar shows the current workspace 'Portfolio roman' and the API network. The left sidebar contains a 'Collections' panel with 'pruebas de API' and 'pruebas de API' sub-collections, and an 'Environments' panel. The main workspace shows a GET request to 'https://jsonplaceholder.typicode.com/albums/2/photos' with a status of '200 OK'. The response body is a JSON array of two photo objects, each containing 'albumId', 'id', 'title', 'url', and 'thumbnailUrl'. The bottom status bar shows the system clock as 08:40 p.m. on 05/06/2025.

```
1 [
2   {
3     "albumId": 2,
4     "id": 51,
5     "title": "non sunt voluptatem placeat consequuntur rem incididunt",
6     "url": "https://via.placeholder.com/600/8e973b",
7     "thumbnailUrl": "https://via.placeholder.com/150/8e973b"
8   },
9   {
10    "albumId": 2,
11    "id": 52,
12    "title": "eveniet pariatur quia nobis reiciendis laboriosam ea",
13    "url": "https://via.placeholder.com/600/121fad",
14    "thumbnailUrl": "https://via.placeholder.com/150/121fad"
15  }
16 ]
```

