

メモ

webについて

世界的に標準的な文書の公開・閲覧システム

クライアント
(情報を要求する側)



サーバー
(要求に応える側)



リクエスト



レスポンス

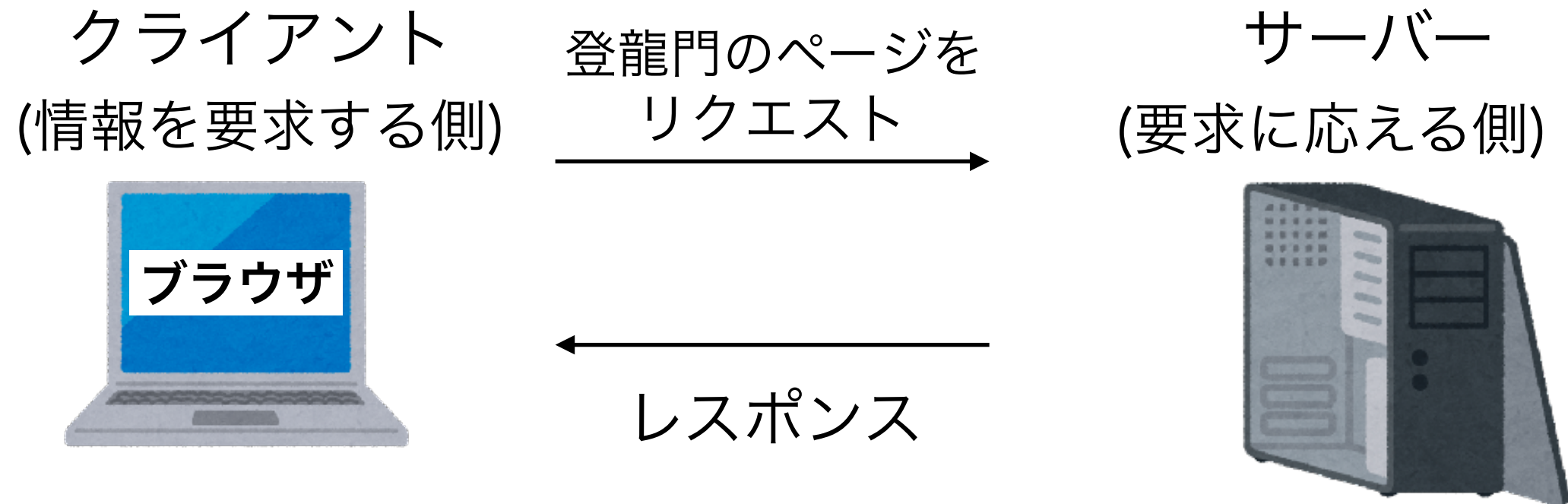


- ・ブラウザ(Chrome,...)
- ・メールアプリ
- ...

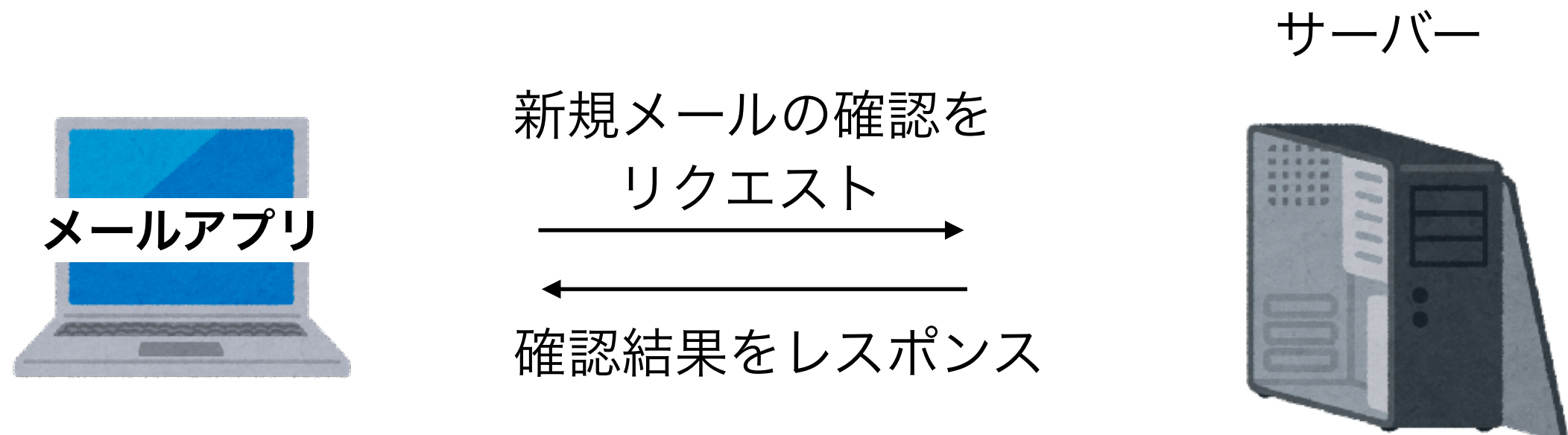
情報を要求する側(クライアント)と要求に応える側(サーバー)とのやりとりとして理解できる

webの具体例

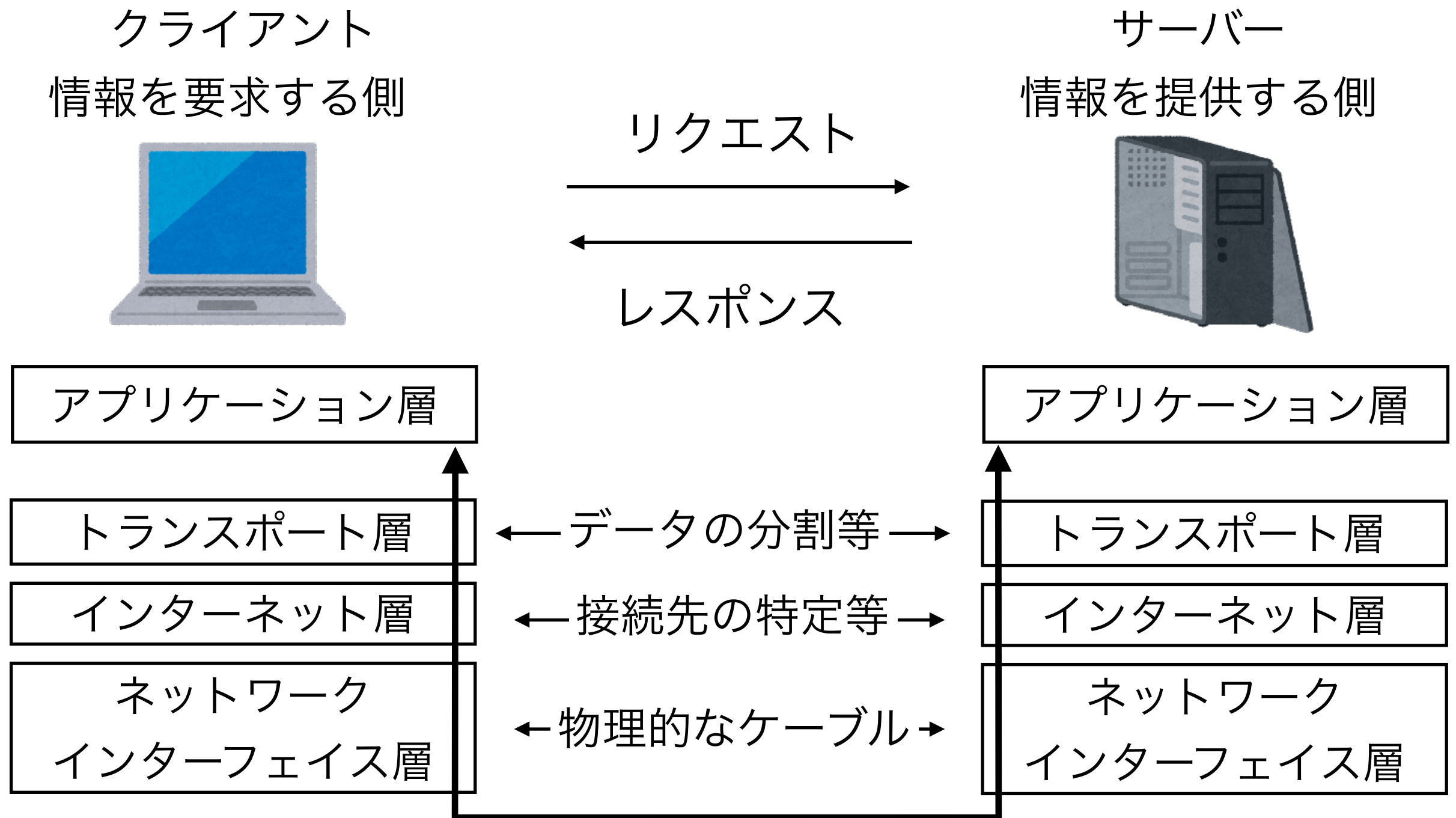
例1) 文書の閲覧: webページにアクセスして情報を得る



例2) アプリのユーザーインターフェイス



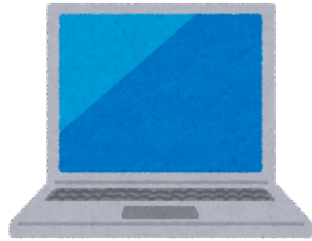
クライアント/サーバーのやりとりについて



- ・クライアント/サーバーでやりとりする情報は階層構造を成す
- ・アプリケーション層は、ユーザーが触れる部分を担う重要な部分(ブラウザの表示、メールの送信 etc)

アプリケーション層のプロトコルの一つ: HTTP

クライアント



アプリケーション層

リクエストの中身

```
POST / HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh; ... )
Accept: text/html,application/xhtml+xml,...
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary
Content-Length: 345
```

```
-12656974
(more data)
```

リクエスト

レスポンス

サーバー



アプリケーション層

レスポンスの中身

```
HTTP/1.1 403 Forbidden
Server: Apache
Content-Type: text/html; charset=iso-8859-1
Date: Wed, 10 Aug 2016 09:23:25 GMT
Keep-Alive: timeout=5, max=1000
Connection: Keep-Alive
Age: 3464
Date: Wed, 10 Aug 2016 09:45:25 GMT
X-Cache-Info: caching
Content-Length: 220
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML
2.0//EN">
(more data)
```

<https://developer.mozilla.org/ja/docs/Web/HTTP/Messages>

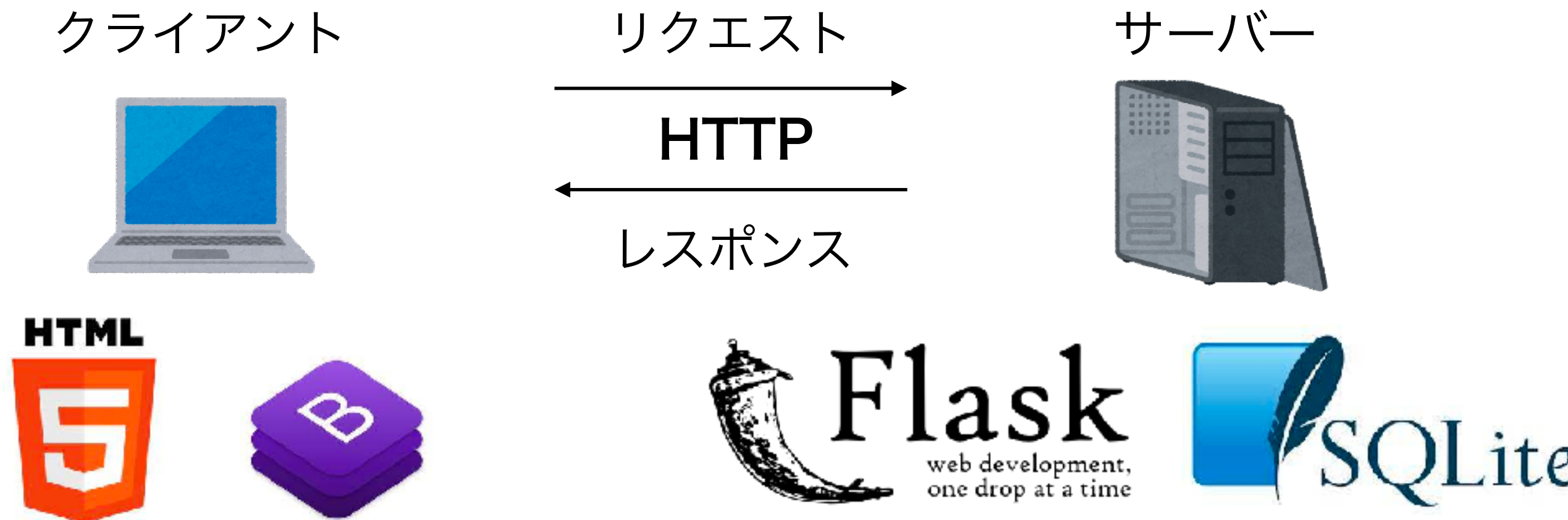
- HTTPはブラウザとサーバー間の通信に適用される通信規則(プロトコル)
- ステータス行、メッセージヘッダ、メッセージボディで統一されたメッセージでやりとりを行う

目次

- webについて
- webアプリ作成演習(HTTPの理解を深める目的)

webアプリ作成演習

ブラウザで動かせるToDoアプリの開発



CRUDの4機能をHTTP経由で動作させる

Create: データベースへの新規要素を追加する機能

Read: データベースを読み込んで画面に表示する機能

Update: データベースの既存要素を変更する機能

Delete: データベースの既存要素を削除する機能