

Scoreboard

Rules

Register yourself on the scoreboard as fast as you can.

Walkthrough

1. Groups

We encourage you to work alone to get the most hands-on experience, but you can form groups if you want.

2. Get your ID

You've received a unique identifier by mail. If you've not received it, check your spam folder or contact us to get yours! If you're on a group, gather all your IDs.

3. Create your request

Following the specification at [our site](#), create your request body with your ID(s) and your preferred nickname, e.g.

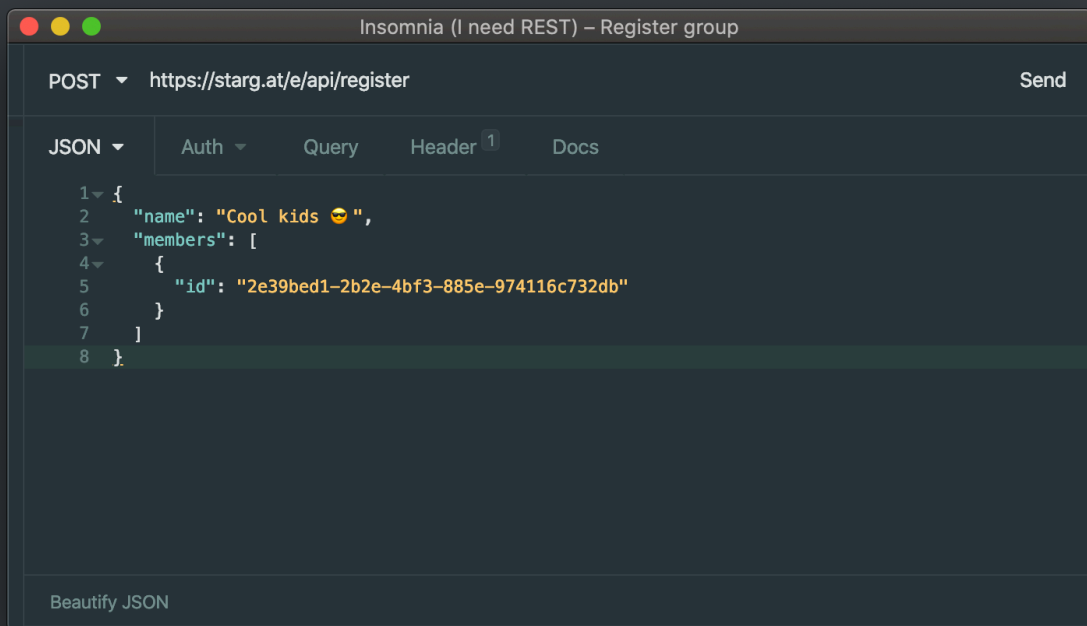
```
{
  "name": "Cool kids 😎",
  "members": [
    {
      "id": "582241eff-8b13-26f0-8292-4326f1b27dea"
    },
    {
      "id": "56121edd-2a15-12e6-1163-1236f1a63def"
    }
  ]
}
```

4. Send your request

Using either a REST client like Insomnia, or using your language of choice, perform a POST request to the `/register` endpoint with your request body. Example of this in Node:

```
const body = {
  name: 'Cool kids 😎',
  members: [
    {
      id: '582241eff-8b13-26f0-8292-4326f1b27dea'
    }
  ]
}
request.post('https://starg.at/e/api/register', body, function(err, res,
body) {
  // If any errors
  console.log(err);
  // The token, if any
  console.log(body);
})
```

Example using Insomnia:



Your response *should* look something like this:

5. JSON Web Tokens and Authorization 🔑

What you've gotten in response is a JSON Web Token (JWT). This is a widely used token format that we can use to store data about the user making a request. When you made a call to the `/register` endpoint, we created this token exclusively for you. This way, you can authenticate yourself when making further requests that only registered users should have access to. You can see what information we stored in your token by pasting it on jwt.io.

When making authenticated requests, a common practice is to include your token in the *headers* of your request. HTTP headers are key-value pairs that can contain lots of metadata about each request. We check the headers for your token using the `Authorization` key. Your headers should then look like this:

```
Authorization: <type> <credentials>
```

This is just to make you familiar with a standard way of making authenticated requests. If you're interested in knowing more, or want a better explanation, just contact one of us!

6. Submitting to the scoreboard 🏆

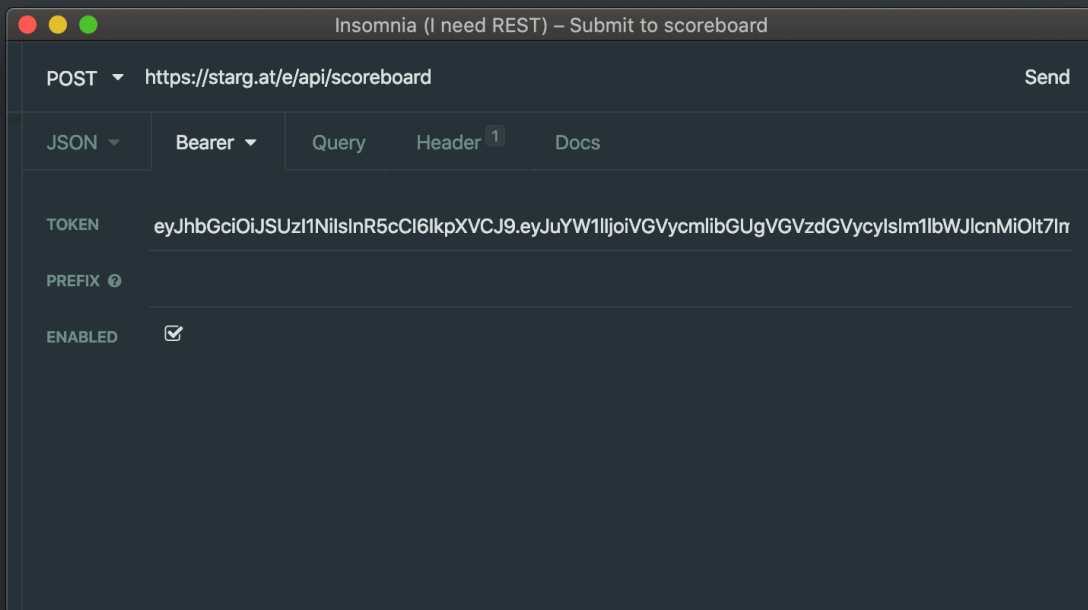
Now that you've got your token, it's time to make your first authenticated request towards the `/scoreboard` endpoint. Since your information is stored in the token, there's no request body needed. You only need to add the `Authorization` header with your token. The HTTP method should be POST. An example of the request in Node:

```
// Setting the headers 'manually'
const options = {
  url: 'https://starg.at/e/api/scoreboard',
  headers: {
    Authorization: `Bearer ${token}`,
  },
}

request.post(options, function(err, res, body) {
  // If error
  console.log(err)
  // If successfull
  console.log(body)
})

// or using the request library's built-in auth requests
request
  .post('https://starg.at/e/api/scoreboard', function(err, res, body) {})
  .auth(null, null, true, token)
```

Example using Insomnia:



Your response should look something like this:

```
{
  "message": "Submitted! 🏆",
  "group": {
    "id": "qSPkndIWQM0YUoIsLrrZ",
    "name": "Cool kids 😎",
    "members": [
      {
        "id": "2e39bed1-2b2e-4bf3-885e-974116c732db"
      }
    ]
  }
}
```

7. Finish! 🏆

Aaand congrats! You've now successfully retrieved a token and made an authenticated request! You should now see yourself on the scoreboard. If you'd like, you can send a GET request to the same endpoint to get a list of the groups on the scoreboard yourself.