

User Management

URL

<https://reqres.in/>

Introduction

The API documentation contains technical information for RESTful APIs in accordance with user management. You can create a new user and/or manage the existing users to the REQRES application. You can use the below APIs which explain about all RESTful operations (CRUD operations), including their request and response parameters.

This application has two pages and each page contains 6 users. In this way, there are total 12 users who already exist in this application.

Resource URL

GET <https://reqres.in/api/users?page=2>

Find users by page

This API is used to retrieve the user details who are listed on page 2.

Query Parameters

Parameter	Description	Type	Required	Notes
page	The numbering of the page to filter users listed on page 2	integer	Required	

Headers

Header Name	Description	Required	Values
Content-Type	The format of data to be posed	Optional	application/json or application.xml. The default format is application/json
Accept	The format of data to be returned	Optional	application/json or application.xml. The default format is application/json

Output Parameters

Parameter	Description	Required	Values
Content-Type	The format of data to be posed	Optional	application/json or application.xml. The default format is application/json

Sample Response

```
{
  "page": 2,
  "per_page": 6,
  "total": 12,
  "total_pages": 2,
  "data": [
    {
      "id": 7,
      "email": "michael.lawson@reqres.in",
      "first_name": "Michael",
      "last_name": "Lawson",
      "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/follettkyle/128.jpg"
    },
    {
      "id": 8,
      "email": "lindsay.ferguson@reqres.in",
      "first_name": "Lindsay",
      "last_name": "Ferguson",
      "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/araa3185/128.jpg"
    },
    {
```

```
"id": 9,

"email": "tobias.funke@reqres.in",

"first_name": "Tobias",

"last_name": "Funke",

"avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/vivekprvr/128.jpg"

},

{

  "id": 10,

  "email": "byron.fields@reqres.in",

  "first_name": "Byron",

  "last_name": "Fields",

  "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/russoedu/128.jpg"

},

{

  "id": 11,

  "email": "george.edwards@reqres.in",

  "first_name": "George",

  "last_name": "Edwards",

  "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/mrmoiree/128.jpg"

},

{

  "id": 12,

  "email": "rachel.howell@reqres.in",

  "first_name": "Rachel",

  "last_name": "Howell",

  "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/hebertialmeida/128.jpg"

}
```

```
],  
  
  "ad": {  
  
    "company": "StatusCode Weekly",  
  
    "url": "http://statuscode.org/",  
  
    "text": "A weekly newsletter focusing on software development, infrastructure, the server,  
performance, and the stack end of things."  
  
  }  
  
}
```

Resource URL

GET <https://reqres.in/api/users?id=3>

Find a user by Id

This API returns details of a user specific to the Id.

Query Parameters

Parameter	Description	Type	Required	Notes
Id (query)	A unique identifier that represents a user	integer	Required	

Headers

Header Name	Description	Required	Values
Content-Type	The format of data to be posed	Optional	application/json or application.xml. The default format is application/json
Accept	The format of data to be returned	Optional	application/json or application.xml. The default format is application/json

Output Parameters

Parameter	Description	Type
-----------	-------------	------

data	A list of user data in columns	Array of user details
id	Unique identifier for the user	Integer
email	The email address of the user	String
First_name	The first name of the user	String
Last_name	The last name of the user	String

Sample Response

```
{
  "data": {
    "id": 3,
    "email": "emma.wong@reqres.in",
    "first_name": "Emma",
    "last_name": "Wong",
    "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/olegpogodaev/128.jpg"
  },
  "ad": {
    "company": "StatusCode Weekly",
    "url": "http://statuscode.org/",
    "text": "A weekly newsletter focusing on software development, infrastructure, the server, performance, and the stack end of things."
  }
}
```

Resource URL

POST <https://reqres.in/api/users>

Create a new user

This API creates a new user into the database.

Input Parameters

Parameter	Description	Type	Required	Notes
name	Name of the user to be created	string	Required	
job	A job type of the user	string	Optional	

Headers

Header Name	Description	Required	Values
Content-Type	The format of data to be posed	Optional	application/json or application.xml. The default format is application/json
Accept	The format of data to be returned	Optional	application/json or application.xml. The default format is application/json

Sample Request

```
{
  "name": "Adam",
  "job": "team lead"
}
```

Output Parameters

Parameter	Description	Type	Required	Notes
id	Unique identifier for user.	string	Required	
createdAt	Date and time of the user creation	string	Optional	The default date-time format is YYYY-MM-DD'T'HH:MM:SS. The local time is in UTC.

Sample Response

```
{
```

```
"id": "442",  
  
"createdAt": "2020-05-13T06:16:51.896Z"  
}
```

Resource URL

PUT https://regres.in/api/users?id=8&first_name=Lindsay

Update last name of user

This API updates the last name of a user whose Id is 8 and first name is Lindsay.

Query Parameters

Parameter	Description	Type	Required	Notes
Id	Unique identifier that represents a user	integer	Required	
first_name	First name of the user	string	Optional	

Header Parameters

Header Name	Description	Required	Values
Content-Type	The format of data to be posed	Optional	application/json or application.xml. The default format is application/json
Accept	The format of data to be returned	Optional	application/json or application.xml. The default format is application/json

Input Parameters

Parameter	Description	Type	Required	Notes
last_name	Last name of the user to be changed	integer	Required	

Sample Request

```
{
  "last_name": "Colon"
}
```

Output Parameters

Element	Description	Type	Required	Notes
updatedAt	Date and time when user details got updated	string	Required	The default date-time format is YYYY-MM-DD'T'HH:MM:SS. The local time is in UTC.

Sample Response

```
{
  "updatedAt": "2020-05-13T06:19:50.633Z"
}
```

Status Codes and Errors

The following table lists the returned HTTP status codes.

Code	Description	Notes
200	OK	Success
201	Created	User is successfully created
401	Unauthorized	The access token is invalid
404	Not found	The user Id not found
500	Server Error	Internal Server Error