
gh.el User Manual

Release 0.3.2

Yann Hodique

Jun 25, 2017

Contents

1	Basics	3
2	Issues	5
3	Indices and tables	7

gh.el is a low-level client library to interact with the *GitHub API* <<http://developer.github.com/>>.

Main features include:

- synchronous and asynchronous modes
- flexible authentication
- allows for multiple parallel sessions

Contents:

CHAPTER 1

Basics

gh.el is built on top of Eieio. The scope of this client library is to provide plumbing primitives that will allow full use of the GitHub API.

gh.el is built on top of Eieio. The scope of this client library is to provide plumbing primitives that will allow full use of the GitHub API.

gh.el allows access to GitHub issues.

First, connect to the API:

```
(gh-issues-api "api")
```

This will OAuth connect to GitHub and return an API connection object.

The API connection object can be passed to issues methods:

```
(gh-issues-issue-list (gh-issues-api "API") "sigma" "gh.el")
```

The issue list has a class *gh-api-paged-response* which has a member *data* which can be used to retrieve the data sent back from GitHub:

```
(oref
  (gh-issues-issue-list (gh-issues-api "API") "sigma" "gh.el")
  data)
```

This returns a list of items of class *gh-issues-issue*. You can further *oref* those to get data. Putting it all together we might have:

```
(defun fill-string (str)
  (with-temp-buffer
    (insert str)
    (fill-paragraph)
    (buffer-string)))

(mapcar
  (lambda (issue)
    (insert
      (format
        "#%s %s -- %s\n%s\n\n"

```

```
(oref it number) ; the issue number
(oref it created_at) ; the data
(fill-string (oref it title)) ; the title, filled
(fill-string
  (replace-regexp-in-string
    "\\r" "\\n" (oref it body)))) ; the body filled
(oref
  (gh-issues-issue-list ghcon "sigma" "gh.el")
  data))
```

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`