

【今日の流れ】

- はじめに 25 mins
- [iLect + ターミナルの使い方] 5 mins
- Webの仕組み [HTTP実験] 10 min
- HTMLとCSS [コーディング] 15 min
- Jupyterの使い方 [Pythonの初歩] 20 min
- アプリケーションの制作[Flask] 30 min

Web工学とビジネスモデル

2017/04/26 5限 16:50 ~ 18:35
3号館 工32号講義室

東京大学大学院 工学系研究科 技術経営戦略学(TMI) 講義

中島 愛

連絡事項

連絡 + 質問 : tltk.io/weben20170426

演習教材 : ilect.net > [web_engineering_pub](#) > day1

スライド :

自己紹介

中島 愛

- 研究活動

HCI (ヒューマン・コンピュータ・インタラクション)

- インタラクション・デザイナー

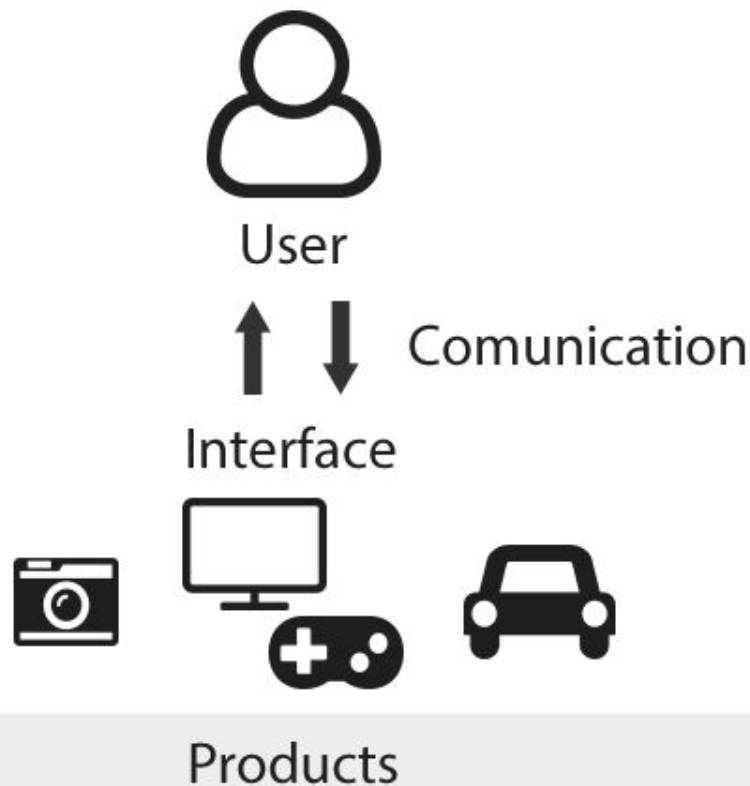
- フロントエンド・エンジニア

- 研究歴

九州大学、東京大学、筑波大学、産総研

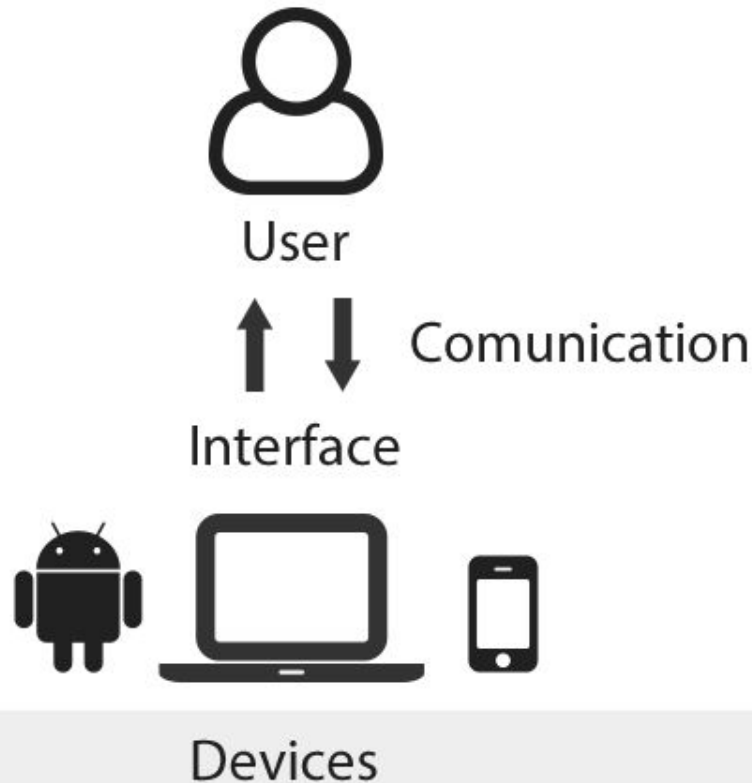
インタラクションデザイン

HCI (Human Computer Interaction)



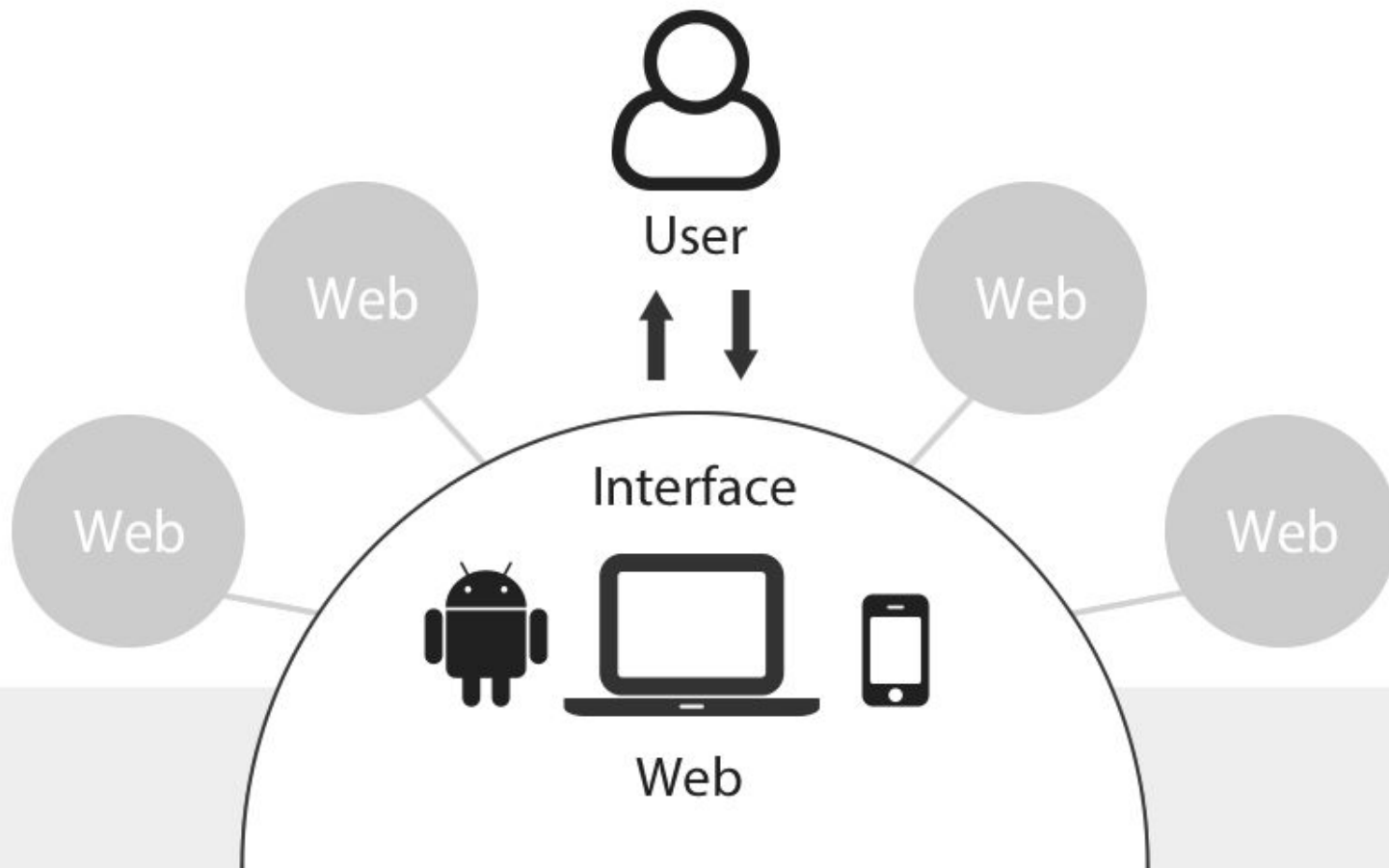
インタラクションデザイン

HCI (Human Computer Interaction)



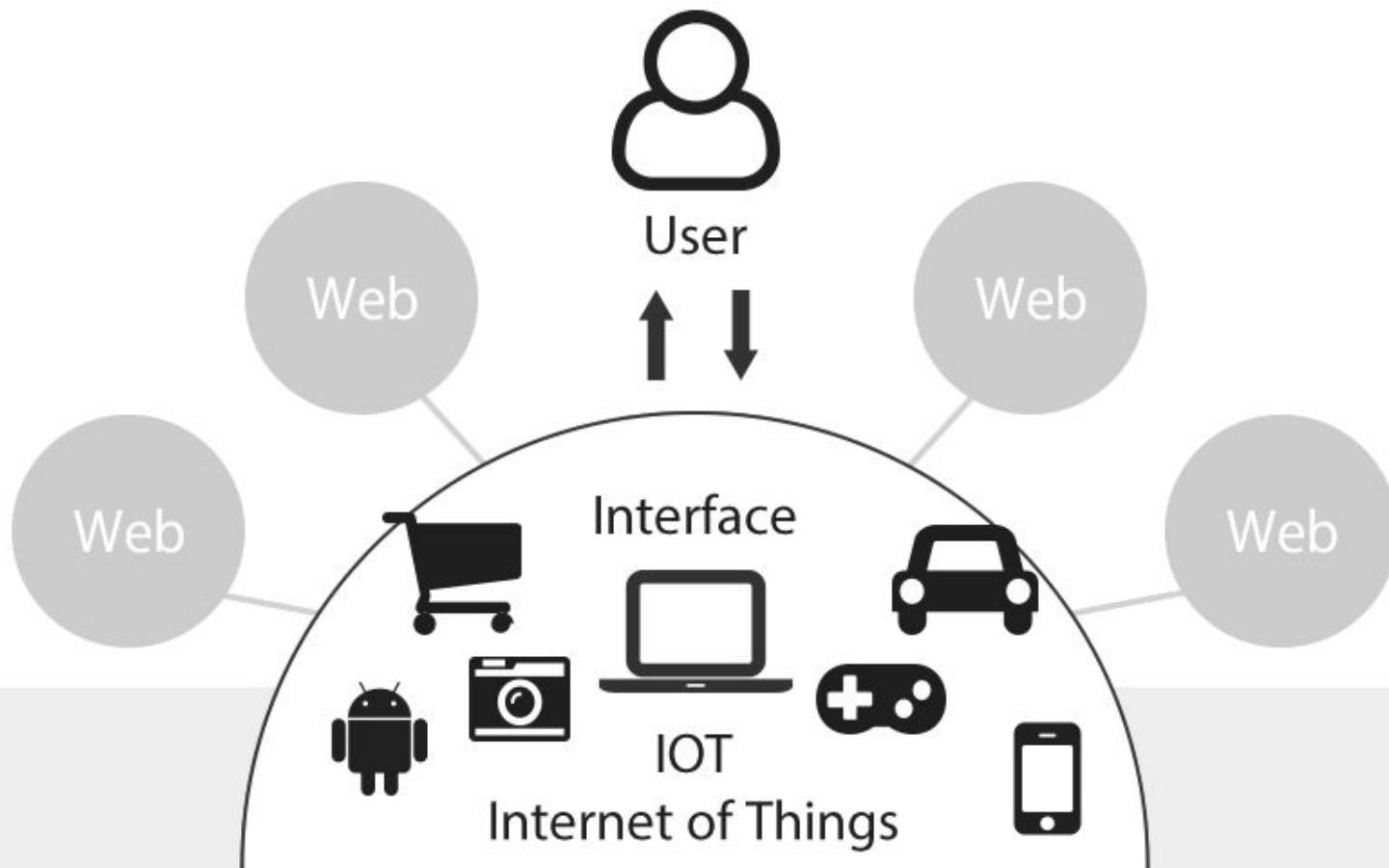
インタラクションデザイン

HCI (Human Computer Interaction)

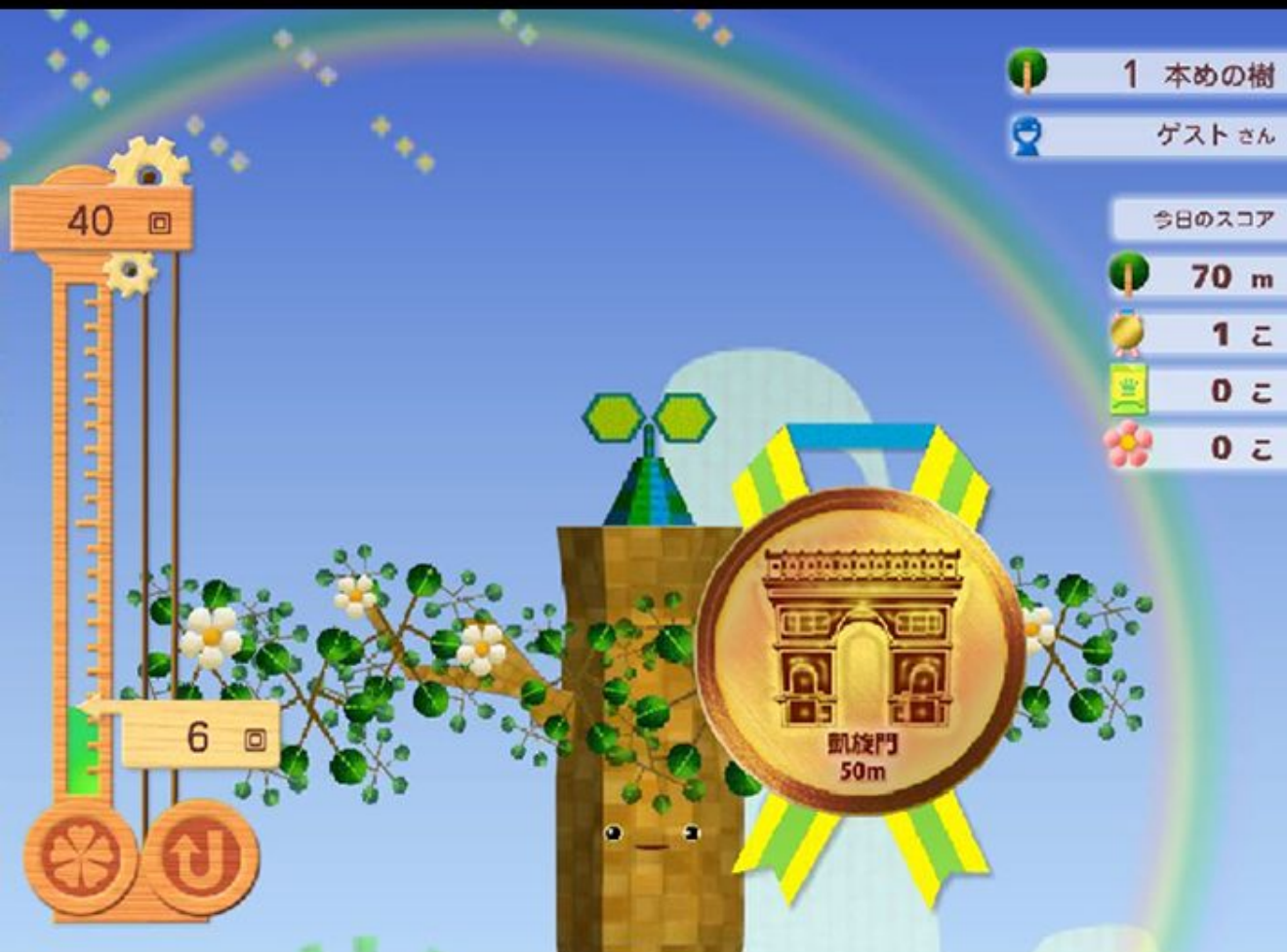


インタラクションデザイン

HCI (Human Computer Interaction)



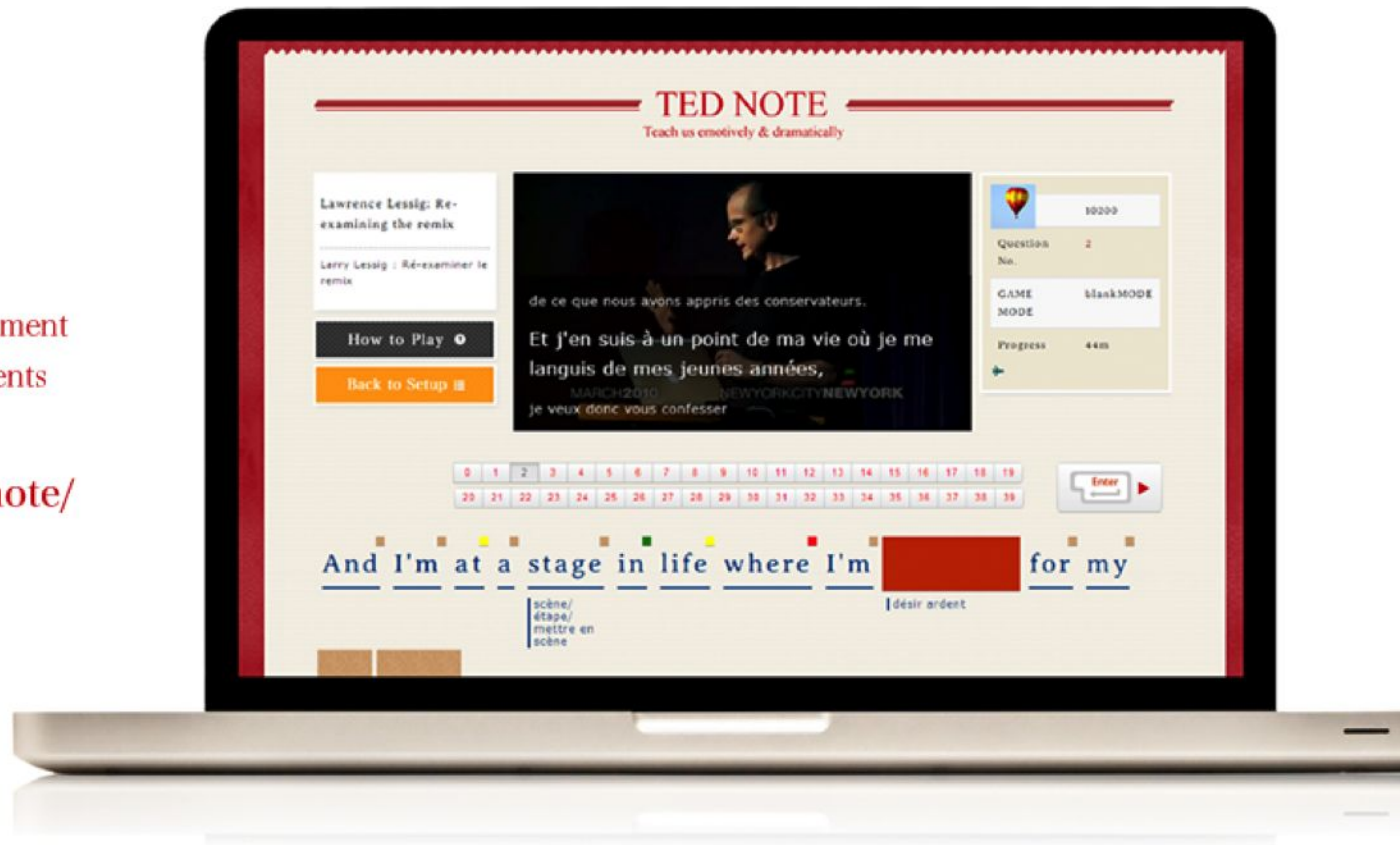
●ゲーミフィケーション 病院用リハビリテーション・ゲーム



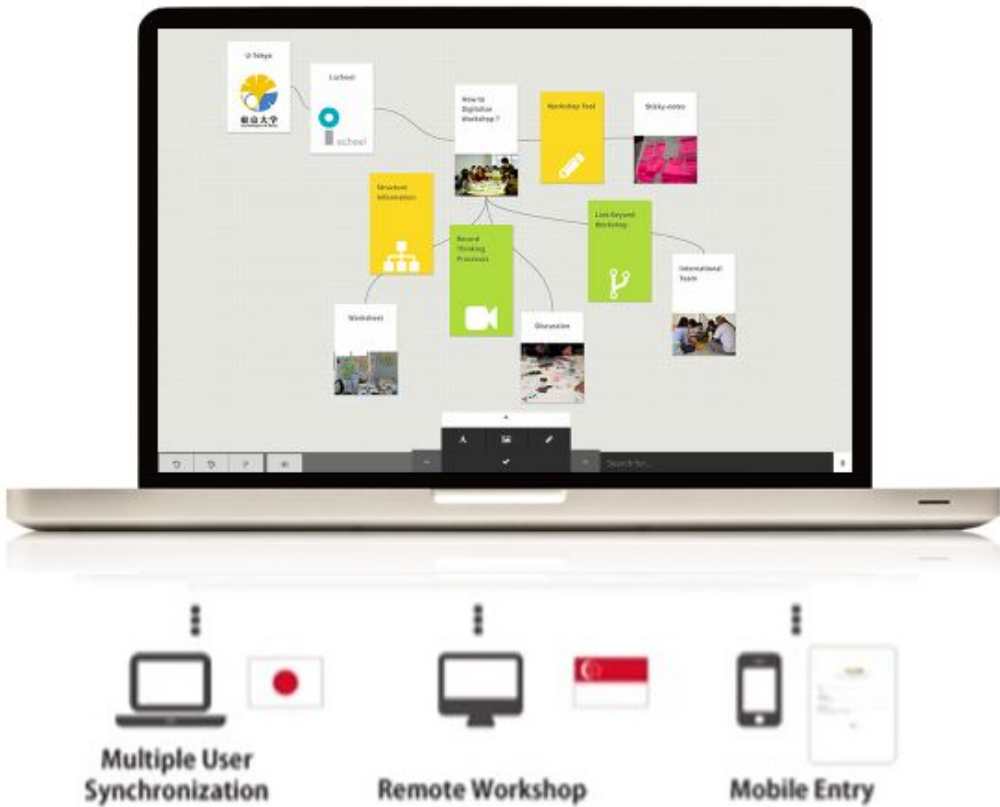
TED NOTE

E-learning system as auto-assignment
generator by utilizing open contents

www.connolab.com/tednote/



● 共創プラットフォーム デジタルポストイット



Webプログラミングの学び方

プログラミング学習の挫折経験 あるある 1

- 技術を一から理解しようとする、膨大な時間を擁する
- 技術が陳腐化しやすいプログラミング業界で、前者は危険
- 技術の栄枯盛衰が激しく、外的要因によってソリューションが変わる

プログラミングを学ぶ心構え 1

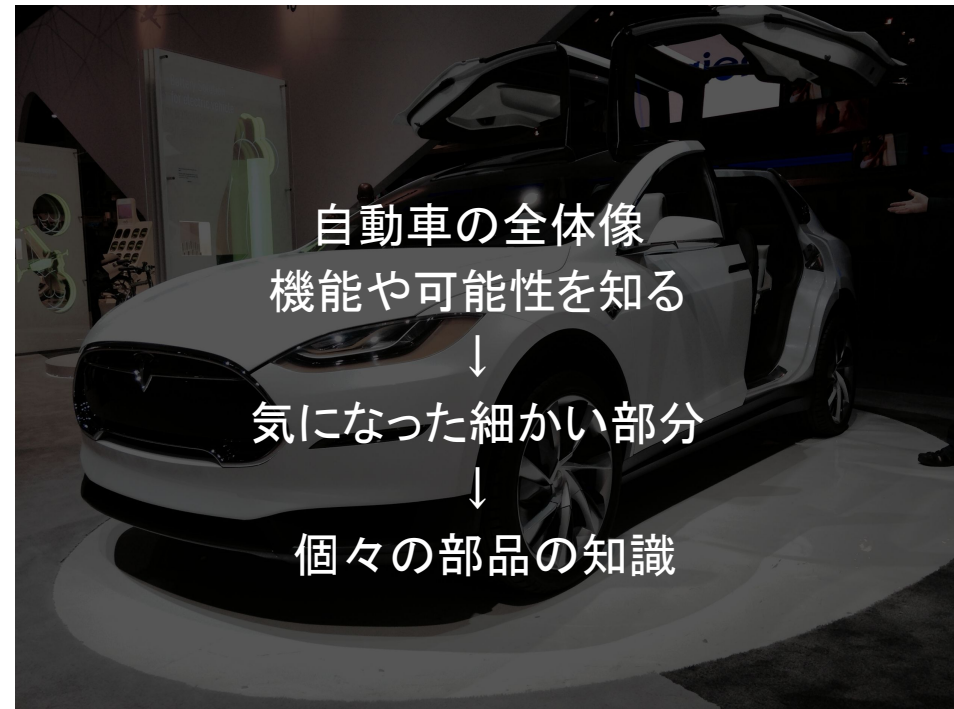
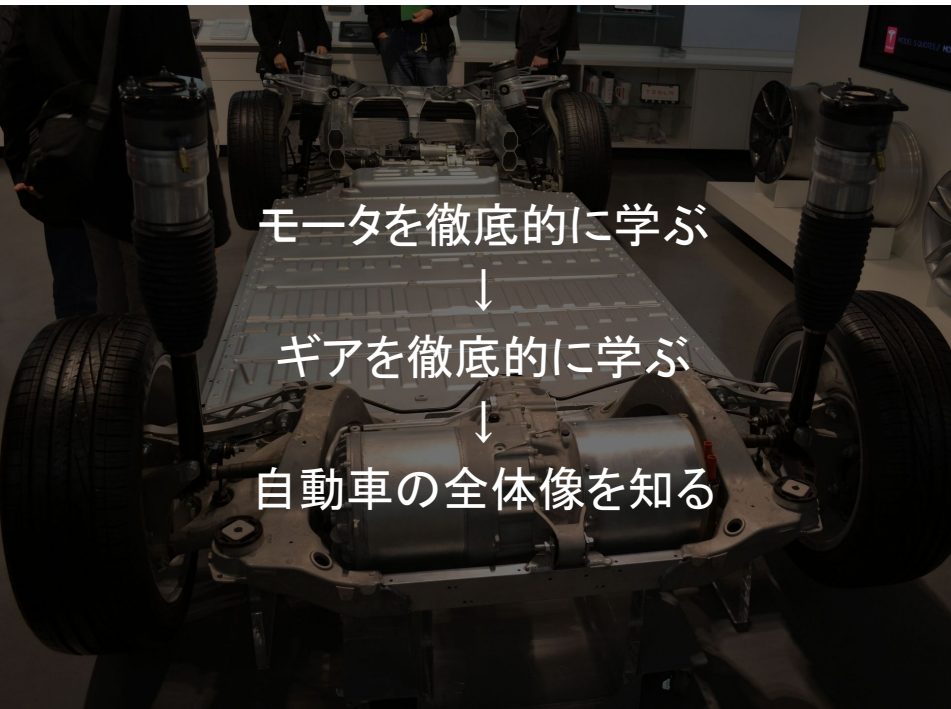
「わかった気になること」

- 繰り返し {突き当たった課題に対し、詳細化していく}
- 完璧主義に陥らない → ムリだとわかったら寝かせる

Webプログラミングの学び方

プログラミング学習の挫折経験 あるある 1

- 技術を一から理解しようとする、膨大な時間を擁する
- 技術が陳腐化しやすいプログラミング業界で、前者は危険
- 技術の栄枯盛衰が激しく、外的要因によってソリューションが変わる

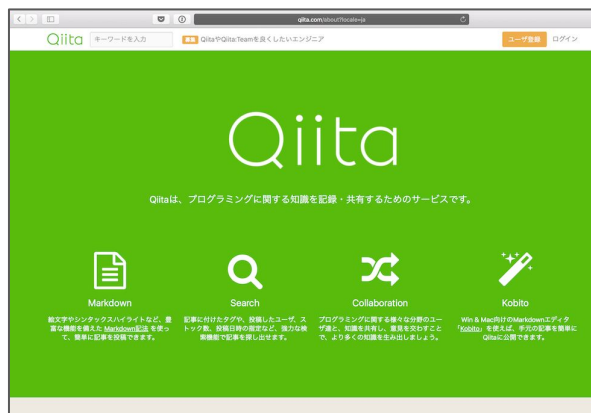


Webプログラミングの学び方

「オープン」の文化を活かした学習環境



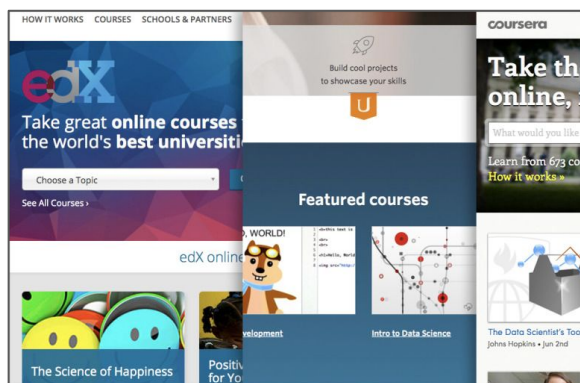
ドットインストール



Qiita



ハッカソン



edX / Coursera / FutureLearn

MOOC



Q&Aサイト



力試し

プログラミング学習の挫折経験 あるある 2

- 地味なミスで、プログラムが動かなくなる
- プレゼンの時に限って、プログラムが動かなくなる



プログラミング学習の挫折経験 あるある 2

- 地味なミスで、プログラムが動かなくなる
- プレゼンの時に限って、プログラムが動かなくなる

プログラミングを学ぶ心構え 2

「感動すること」

- 1つのコードが遠たら、自分を褒めること
- Web技術のもたらす可能性や文化に期待し、ワクワクすること

はじめに

知る

Webのしくみ

4/26

Webでメディア発信する方法
“世界を変える力”

5/10

Webの表現力
ユーザ体験のデザイン

5/17

ゲスト
海外と日本のトレンド

5/24

チーム開発について

5/31

ゲスト
IOTとデジタルファブリケーション

7/12

体験 する

Webアプリを作成

Webで発信するツールの作成

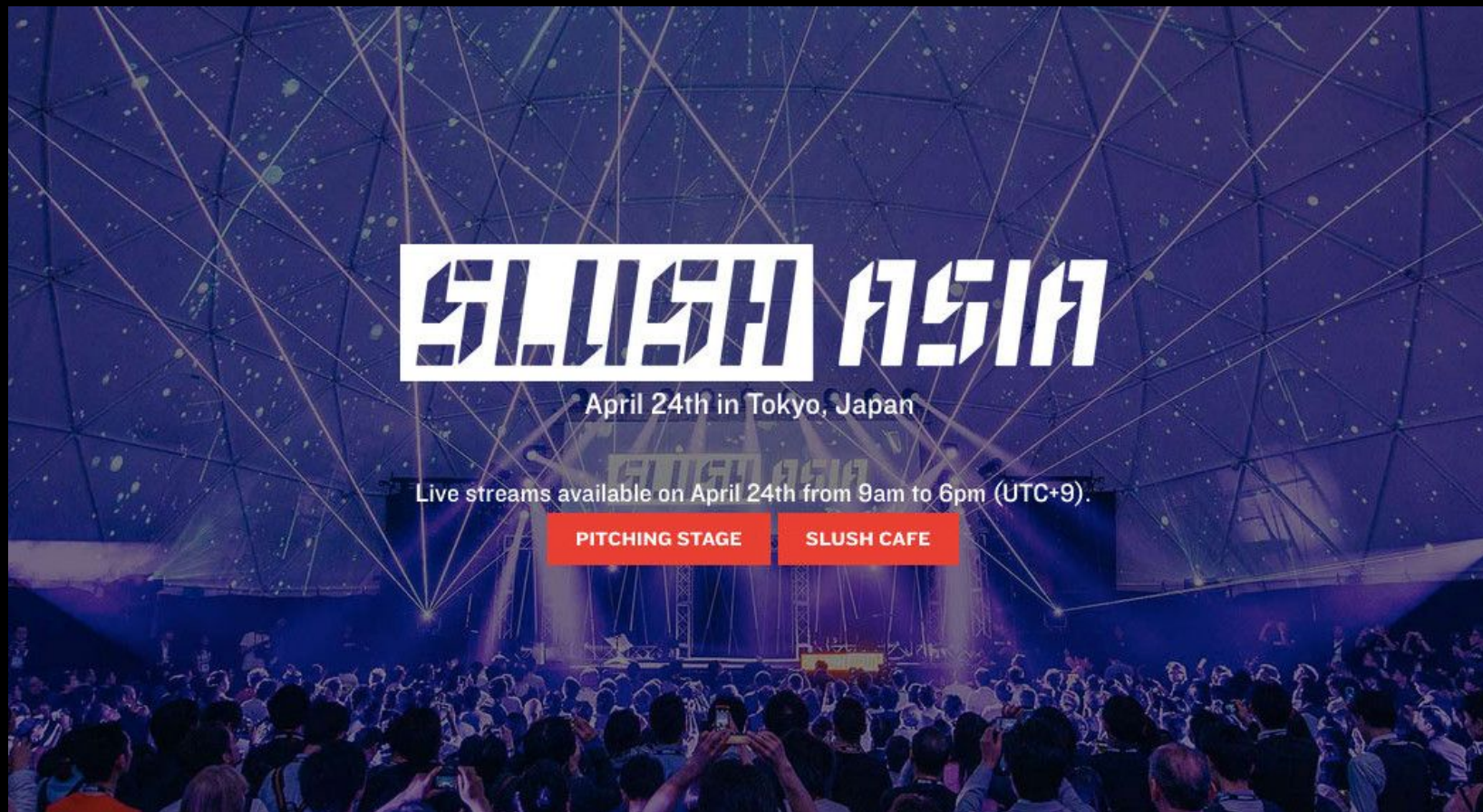
Webでリッチな体験をデザイン

ゲスト
Dentsu Lab Tokyo
SLASH ASIA

チームビルディング

ゲスト
馬場哲明先生(首都大)

ゲストスピーカー



SLUSH ASIA

世界最大級のスタートアップイベント

ゲストスピーカー



Dentsu Lab Tokyo

研究・企画・開発が一体となった”創りながら考える”チームテクノロジーを起点とした新しい表現開発

ゲストスピーカー



馬場 哲明 先生（首都大学東京）

IOT, プロトタイピング, デジタルファブリケーション

Web工学とは

世界を変える力

HTTP、Web Socket、
ブラウザ、HTML5、CSS、レ
ンダリング、Ajax、
SNS、CMS、Wiki、
Wordpress、スクリプト、Web
ブランディング、リッチクライ
アント、
モバイル、
セキュリティ

DB技術
NoSQL
API技術
JSON
Ajax
JavaScript

Webマイニング、リンク構造解
析、クローラー技術、パーサー
技術、DOM、クローリング戦
略、ビッグデータ解析、
機械学習、クラスタリング、協
調フィルタリング、PageRank、
分散処理、自然言語処理、ラン
キング、
ログ解析、感情分析

世界を知る力

コミュニケーションの発明するというこ

コミュニケーションにおける発明は、実際に世界を変えてきた

表 18-1 情報通信技術の発展

	採集経済社会	農業社会	工業社会	情報社会(1980 年代 ~)
情報蓄積	文字 木簡	紙、印刷 本	新聞、レコー ド、 CD	MD DVD
情報処理		算盤	機械式 計算機	コンピュータ パソコン HD、FD、CD 銀行などのデータ通信 インターネット
通信	言葉、のろし	早馬、飛脚	郵便制度 電信・電話 ラジオ、テレビ	
				携帯電話

<https://ja.wikipedia.org/wiki/情報技術史>

Kahn Academy Journey into Information Theory <http://p.tl/wLX>

ニコニコ動画【情報理論】コミュニケーションの歴史と科学 [nicovideo.jp/watch/sm21811249](https://www.nicovideo.jp/watch/sm21811249)

コミュニケーションの発明

さて、みなさんが手にする
コンピューターというメディア...



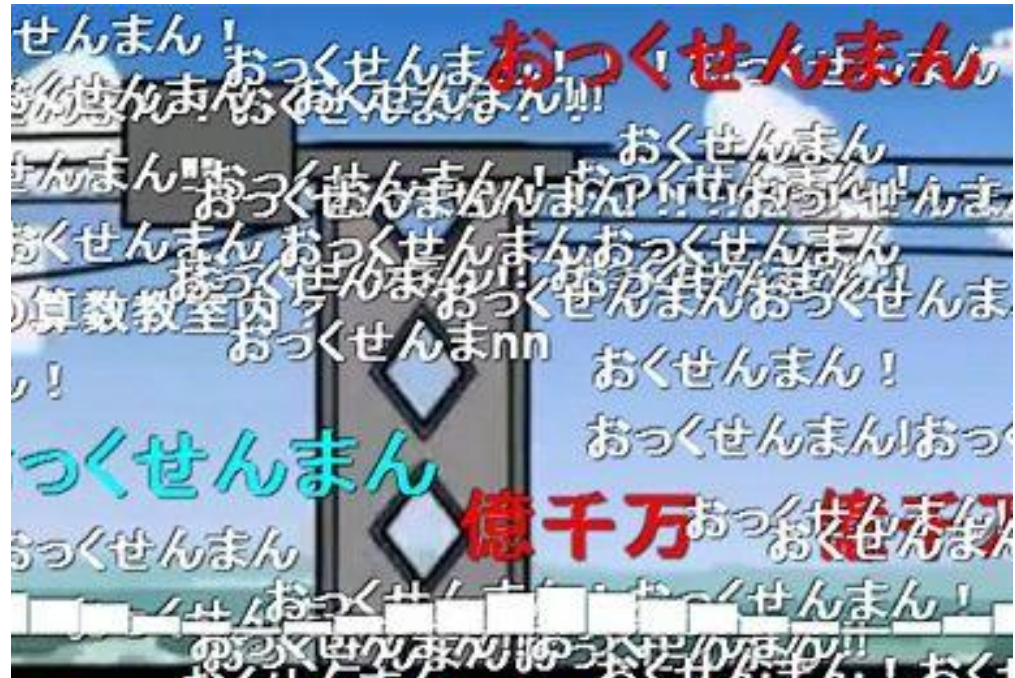
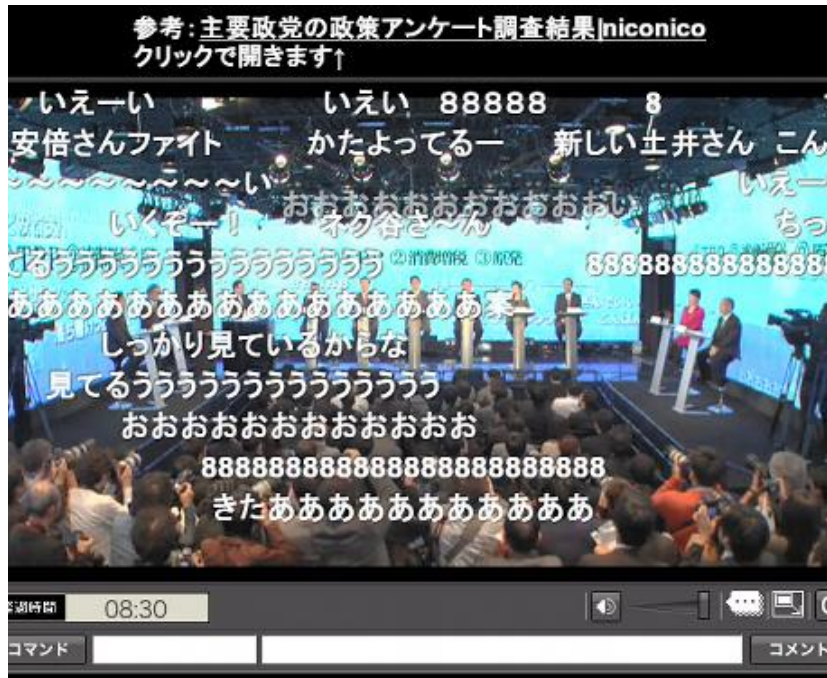
コミュニケーションの発明



クラウドおばあちゃん

Granny Cloud

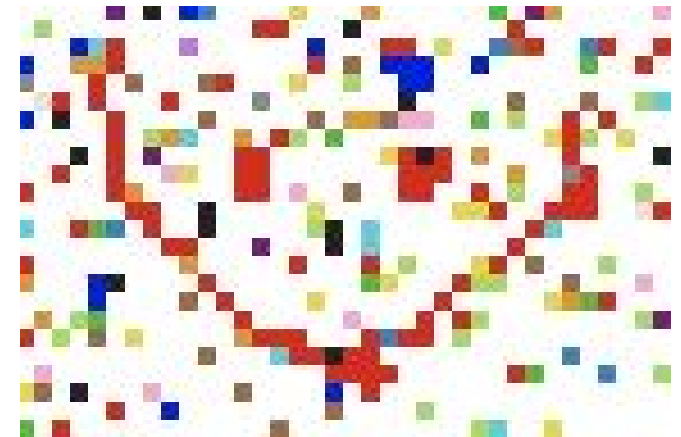
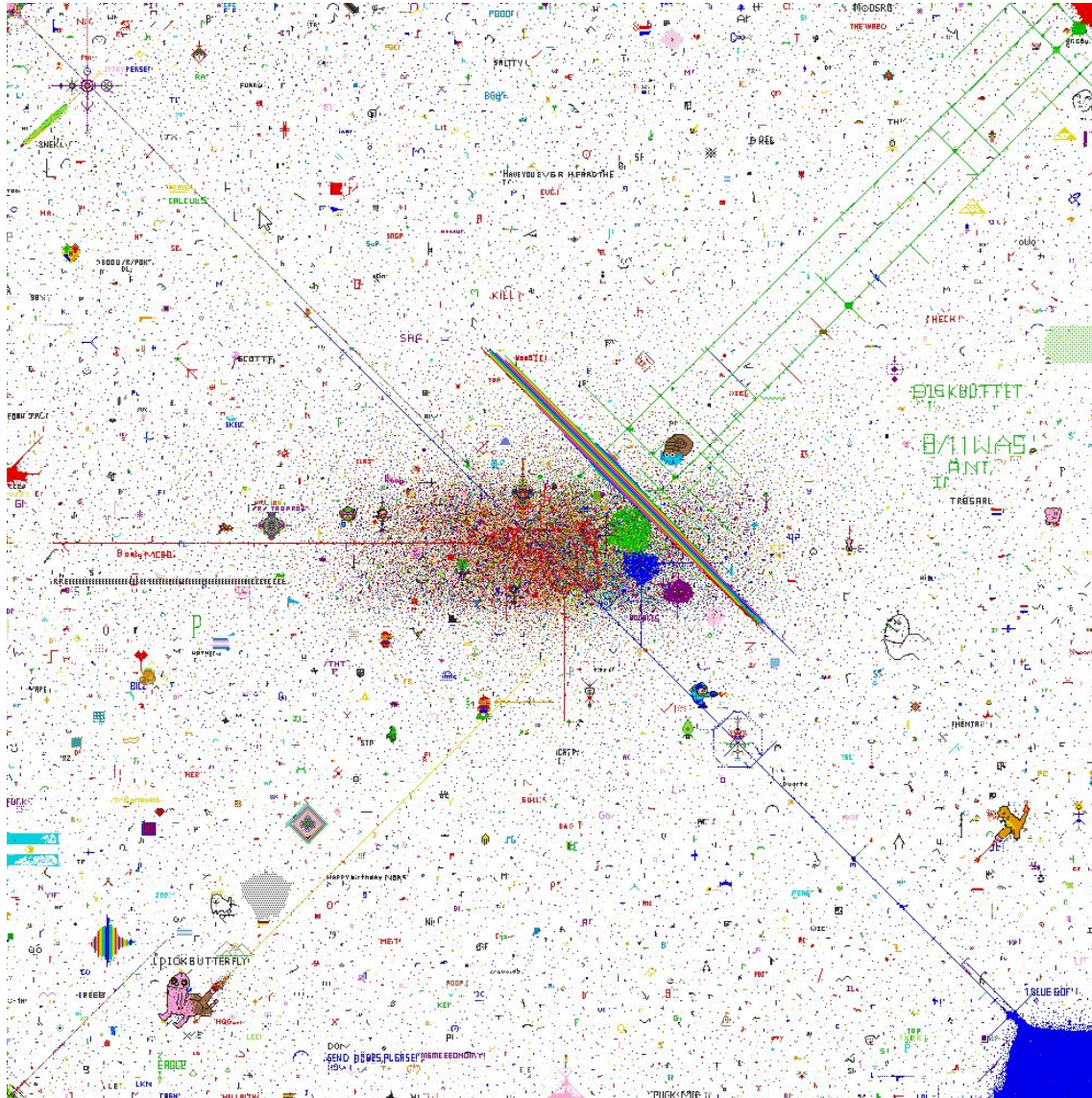
コミュニケーションの発明



コミュニケーションの発明

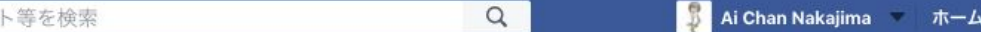
reddit-place

<http://sudoscript.com/reddit-place/>



コミュニケーションの発明

Facebook「いいね」



IDPW「どうでもいいね」



インターネット中に溢れる「いいね！」ボタン。増えすぎちゃってそろそろ溺れてしまいそう。いいも悪いも義理もとおきも全部まとめて「いいね！」したい。そんなあなたにお届けする“どうでもいいね！”ボタンはワンクリックで画面上に表示されている「いいね！」を自動で、だいたい全部をクリックしてくれる※1、Google Chrome機能拡張です。

※1 一度に大量に「いいね！」すると、警告を受ける場合があります。「いいね！」はほぼほぼに。また、実行しても失敗することがありますが、IDPWでは本機能拡張に対しての責任を一切負いません、自己責任でおねがいします。

700

✓ いいね！

INSTALL

【演習】

初級演習

初心者

チャレンジ演習

上級者

【演習】

初級演習

講義

: Webのしくみ

演習

: HTTP

(curlコマンド)

チャレンジ演習

connolab.com/weben1/img/

におかれた画像

- [photo1.jpg](#)
- [photo2.jpg](#)
- [photo3.jpg](#)

をcurlで一行で取得しなさい

connolab.com/weben1/apply.php

へポストデータを送りなさい。

POSTデータの変数

- `sirname`
- `firstname`

HTML (HyperText Markup Language)

HTMLと構成するファイルを見てみる

Mac ⌘ + Option + U

Win control + U

Mac ⌘ + F

Win control + F

.img
.css
js

Webのしくみ

▼ブラウザ無しの場合

Web サーバ



コード



Web クライアント



コードを送られてきても
読めないよ

▼ブラウザ有りの場合



ブラウザがコードを
読み取り、綺麗に
表示してくれる!

Webのしくみ

URL

http://Webサーバ名:ポート番号/パス/

プロトコル

ドメイン

ディレクトリ

🌐 ホームページはなぜ見える？（簡易図）

WebブラウザにURLを入力



1

①URLの場所 (WEBサーバー) を探す

2

②webサーバーに保存されているデータが
webブラウザの画面に表示される

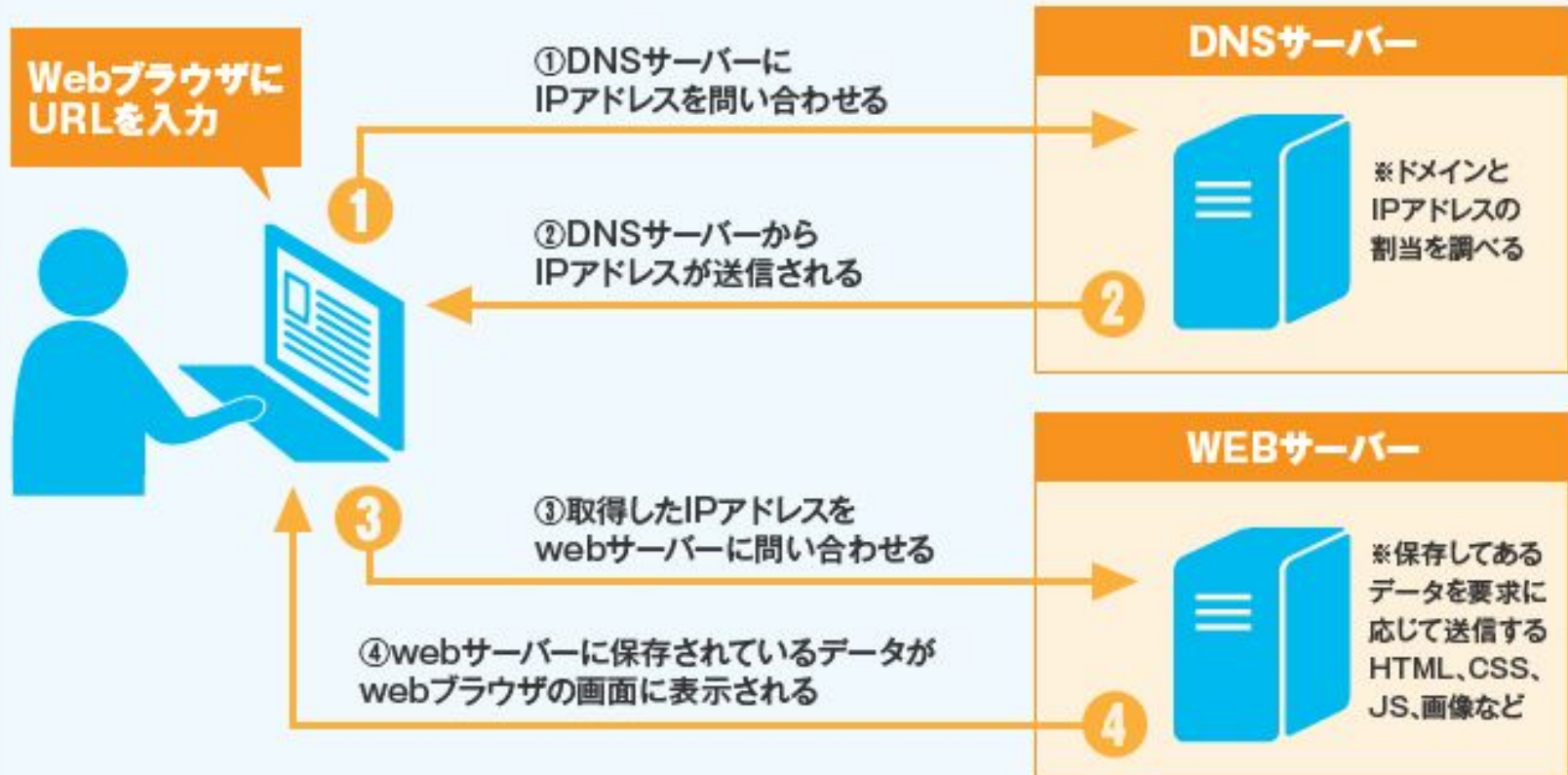
WEBサーバー



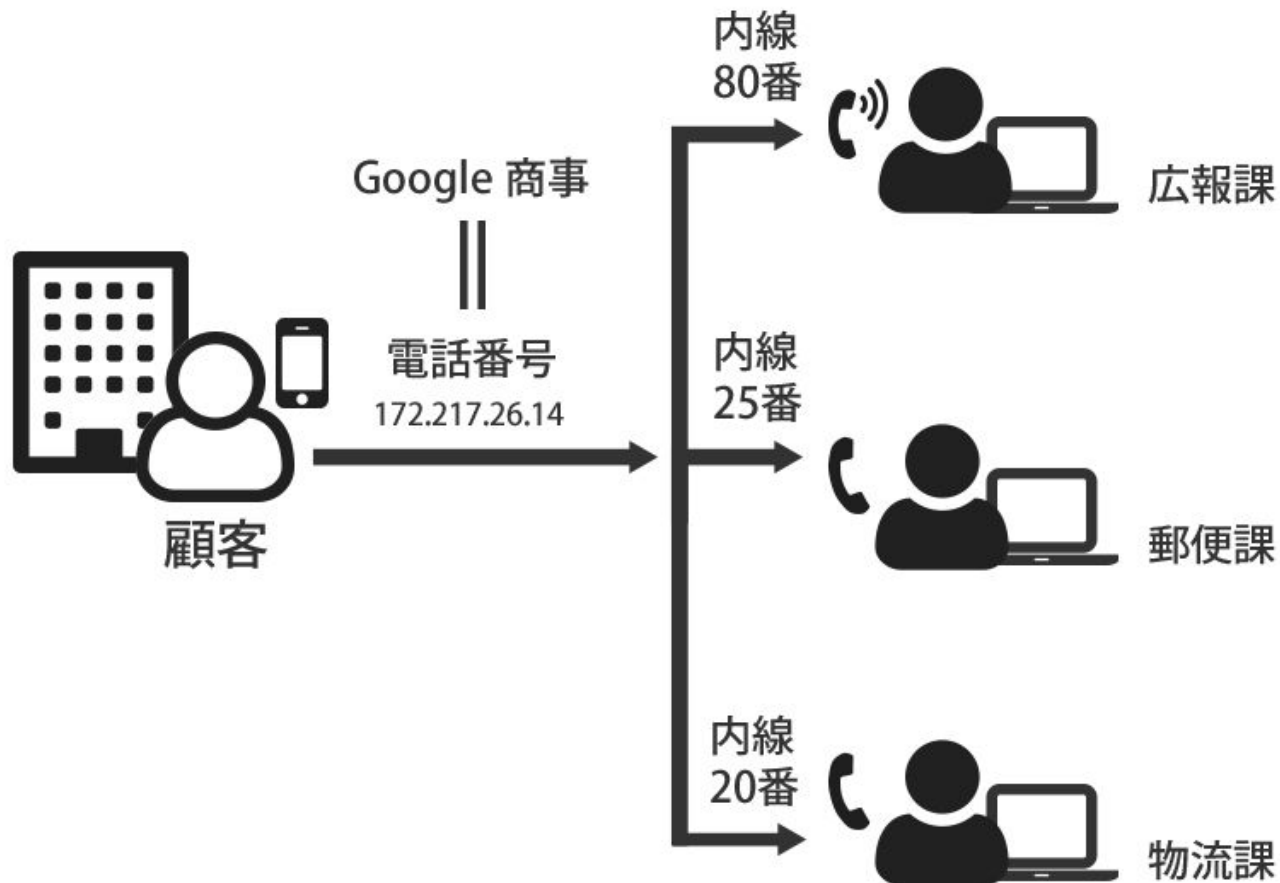
※保存してある
データを要求に
応じて送信する
HTML、CSS、
JS、画像など

Webのしくみ

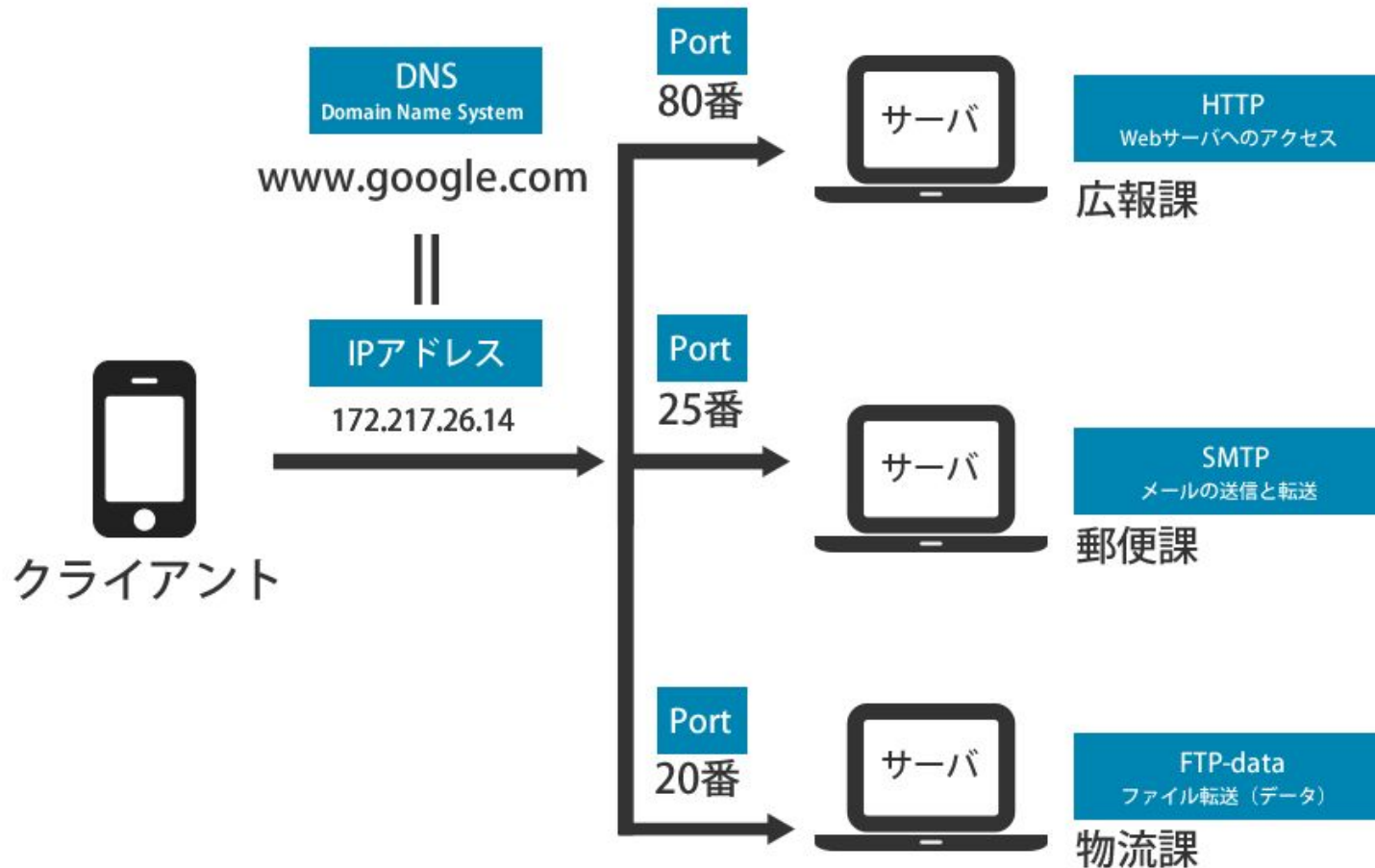
🌐 ホームページが見える仕組み(詳細図)



IPアドレスとポート番号



IPアドレスとポート番号



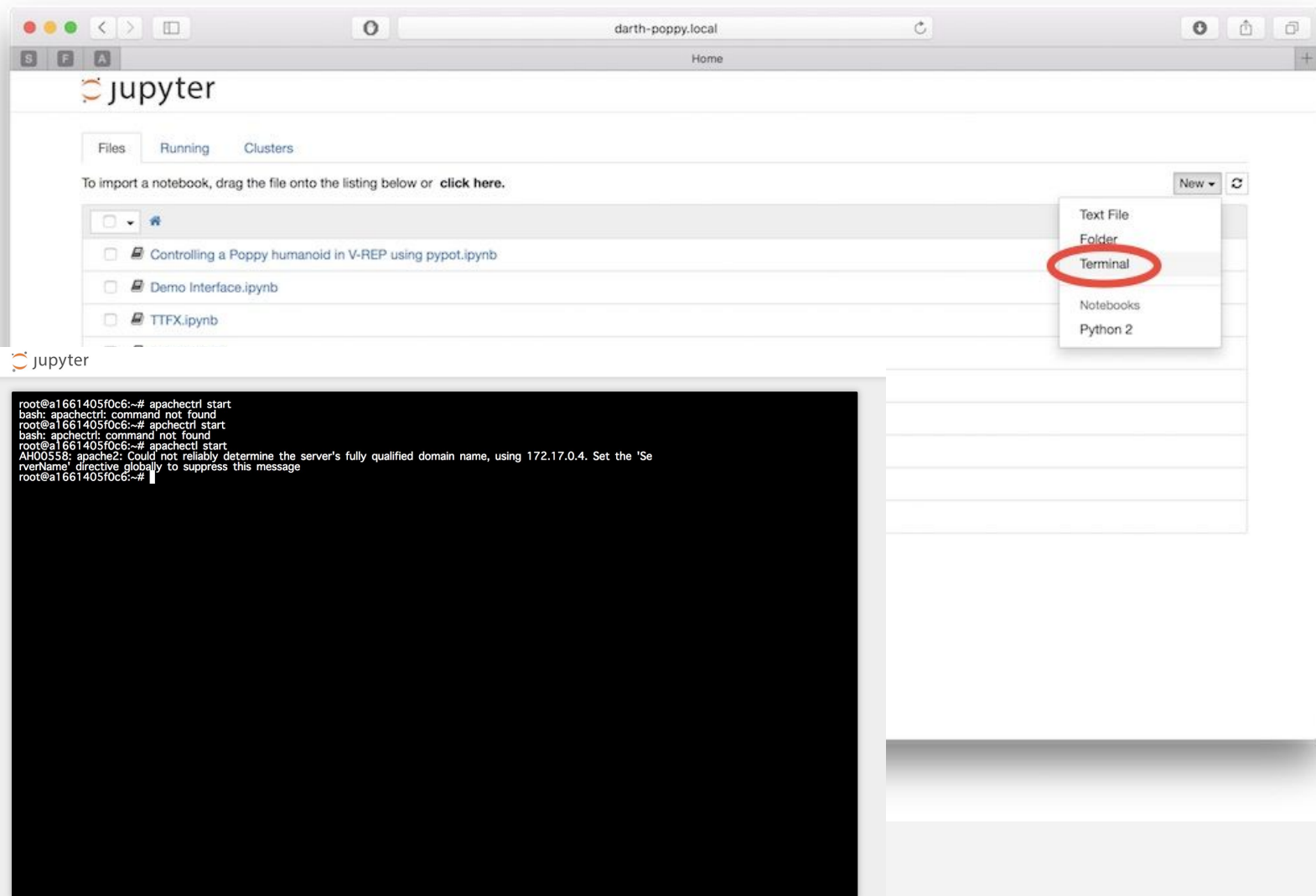
Webのしくみ

ポート群	状態	0 ～ 65535
WellKnown ポート番号	メジャーなサービスやプロトコルが利用するために予約されたポート番号。	0 ～ 1023
Registered ポート番号	特定のアプリケーションなどが使用するポート郡。IANAが登録を受け付け公開。	1024 ～ 49151
その他 ポート番号	ユーザが自由にとりあつかえるポート番号。登録の必要なし。	49152 ～ 65535

ポートを閉じる
↓
Firewall

ポート	機能	名称
20	ファイル転送(データ)	ftp-data
21	ファイル転送(制御)	ftp
23	ネットワークを経由した他コンピュータの操作	telnet
25	メールの送信と転送	smtp
53	コンピュータ名からIPアドレスへの変換	domain
80	Webサーバへのアクセス	http
110	メールの読み出し	pop3
123	コンピュータの時計あわせ	ntp
443	Webサーバへのアクセス(暗号化あり)	https

【演習】Webのしくみ



【演習】Webのしくみ

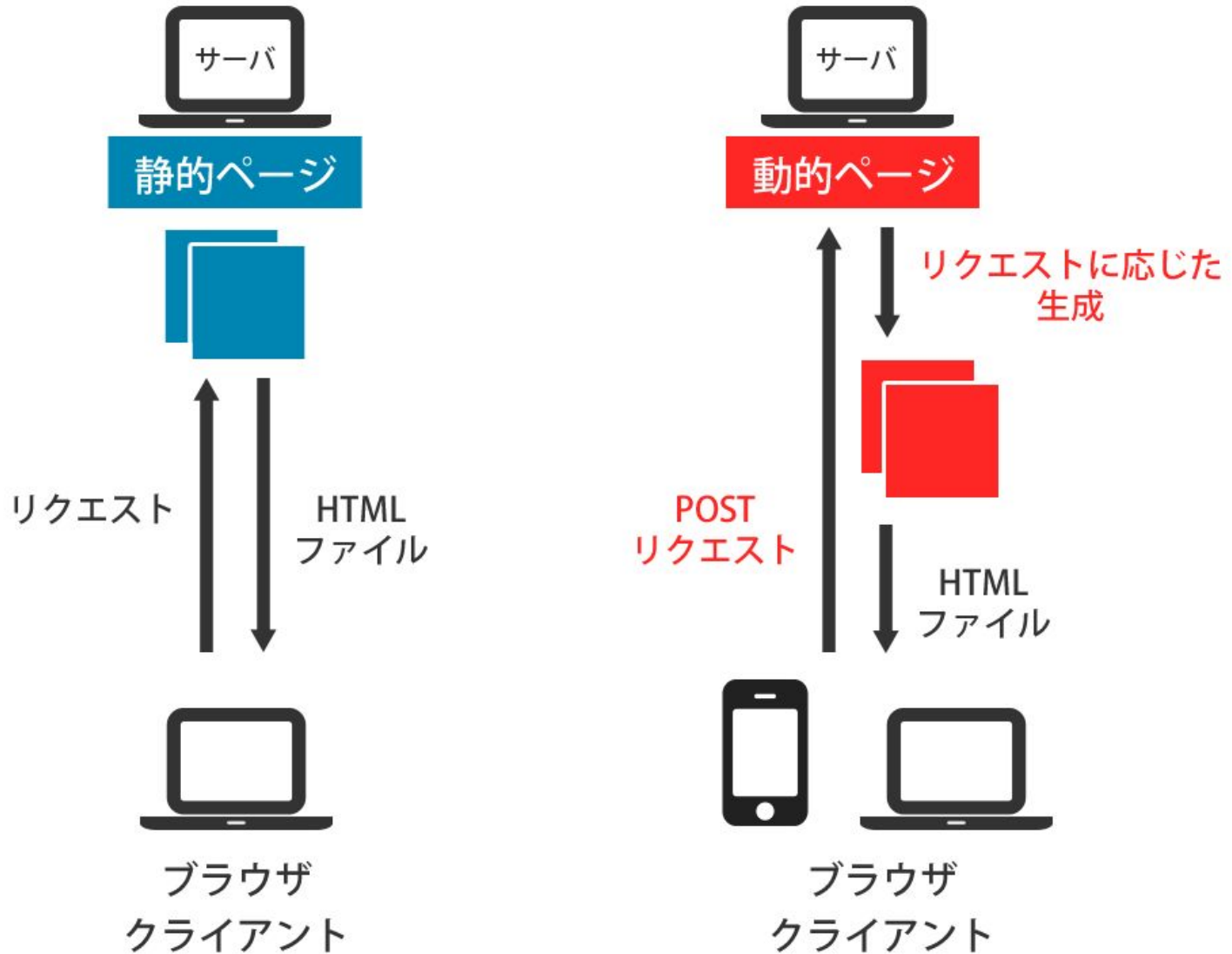
意味	ステータスコード	結果フレーズ
正常	200	OK
リクエストがおかしい	400	Bad Request
ログインに失敗した	401	Unauthorized
読み出す権利がない	403	Forbidden
見つからない	404	Not Found
サーバ内部にエラー発生	500	Internal Server Error

区分	フィールド名	意味
Webブラウザの基本情報	Authorization	認証のためのユーザ名とパスワード
	Host	接続先のWebサーバ名
	Referer	直前に見ていたページ
	User-Agent	Webブラウザの種類
取り出し要求に関する情報	Data	日付情報
	Connection	接続状態の管理
やりとりする内容に関する情報	Content-Type	やりとりする情報の種類
	Content-Length	やりとりする情報のサイズ
	Last-Modified	最終更新日

【演習】Webのしくみ

メソッド	機能
HEAD	情報本体ではなく概要を取り出す
GET	情報をWebサーバから取り出す
POST	情報をWebサーバに送り込む
PUT	Webサーバ上の情報を書き換える
DELETE	Webサーバ上の情報を削除する
TRACE	動作を確認するための情報を得る
OPTIONS	使用できるメソッドの一覧を得る
CONNECT	プロキシと呼ばれる特殊なサーバへの指示

静的Webコンテンツと動的Webコンテンツ



【演習】HTMLとCSSの基礎

初級演習

- apacheを起動し
Hello Worldをします
- CSSにプロパティを記入し
外観を変えてみます
- サンプルファイルをCurlで保存し
HTMLとCSSの各要素がどの様に関
連しているか観察します

チャレンジ演習

- 下記URLと同じ外観で
ローカル環境にHTMLとCSSを作成し
なさい
- <http://connolab.com/weben1/>
- 後ほどIpythonにアップロード

サーバ側 の 動的Web(レガシーWebアプリともいう)

動的Webに必要な環境

Webサーバ上でプログラムを実行させる環境

- PHP
- Perl
- Java
- python
- Node

今回は Pythonの軽量なWebフレームワーク Flask
を使ってみます

【演習】Pythonの基礎

初級演習

- JupyterNotebook の使い方を学びます
- Pythonの基本を簡単におさらいします
- 演習問題1に挑戦します

チャレンジ演習

- 各演習問題をやってみる

【応用演習】アプリの作成

初級演習

- Flaskで Hello Worldします
- ページ遷移とテンプレートを学びます
- GETとPOSTに挑戦します

チャレンジ演習

- app.pyやテンプレートをいじって簡単な Flaskアプリを作ってみる