



# RESTful Web API

Author: 青シャツ(@EqualL2)

# RESTful Web API?

- HTTPベースでデータ操作を行なうインターフェース
- リソースに対して対応するURL(エンドポイント)が存在する
- 操作の種類をHTTPメソッドで示す
- 現代的なWebサービスの多くがAPIとして採用

# リソースとURLの対応

- 例：ユーザ equal-l2 がスターしたレポジトリ一覧

/users/equal-l2/starred

- users : ユーザ一覧
- equal-l2 : ユーザequal-l2
- starred : スターしたレポジトリ一覧

# 操作の種類

- HTTPメソッドごとに役割が決まっている
  - GET/HEAD : リソースの状態を取得する
  - PUT : リソースを置き換える
  - PATCH : リソースの情報を一部置き換える
  - DELETE : リソースを削除する
  - POST : リソースを新規作成する
- 例:

`/notifications/threads/:thread_id/subscription`

- GET : 現在の通知購読の状態を取得
- PUT : 通知購読の状態を設定
- DELETE : 通知購読を削除 (購読を取りやめる)

# RESTfulもどき

世界の全てを消滅させるAPIを作る



`GET /world/eraseEverything`

- GETは状態を変更してはいけない
- URLはメソッドでなくリソースに紐づくべき
- Googleクローラが世界を消滅させてしまう



`DELETE /world`

- 世界そのものが消滅してしまう（消したいのは世界の中身全て）



`PUT /world` (空リストを送る)

# RESTful以外のWeb API

- SOAP  
XMLを用いてリモートに命令を送る(RPC)ためのプロトコル
  - GraphQL  
Facebookなどが主体になって作られているクエリ言語
- いずれも単一エンドポイントに対してリクエストを送って使う

# 参考

- ・「RESTアンチパターン」

<https://www.infoq.com/jp/articles/rest-anti-patterns/>

- ・「REST, SOAP, GraphQL – Gesundheit!」

<https://medium.com/better-practices/rest-soap-graphql-gesundheit-6544053f65cf>

- ・「GitHub API v3 | GitHub Developer Guide」

<https://developer.github.com/v3/>