

Web API 完全に理解した

～黎明編～




@unsoluble_sugar

**TOPGATE, Inc.
Leader/Engineer**

Web API
完全に理解してますか？




 聞いたことある

 完全に理解した

 何もわからない

 チョットデキル

 聞いたことある

 完全に理解した

 何もわからない

 ちょっとデキル

概要

- 対象は 🤔 **聞いたことある人**
- 😊 **完全に理解した**状態になるための入門編
- 細かい説明は省きます
- 不足部分は書籍やWebの情報で補完してね 😊

Web APIとは？



Web APIとは

- APIとは「**A**pplication **P**rogramming **I**nterface」の略
- 外部から機能を利用するための仕組み
- HTTPプロトコルを用いて、ネットワーク越しに機能呼び出す
APIのことを「**Web API**」と呼ぶ

どんなサービスで使われてるの？



Web APIが使われているサービス

- Google、Amazon、Twitter、Facebook、 etc...
- 世の中にはWeb APIを前提としたサービスが溢れている
ニュース、天気、地図、SNS、ショッピング、ゲーム、エンタメなどなど
- サーバ通信を行うスマホアプリでも利用
- 公開/非公開のWeb APIが存在

具体的には？

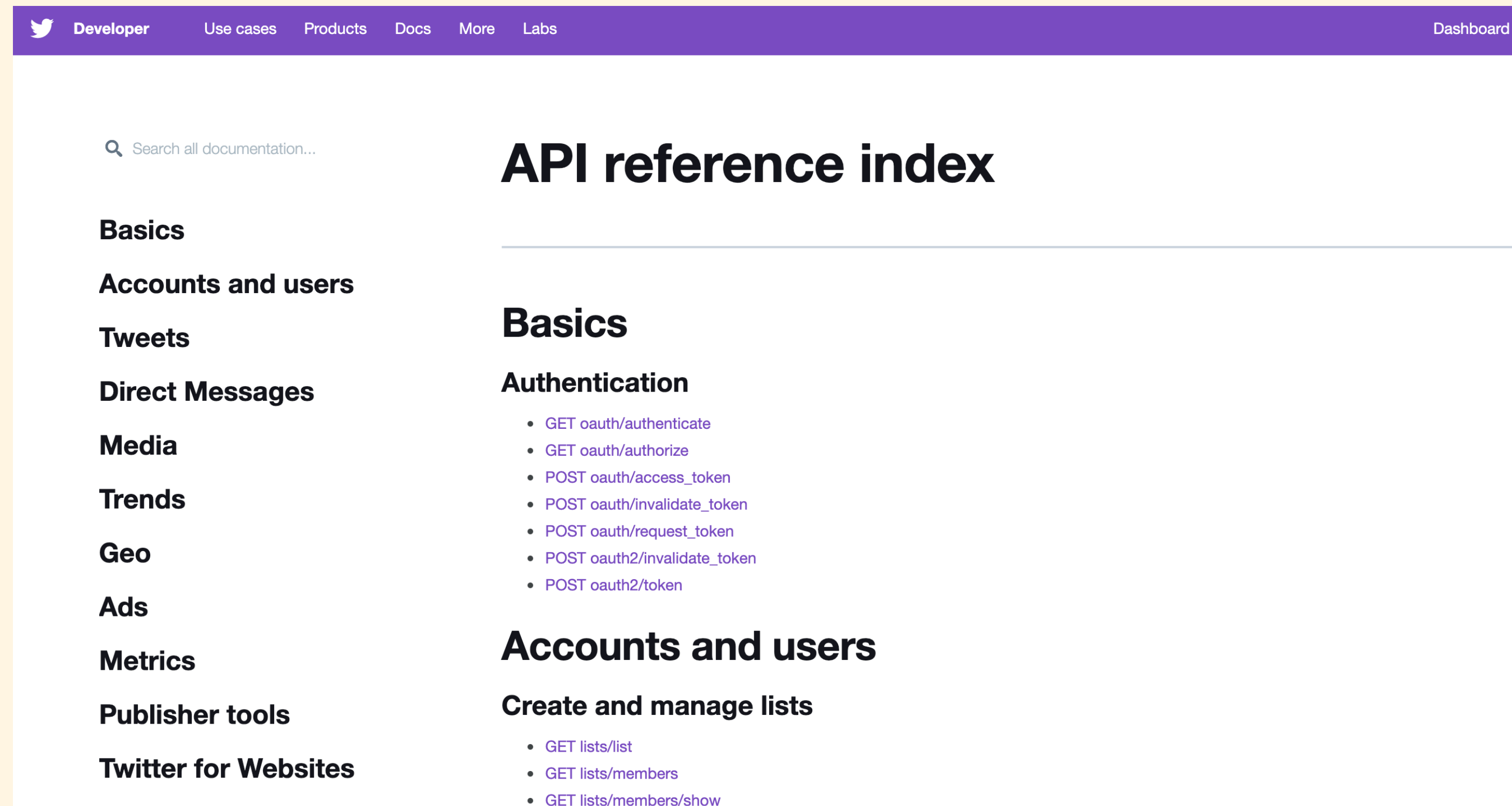


Twitter APIの場合

- TwitterはWeb APIを提供している
- 開発者はWeb APIを利用してTwitterの様々な機能が使える
- 多くのサードパーティ製クライアントが開発され発展
- 最近の使用制限が厳しい...😓

APIリファレンス

<https://developer.twitter.com/en/docs/api-reference-index>



The screenshot shows the Twitter API Reference Index page. The top navigation bar is purple and contains the Twitter logo, 'Developer', 'Use cases', 'Products', 'Docs', 'More', 'Labs', and 'Dashboard'. Below the navigation bar is a search bar with the placeholder text 'Search all documentation...'. The main content area is white and features a large heading 'API reference index'. On the left side, there is a vertical list of categories: 'Basics', 'Accounts and users', 'Tweets', 'Direct Messages', 'Media', 'Trends', 'Geo', 'Ads', 'Metrics', 'Publisher tools', and 'Twitter for Websites'. The main content area is divided into sections: 'Basics' (with a sub-section 'Authentication' containing a list of endpoints: GET oauth/authenticate, GET oauth/authorize, POST oauth/access_token, POST oauth/invalidate_token, POST oauth/request_token, POST oauth2/invalidate_token, and POST oauth2/token), 'Accounts and users' (with a sub-section 'Create and manage lists' containing a list of endpoints: GET lists/list, GET lists/members, and GET lists/members/show).

仕様詳細が公開されている

開発者はAPIを使って独自のアプリやWebサービスを作ることができる

GET oauth/authenticate

Allows a Consumer application to use an OAuth `request_token` to request user authorization.

This method is a replacement of [Section 6.2](#) of the [OAuth 1.0 authentication flow](#) for applications using the callback authentication flow. The method will use the currently logged in user as the account for access authorization unless the `force_login` parameter is set to `true`.

This method differs from [GET oauth / authorize](#) in that if the user has already granted the application permission, the redirect will occur without the user having to re-approve the application. To realize this behavior, you must enable the *Use Sign in with Twitter* setting on your [application record](#).

Resource URL

<https://api.twitter.com/oauth/authenticate>

Resource Information

Response formats	JSON
Requires authentication?	Yes
Rate limited?	Yes

Parameters

Name	Required	Description	Default Value	Example
<code>force_login</code>	optional	Forces the user to enter their credentials to ensure the correct users account is authorized.		<code>true</code>
<code>screen_name</code>	optional	Prefills the username input box of the OAuth login screen with the given value.		

実際のAPIの動きは？



ユーザータイムライン取得

HTTPメソッド	GET
レスポンス形式	JSON
ドメイン	https://api.twitter.com
エンドポイント	/1.1/statuses/user_timeline.json
パラメータ	screen_name, count, include_rts, etc...
認証	Bearer Token

APIリクエストを投げると...

```
$ curl "https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name=unsoluble_sugar&count=10" -H "Authorization: Bearer <ACCESS_TOKEN>" | jq
```

タイムラインの情報がJSONで取得できる

```
{
  "created_at": "Sun Jan 19 12:33:32 +0000 2020",
  "id": 1218874129316638700,
  "id_str": "1218874129316638720",
  "text": "リクエストヘッダーのAuthorizationって、日本語で表現すると「認証方式」とかで良いのかしら",
},
"source": "<a href=\"https://about.twitter.com/products/tweetdeck\" rel=\"nofollow\">TweetDeck</a>",
"user": {
  "id": 149069183,
  "id_str": "149069183",
  "name": "星影",
  "screen_name": "unsoluble_sugar",
  "location": "二次元",
  "description": "Tech Hunter代表。全文検索/スマホゲーム/チャットbot/IoTとかやってる雑多なエンジニア。最近はPMも兼任。Flutter/GCP/Firebase勉強中。ガジェオタでアニメオタでゲーオタ。累計1000万PVの個人ブログ運営。2児のパパ。本棚：https://t.co/kGyfPuFvIO",
  "url": "https://t.co/h7lq47ug5Z",
  "entities": {
    "url": {
      "urls": [
        {
          "url": "https://t.co/h7lq47ug5Z",
          "expanded_url": "https://unsolublesugar.com",
          "display_url": "unsolublesugar.com",
          ...
        }
      ]
    }
  }
}
```

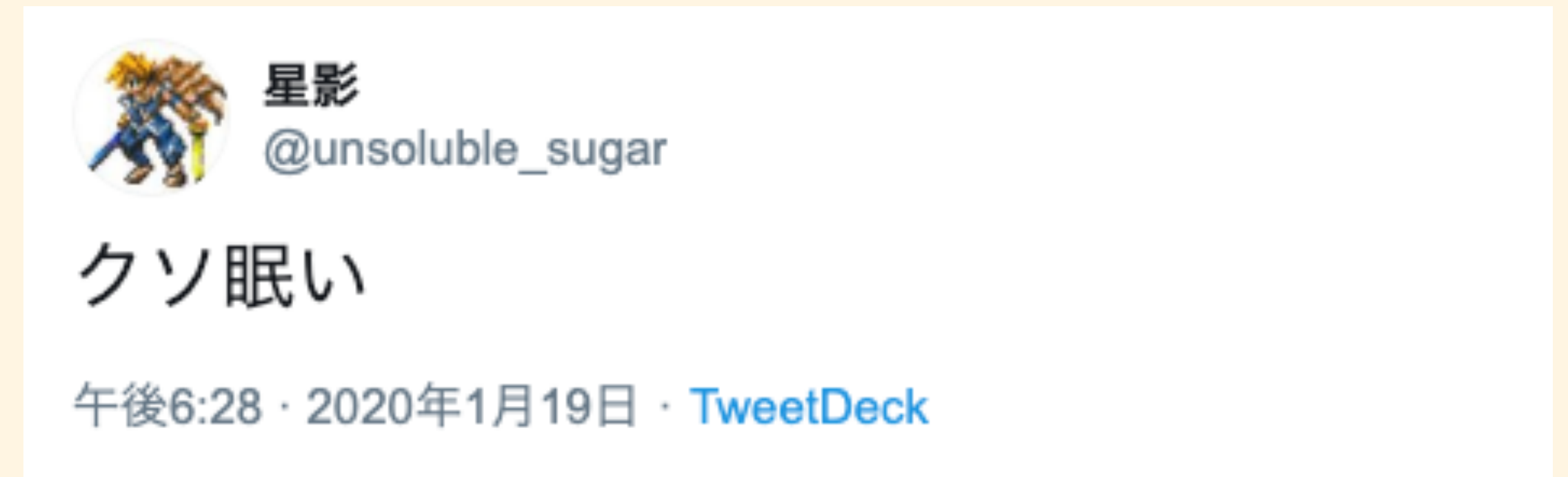
レスポンスデータを解析して画面に反映

- ツイート情報

text : クソ眠い

created_at : Sun Jan 19 12:33:32 +0000 2020

source : TweetDeck



- ユーザー情報

user.profile_image_url : http://pbs.twimg.com/profile_images/1109649870594293766/AQHwKd6N_normal.jpg

user.name : 星影

user.screen_name : unsoluble_sugar

様々な情報がAPIで取得できる

- ホームタイムラインの取得
GET statuses/home_timeline
- 個別ツイートの取得
GET statuses/show/:id
- リストの取得
GET lists/list
- フォロワー情報一覧の取得
GET followers/list

GET以外はデータの生成、更新、削除

- ツイートの投稿

POST statuses/update

- 画像のアップロード

POST media/upload

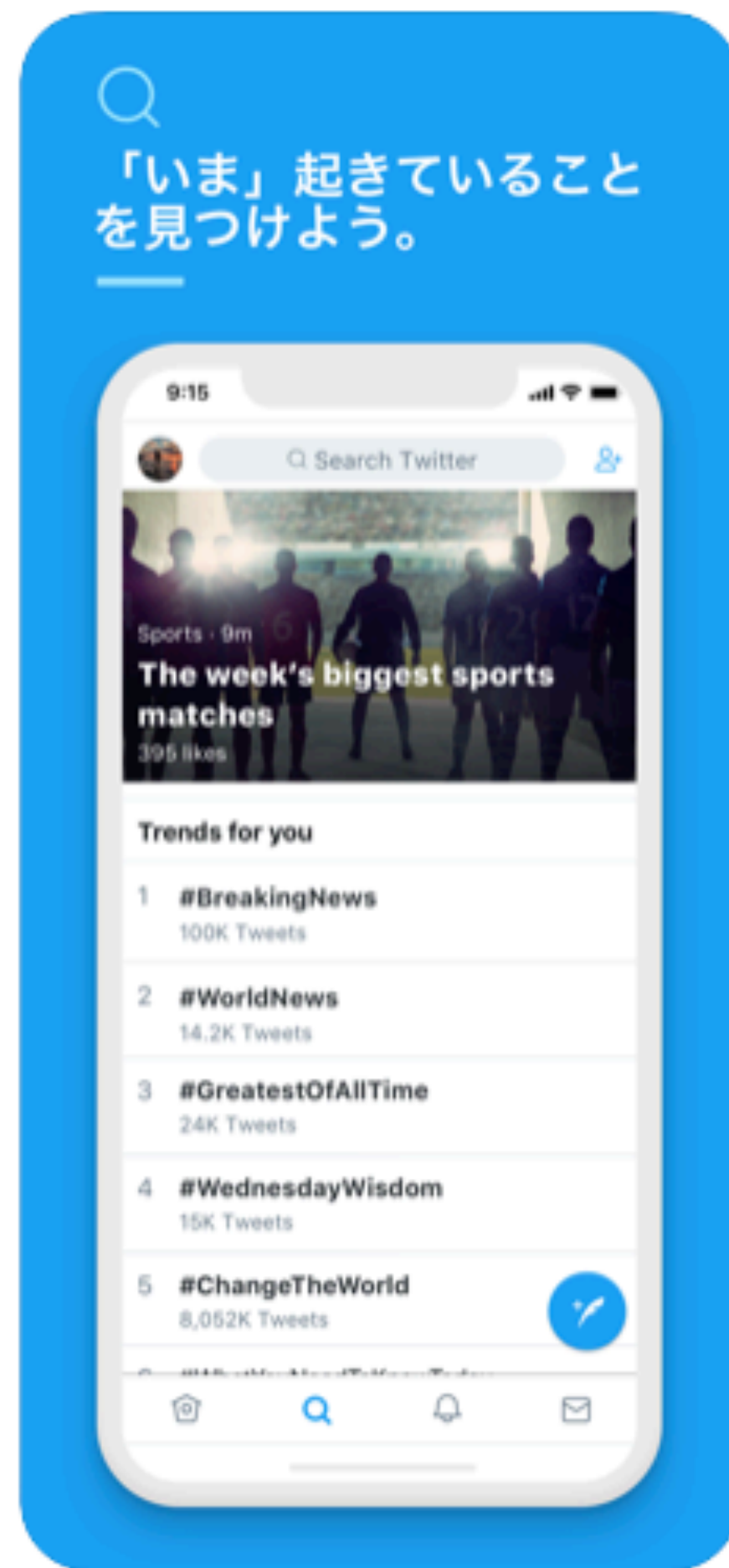
- DMの削除

DELETE direct_messages/events/destroy

- DMのウェルカムメッセージ変更（こんな機能あったんか...）

PUT direct_messages/welcome_messages/update

TwitterはWeb APIで出来ている

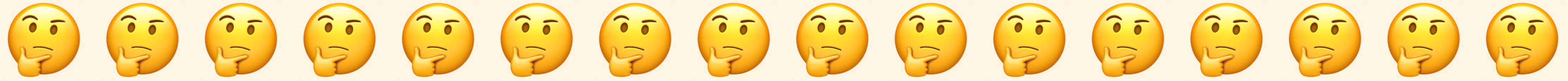


Web API



完全に理解した？





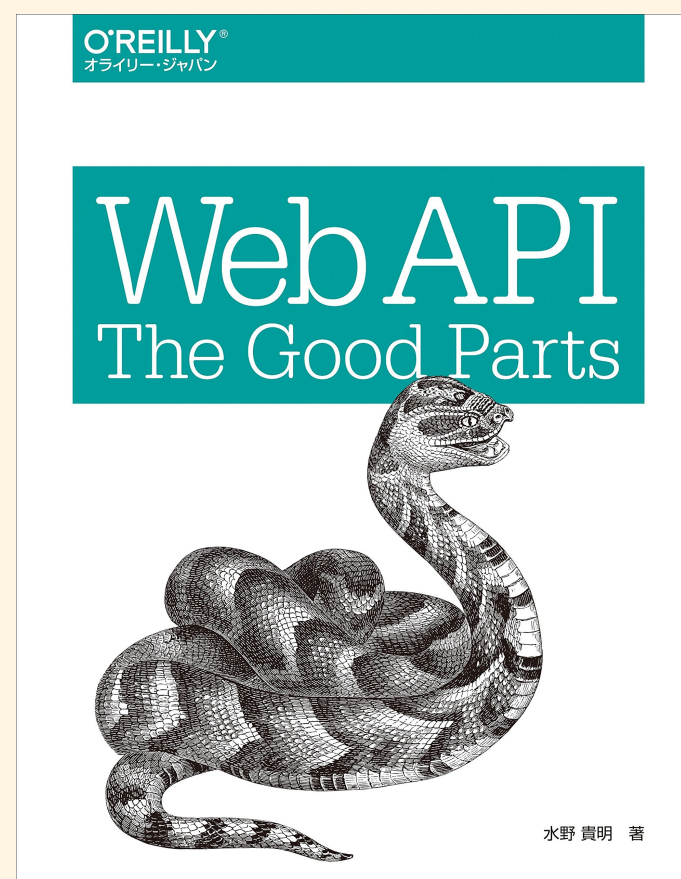
- HTTPヘッダー、ボディ (リクエスト・レスポンス)
- HTTPメソッド
- エンドポイント
- クエリストリング (URLパラメーター)
- URLエンコーディング
- フィルタ、ソート、検索、ページネーション
- 認証、認可 (Basic, OAuth2, OpenID Connect)
- セッション管理
- データ型
- 出力形式 (JSON, XML, HTML)
- エラーハンドリング
- ステータスコード (200, 404, 502)
- APIサーバ、Webサーバ
- バージョン管理
- 設計 (RESTful)
- セキュリティ



参考書籍



- Webを支える技術 — HTTP、URI、HTML、そしてREST
山本陽平 著 (WEB+DB PRESS plus)
2010年4月 発行



- Web API: The Good Parts
水野 貴明 著 (オライリージャパン)
2014年11月 発行

ようこそ

Web API の世界へ

