

Create a NodeJS project with following APIs

APIs to create

- **GET** /todos - returns a list of todos without user id field
- **GET** /user/<pass-your-user-id> - returns user information along with todo items where userid matches with the one provided in the URL

GET /todos

Description:

Your API should internally call a third party API to get the list of todo items. The third party API URL is <https://jsonplaceholder.typicode.com/todos> & the method is **GET** which will return a list of todo items in a JSON array of objects. Each object contains fields like

1. userId
2. id
3. title
4. completed

Sample output of the third party API

The screenshot shows a REST client interface with a GET request to `https://jsonplaceholder.typicode.com/todos`. The response status is 200 OK and the time taken is 525 ms. The response body is displayed in JSON format, showing an array of five todo items.

```
[{"userId": 1, "id": 1, "title": "delectus aut autem", "completed": false}, {"userId": 1, "id": 2, "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", "completed": true}, {"userId": 1, "id": 5, "title": "laboriosam mollitia et enim quasi adipisci quia provident illum", "completed": false}]
```

Expected Response:

The API should return all the objects with all the fields returned by the third party API **except the `userId` field**.

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:3000/todos`
- Method:** GET
- Authorization:** No Auth
- Status:** 200 OK
- Time:** 269 ms
- Response Body (JSON):**

```
[{"id": 1, "title": "delectus aut autem", "completed": false}, {"id": 2, "title": "quis ut nam facilis et officia qui", "completed": false}, {"id": 3, "title": "fugiat veniam minus", "completed": false}, {"id": 4, "title": "et porro tempora", "completed": true}, {"id": 5, "title": "laboriosam mollitia et enim quasi adipisci quia provident illum", "completed": false}, {"id": 6, "title": "qui ullam ratione quibusdam voluptatem quia omnis", "completed": false}]
```

GET /user/<pass your user id here>

Description:

You have to get the user information for the given user id (in the url) from a third party API. The third party API URL is : <https://jsonplaceholder.typicode.com/users/><pass your user id here>.

You also need to get all the todo items from this API <https://jsonplaceholder.typicode.com/todos> and find out all the todo items where user id matches with the above user id passed in the URL and append it to the output

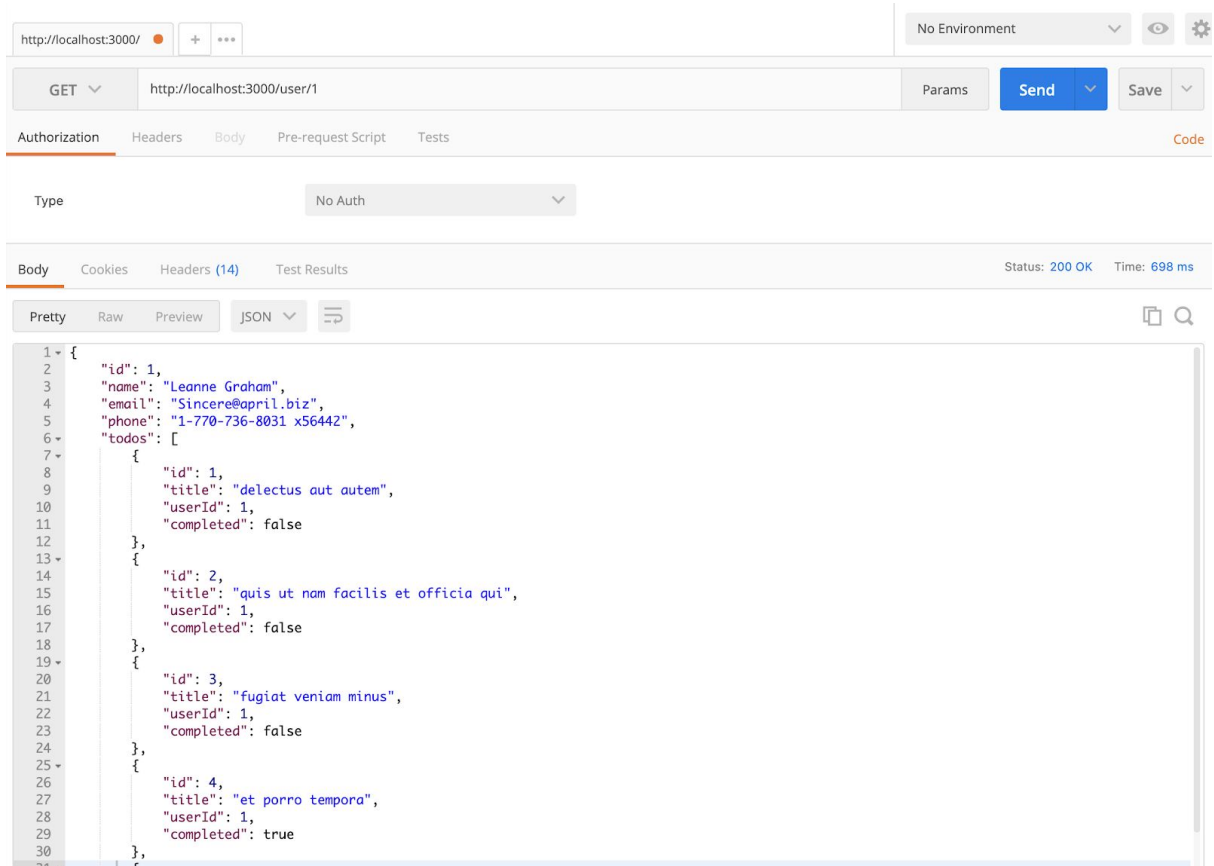
Sample output of the third party API

The screenshot shows a REST client interface with a GET request to `https://jsonplaceholder.typicode.com/users/1`. The response is a JSON object representing a user, displayed in a pretty-printed format. The status is 200 OK and the time taken is 470 ms.

```
1 {
2   "id": 1,
3   "name": "Leanne Graham",
4   "username": "Bret",
5   "email": "Sincere@april.biz",
6   "address": {
7     "street": "Kulas Light",
8     "suite": "Apt. 556",
9     "city": "Gwenborough",
10    "zipcode": "92998-3874",
11    "geo": {
12      "lat": "-37.3159",
13      "lng": "81.1496"
14    }
15  },
16  "phone": "1-770-736-8031 x56442",
17  "website": "hildegard.org",
18  "company": {
19    "name": "Romaguera-Crona",
20    "catchPhrase": "Multi-layered client-server neural-net",
21    "bs": "harness real-time e-markets"
22  }
23 }
```

Expected Output

The output contains user information obtained by third party API and also the todo items where user id field matches with the user id provided in the URL



The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:3000/user/1`
- Method:** GET
- Status:** 200 OK
- Time:** 698 ms
- Response Body (JSON):**

```
1 {
2   "id": 1,
3   "name": "Leanne Graham",
4   "email": "Sincere@april.biz",
5   "phone": "1-770-736-8031 x56442",
6   "todos": [
7     {
8       "id": 1,
9       "title": "delectus aut autem",
10      "userId": 1,
11      "completed": false
12    },
13    {
14      "id": 2,
15      "title": "quis ut nam facilis et officia qui",
16      "userId": 1,
17      "completed": false
18    },
19    {
20      "id": 3,
21      "title": "fugiat veniam minus",
22      "userId": 1,
23      "completed": false
24    },
25    {
26      "id": 4,
27      "title": "et porro tempora",
28      "userId": 1,
29      "completed": true
30    }
31  ]
32 }
```