



File ▾

Edit ▾

Insert ▾

Generate Server ▾

Generate Client ▾

About ▾

Sign in to API Hub

Try API Hub

JSONPlaceholder API 1.0.0 OAS 3.0

Documentación generada a partir de los archivos [.bru](#) de JSONPlaceholder de la Clase 008-009. JSONPlaceholder es una API REST gratuita utilizada para pruebas, demostraciones y prototipado rápido de aplicaciones.

[JSONPlaceholder - Website](#)

Servers

<https://jsonplaceholder.typicode.com> - Servidor principal de la API

Albums

Endpoints relacionados con álbumes de usuarios.



GET [/albums](#) Obtener todos los álbumes



Devuelve una lista completa de todos los álbumes disponibles.

Parameters

[Cancel](#)

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/albums' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/albums>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "userId": 1, "id": 1, "title": "quidem molestiae enim" }, { "userId": 1, "id": 2, "title": "sunt qui excepturi placeat culpa" }, {</pre>

Code	Details
	<pre>"userId": 1, "id": 3, "title": "omnis laborum odio" }, { "userId": 1, "id": 4, "title": "non esse culpa molestiae omnis sed optio" }, { "userId": 1.</pre>

[Download](#)

Response headers

```
cache-control: max-age=43200  
content-length: 2051  
content-type: application/json; charset=utf-8  
expires: -1  
pragma: no-cache
```

Responses

Code	Description	Links
200	<p>Lista de álbumes obtenida correctamente.</p> <p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p>	No links

POST /albums Crear un nuevo álbum ^

Crea un nuevo álbum asociado a un usuario.

Parameters

CancelReset

No parameters

Request body required

application/json

Edit Value Schema

```
{  
    "title": "Vacation",  
    "userId": 1  
}
```

//

Execute**Clear**

Responses

Curl

```
curl -X 'POST' \  
  'https://jsonplaceholder.typicode.com/albums' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
      "title": "Vacation",  
      "userId": 1  
  }'
```

Request URL

<https://jsonplaceholder.typicode.com/albums>

Server response

Code	Details
------	---------

201

Response body

```
{  
    "title": "Vacation",  
    "userId": 1,  
    "id": 101  
}
```

[Download](#)

Response headers

```
cache-control: no-cache  
content-length: 53  
content-type: application/json; charset=utf-8  
expires: -1  
location: https://jsonplaceholder.typicode.com/albums/101  
pragma: no-cache
```

Responses

Code	Description	Links
201	Álbum creado exitosamente. Media type application/json Controls Accept header. Example Value Schema <pre>{ "userId": 0, "id": 0, "title": "string" }</pre>	No links

GET /albums/{id} Obtener un álbum por ID ^

Devuelve la información de un álbum específico identificado por su ID.

Parameters

Cancel

Name	Description
id * required integer (path)	<input type="text" value="1"/>

[Execute](#)[Clear](#)

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/albums/1' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/albums/1>

Server response

Code	Details
------	---------

200

[Response body](#)

```
{  
  "userId": 1,  
  "id": 1,  
  "title": "quidem molestiae enim"  
}
```

[Download](#)

[Response headers](#)

```
cache-control: max-age=43200  
content-type: application/json; charset=utf-8  
expires: -1  
pragma: no-cache
```

Responses

Code	Description	Links
200	<p>Álbum encontrado exitosamente.</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "userId": 0, "id": 0, "title": "string" }</pre>	No links

Comments

Endpoints relacionados con los comentarios de las publicaciones.



GET /comments Obtener todos los comentarios



Devuelve una lista de todos los comentarios o los de una publicación específica si se indica **postId**.

Parameters

Cancel

Name	Description
postId integer <i>(query)</i>	ID de la publicación cuyos comentarios se desean obtener. 1

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/comments?postId=1' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/comments?postId=1>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "postId": 1, "id": 1, "name": "id labore ex et quam laborum", "body": "quia et suscipit sed optio cum occaecati cupiditate non provident similique sunt in culpa qui officia deserunt mollitia animi est harum quasi nihil quibusdam voluptatem sit eveniet ut et voluptates repudiandae sint et molestiae consequatur doloremque laudantium fugiat enim adipisci nesciunt neque nisi nulla }]</pre>

Code	Details	
	<pre>"email": "Eliseo@gardner.biz", "body": "laudantium enim quasi est quidem magnam voluptate ipsam eos\nntempora quo necessitatibus\nndolor quam autem quasi\nnreiciendis et nam sapiente accusantium" }, { "postId": 1, "id": 2, "name": "quo vero reiciendis velit similique earum", "email": "Jayne_Kuhic@sydney.com", "body": "est natus enim nihil est dolore omnis voluptatem numquam\\net omnis occaecati quod ullam at\\nvolutatem error expedita pariatur\\nnihil sint nostrum voluptatem reiciendis et" }, { "postId": 1, "id": 3, "name": "odio adipisci rerum aut animi", "email": "Nikita@garfield.biz", "body": "quia molestiae reprehenderit quasi aspernatur\\naut expedita occaecati aliquam eveniet lauda</pre>	
Response headers		
<pre>cache-control: max-age=43200 content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre>		
Responses		
Code	Description	Links
200	<p>Lista de comentarios obtenida correctamente.</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p>	<i>No links</i>
Example Value Schema		

Code	Description	Links
[{ "postId": 0, "id": 0, "name": "string", "email": "string", "body": "string" }]		

POST /comments Crear un nuevo comentario ^

Crea un nuevo comentario asociado a una publicación.

Parameters

CancelReset

No parameters

Request body required

application/json

[Edit Value](#) Schema

```
{  
  "name": "test",  
  "email": "test@example.com",  
  "body": "Nice!",  
  "postId": 1  
}
```

//

Execute**Clear**

Responses

Curl

```
curl -X 'POST' \  
  'https://jsonplaceholder.typicode.com/comments' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "name": "test",  
    "email": "test@example.com",  
    "body": "Nice!",  
    "postId": 1  
}'
```

Request URL

<https://jsonplaceholder.typicode.com/comments>

Server response

Code	Details
------	---------

201

Response body

```
{  
  "name": "test",  
  "email": "test@example.com",  
  "body": "Nice!",  
  "postId": 1,  
  "id": 501  
}
```

[Download](#)

Response headers

```
cache-control: no-cache  
content-length: 98  
content-type: application/json; charset=utf-8  
expires: -1  
location: https://jsonplaceholder.typicode.com/comments/501  
pragma: no-cache
```

Responses

Code	Description	Links
201	Comentario creado exitosamente.	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
  "postId": 0,  
  "id": 0,
```

Code	Description	Links
	<pre>"name": "string", "email": "string", "body": "string" }</pre>	

Photos Endpoints relacionados con fotos y galerías.



GET /photos Obtener todas las fotos



Devuelve una lista completa de fotos o las de un álbum específico si se indica **albumId**.

Parameters

[Cancel](#)

Name Description

albumId ID del álbum al que pertenecen las fotos.

integer

1

(query)

[Execute](#)

[Clear](#)

Responses

Curl

```
curl -X 'GET' \
'https://jsonplaceholder.typicode.com/photos?albumId=1' \
-H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/photos?albumId=1>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "albumId": 1, "id": 1, "title": "accusamus beatae ad facilis cum similique qui sunt", "url": "https://via.placeholder.com/600/92c952", "thumbnailUrl": "https://via.placeholder.com/150/92c952" }, { "albumId": 1, "id": 2, "title": "reprehenderit est deserunt velit ipsam", "url": "https://via.placeholder.com/600/771796", "thumbnailUrl": "https://via.placeholder.com/150/771796" }, { "albumId": 1, "id": 3, "title": "officia porro iure quia iusto qui ipsa ut modi", "url": "https://via.placeholder.com/600/24f355", "thumbnailUrl": "https://via.placeholder.com/150/24f355" }, { "albumId": 1, "id": 4, "title": "tempore et doloremque velit", "url": "https://via.placeholder.com/600/2171ac", "thumbnailUrl": "https://via.placeholder.com/150/2171ac" }]</pre>

Download ▾

Code	Details	
Response headers		
	<pre>cache-control: max-age=43200 content-length: 1820 content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre>	
Responses		
Code	Description	Links
200	<p>Lista de fotos obtenida correctamente.</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p>	<i>No links</i>
Example Value Schema		
<pre>[{ "albumId": 0, "id": 0, "title": "string", "url": "string", "thumbnailUrl": "string" }]</pre>		

Posts

Endpoints relacionados con publicaciones de usuarios.



GET **/posts** Obtener todas las publicaciones



Devuelve una lista de todas las publicaciones disponibles.

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/posts' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/posts>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "userId": 1, "id": 1, "title": "de "body": "s }]</pre>

Code	Details	
	<pre>"title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit", "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto" }, { "userId": 1, "id": 2, "title": "qui est esse", "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui ne que nisi nulla" }, { "userId": 1, "id": 3, "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut", "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut" }, </pre> <div style="text-align: right;">Download</div>	
Response headers		
<pre>cache-control: max-age=43200 content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre>		
Responses		
Code	Description	Links
200	<p>Lista de publicaciones obtenida correctamente.</p> <p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p>	No links
Example Value Schema		

Code	Description	Links
<pre>[{ "userId": 0, "id": 0, "title": "string", "body": "string" }]</pre>		

POST /posts Crear una nueva publicación ^

Crea una nueva publicación asociada a un usuario.

Parameters

[Cancel](#)

[Reset](#)

No parameters

Request body required

application/json

[Edit Value](#) Schema

```
{  
  "title": "foo",  
  "body": "bar",  
  "userId": 1  
}
```

//

Execute**Clear**

Responses

Curl

```
curl -X 'POST' \  
  'https://jsonplaceholder.typicode.com/posts' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "title": "foo",  
    "body": "bar",  
    "userId": 1  
  }'
```

Request URL

<https://jsonplaceholder.typicode.com/posts>

Server response

Code	Details
------	---------

201

Response body

```
{  
  "title": "foo",  
  "body": "bar",  
  "userId": 1,  
  "id": 101  
}
```

[Download](#)

Response headers

```
cache-control: no-cache  
content-length: 65  
content-type: application/json; charset=utf-8  
expires: -1  
location: https://jsonplaceholder.typicode.com/posts/101  
pragma: no-cache
```

Responses

Code	Description	Links
------	-------------	-------

201

Publicación creada exitosamente.

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
  "userId": 0,  
  "id": 0,  
  "title": "string",
```

Code	Description	Links
	<pre>"body": "string" }</pre>	

GET /posts/{id} Obtener una publicación por ID ^

Devuelve los datos de una publicación específica según su ID.

Parameters

[Cancel](#)

Name	Description
------	-------------

id * required integer (path)	<input type="text" value="1"/>
---	--------------------------------

[Execute](#)

[Clear](#)

Responses

Curl

```
curl -X 'GET' \  
  'https://jsonplaceholder.typicode.com/posts/1' \  
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/posts/1>

Server response

Code	Details	
200	Response body <pre>{\n "userId": 1,\n "id": 1,\n "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",\n "body": "quia et suscipit\\nsuscipit recusandae consequuntur expedita et cum\\nreprehenderit molestiae ut\\n ut quas totam\\nnostrum rerum est autem sunt rem eveniet architecto"\n}</pre> Download	
Response headers		
<code>cache-control: max-age=43200</code> <code>content-type: application/json; charset=utf-8</code> <code>expires: -1</code> <code>pragma: no-cache</code>		
Responses		
Code	Description	Links
200	Publicación encontrada exitosamente. Media type application/json Controls Accept header.	<i>No links</i>
Example Value Schema		
{ "userId": 0, "id": 0, "title": "string",		

Code	Description	Links
<pre>"body": "string" }</pre>		
PUT	/posts/{id} Reemplazar una publicación	^
Actualiza completamente una publicación existente mediante PUT.		
Parameters		Cancel Reset
Name	Description	
id * required integer (path)	<input type="text" value="1"/>	
Request body required		application/json
Edit Value Schema		

```
{  
  "id": 1,  
  "title": "foo",  
  "body": "bar",  
  "userId": 1  
}
```

//

Execute**Clear**

Responses

Curl

```
curl -X 'PUT' \  
  'https://jsonplaceholder.typicode.com/posts/1' \  
  -H 'accept: */*' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "id": 1,  
    "title": "foo",  
    "body": "bar",  
    "userId": 1  
  }'
```

Request URL

<https://jsonplaceholder.typicode.com/posts/1>

Server response

Code	Details
------	---------

200

Response body

```
{  
  "id": 1,  
  "title": "foo",  
  "body": "bar",  
  "userId": 1  
}
```

[Download](#)

Response headers

```
cache-control: no-cache  
content-type: application/json; charset=utf-8  
expires: -1  
pragma: no-cache
```

Responses

Code	Description	Links
------	-------------	-------

200

Publicación actualizada correctamente.

No links

PATCH [`/posts/{id}`](#) Actualizar parcialmente una publicación ^

Actualiza solo algunos campos (como el título) de una publicación existente.

Parameters

[Cancel](#)[Reset](#)

Name	Description
------	-------------

id * required	
----------------------	--

1

integer

(path)

Request body required	application/json
-----------------------	------------------

Edit Value	Schema
------------	--------

{ "title": "updated title" }

Execute	Clear
---------	-------

Responses

Curl

```
curl -X 'PATCH' \
  'https://jsonplaceholder.typicode.com/posts/1' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "title": "updated title"
}'
```

Request URL

<https://jsonplaceholder.typicode.com/posts/1>

Server response

Code	Details
200	<p>Response body</p> <pre>{ "userId": 1, "id": 1, "title": "updated title", "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nnreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"</pre> <p>Download</p> <p>Response headers</p> <pre>cache-control: no-cache content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre> <p>Responses</p>

Code	Description	Links
200	Publicación actualizada parcialmente.	No links

DELETE /posts/{id} Eliminar una publicación ^

Elimina una publicación existente por su ID.

Parameters

[Cancel](#)

Name	Description
------	-------------

id * required	<input type="text" value="1"/>
integer	
(path)	

[Execute](#)

[Clear](#)

Responses

Curl

```
curl -X 'DELETE' \
  'https://jsonplaceholder.typicode.com/posts/1' \
  -H 'accept: */*'
```

Request URL

<https://jsonplaceholder.typicode.com/posts/1>

Server response**Code** **Details**

200

Response body

{}

[Download](#)**Response headers**

```
cache-control: no-cache
content-type: application/json; charset=utf-8
expires: -1
pragma: no-cache
```

Responses**Code** **Description****Links**

200

Publicación eliminada correctamente.

No links

Todos

 Endpoints relacionados con tareas (to-do list). ^

GET **/todos** Obtener todas las tareas ^

Devuelve una lista de todas las tareas (to-dos) registradas.

Parameters

Cancel

No parameters

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/todos' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/todos>

Server response

Code	Details
------	---------

200

Response body

```
[  
  {  
    "userId": 1,  
    "id": 1,  
    "title": "delectus aut autem",  
    "completed": false  
  },  
  {  
    "userId": 1,  
    "id": 2,
```

Code**Details**

```
"title": "quis ut nam facilis et officia qui",
"completed": false
},
{
  "userId": 1,
  "id": 3,
  "title": "fugiat veniam minus",
  "completed": false
},
{
  "userId": 1,
  "id": 4,
  "title": "et porro tempora",
```

[Download](#)**Response headers**

```
cache-control: max-age=43200
content-length: 4005
content-type: application/json; charset=utf-8
expires: -1
pragma: no-cache
```

Responses**Code****Description****Links**

200

Lista de tareas obtenida correctamente.

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[

  {
    "userId": 0,
    "id": 0,
    "title": "string",
```

Code	Description	Links
	<pre> "completed": true }]</pre>	
POST	/todos Crear una nueva tarea	^
	Crea una nueva tarea asociada a un usuario.	
Parameters		Cancel Reset
No parameters		
Request body <small>required</small>		application/json
Edit Value <small>Schema</small>		

```
{  
  "title": "Buy milk",  
  "completed": false,  
  "userId": 1  
}
```

//

Execute**Clear**

Responses

Curl

```
curl -X 'POST' \  
  'https://jsonplaceholder.typicode.com/todos' \  
  -H 'accept: */*' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "title": "Buy milk",  
    "completed": false,  
    "userId": 1  
}'
```

Request URL

<https://jsonplaceholder.typicode.com/todos>

Server response

Code	Details
------	---------

201

Response body

```
{  
    "title": "Buy milk",  
    "completed": false,  
    "userId": 1,  
    "id": 201  
}
```

[Download](#)

Response headers

```
cache-control: no-cache  
content-length: 75  
content-type: application/json; charset=utf-8  
expires: -1  
location: https://jsonplaceholder.typicode.com/todos/201  
pragma: no-cache
```

Responses

Code	Description	Links
------	-------------	-------

201

Tarea creada exitosamente.

No links

GET [/todos/{id}](#) Obtener una tarea por ID ^

Devuelve los detalles de una tarea específica según su ID.

Parameters

Cancel

Name	Description
------	-------------

id * required

integer

(path)

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
'https://jsonplaceholder.typicode.com/todos/1' \
-H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/todos/1>

Server response

Code	Details
------	---------

200

Response body

```
{
  "userId": 1,
  "id": 1,
```

Code	Details	Download
	<pre>"title": "delectus aut autem", "completed": false }</pre> <p>Response headers</p> <pre>cache-control: max-age=43200 content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre>	
Responses		
Code	Description	Links
200	Tarea encontrada exitosamente. Media type application/json <small>Controls Accept header.</small>	No links
	Example Value	Schema
		<pre>{ "userId": 0, "id": 0, "title": "string", "completed": true }</pre>

Users Endpoints relacionados con la información de los usuarios.



GET /users Obtener todos los usuarios



Devuelve una lista con todos los usuarios registrados.

Parameters

[Cancel](#)

No parameters

[Execute](#)

[Clear](#)

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/users' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/users>

Server response

Code	Details
200	Response body

[
 {
 "id": 1,
 "name": "Leanne Graham",
 }

Code	Details	
	<pre>"username": "Bret", "email": "Sincere@april.biz", "address": { "street": "Kulas Light", "suite": "Apt. 556", "city": "Gwenborough", "zipcode": "92998-3874", "geo": { "lat": "-37.3159", "lng": "81.1496" } }, "phone": "1-770-736-8031 x56442", "website": "hildegard.org", "company": { "name": "Romaguera-Crona", "catchPhrase": "Multi-layered client-server neural-net", "bs": "harness real-time e-markets" }</pre> <div style="float: right;">Download</div>	
Response headers		
<pre>cache-control: max-age=43200 content-length: 1847 content-type: application/json; charset=utf-8 expires: -1 pragma: no-cache</pre>		
Responses		
Code	Description	Links
200	<p>Lista de usuarios obtenida correctamente.</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p>	<i>No links</i>
	Example Value	Schema

Code	Description	Links
<pre>[{ "id": 0, "name": "string", "username": "string", "email": "string", "phone": "string", "website": "string" }]</pre>		

GET /users/{id} Obtener un usuario por ID ^

Devuelve la información de un usuario específico identificado por su ID.

Parameters

Cancel

Name	Description
id * required integer (path)	<input type="text" value="1"/>

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'https://jsonplaceholder.typicode.com/users/1' \
  -H 'accept: application/json'
```

Request URL

<https://jsonplaceholder.typicode.com/users/1>

Server response

Code	Details
------	---------

200

Response body

```
{
  "id": 1,
  "name": "Leanne Graham",
  "username": "Bret",
  "email": "Sincere@april.biz",
  "address": {
    "street": "Kulas Light",
    "suite": "Apt. 556",
    "city": "Gwenborough",
    "zipcode": "92998-3874",
    "geo": {
      "lat": "-37.3159",
      "lng": "81.1496"
    }
  },
  "phone": "1-770-736-8031 x56442",
  "website": "hildegard.org",
  "company": {
    "name": "Romaguera-Crona",
    "catchPhrase": "Multi-layered client-server neural-net",
    "bs": "harness real-time e-markets"
  }
},
```

[Download](#)

Code**Details**

}

Response headers

```
cache-control: max-age=43200
content-type: application/json; charset=utf-8
expires: -1
pragma: no-cache
```

Responses**Code****Description****Links**

200

Usuario encontrado exitosamente.

No link

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
  "id": 0,
  "name": "string",
  "username": "string",
  "email": "string",
  "phone": "string",
  "website": "string"
}
```

Schemas



```
Album {  
    userId      integer  
    id          integer  
    title       string  
}
```



```
AlbumInput {  
    userId*    integer  
    title*     string  
}
```



```
Comment {  
    postId     integer  
    id         integer  
    name       string  
    email      string  
    body       string  
}
```



```
CommentInput {  
    postId*           integer  
    name*            string  
    email*           string  
    body*            string  
}
```



```
Photo {  
    albumId          integer  
    id               integer  
    title            string  
    url              string  
    thumbnailUrl     string  
}
```



```
Post {  
    userId           integer  
    id               integer  
    title            string  
    body             string  
}
```



```
PostInput {  
    userId*           integer  
    title*            string  
    body*             string  
}
```



```
Todo {  
    userId           integer  
    id               integer  
    title            string  
    completed        boolean  
}
```



```
TodoInput {  
    userId*          integer  
    title*           string  
    completed*       boolean  
}
```



```
User {  
    id          integer  
    name        string  
    username    string  
    email       string  
    phone       string  
    website     string  
}
```