

みんなのPython勉強会 #14

2016.7.6

いま、ふたたびの Web入門

辻 真吾

@tsjshg

自己紹介

- ❖ 1975年生まれ
- ❖ 都内のとある大学で研究やっております
 - ❖ 研究室のテーマは癌とゲノム
 - ❖ 私がやっているのはPythonでデータ解析
- ❖ データ解析が専門と思われがちですが、IT系ベンチャーでWebアプリ作っていたことがあります！
- ❖ <http://www.tsjshg.info/>

Udemy

コース検索

udemy

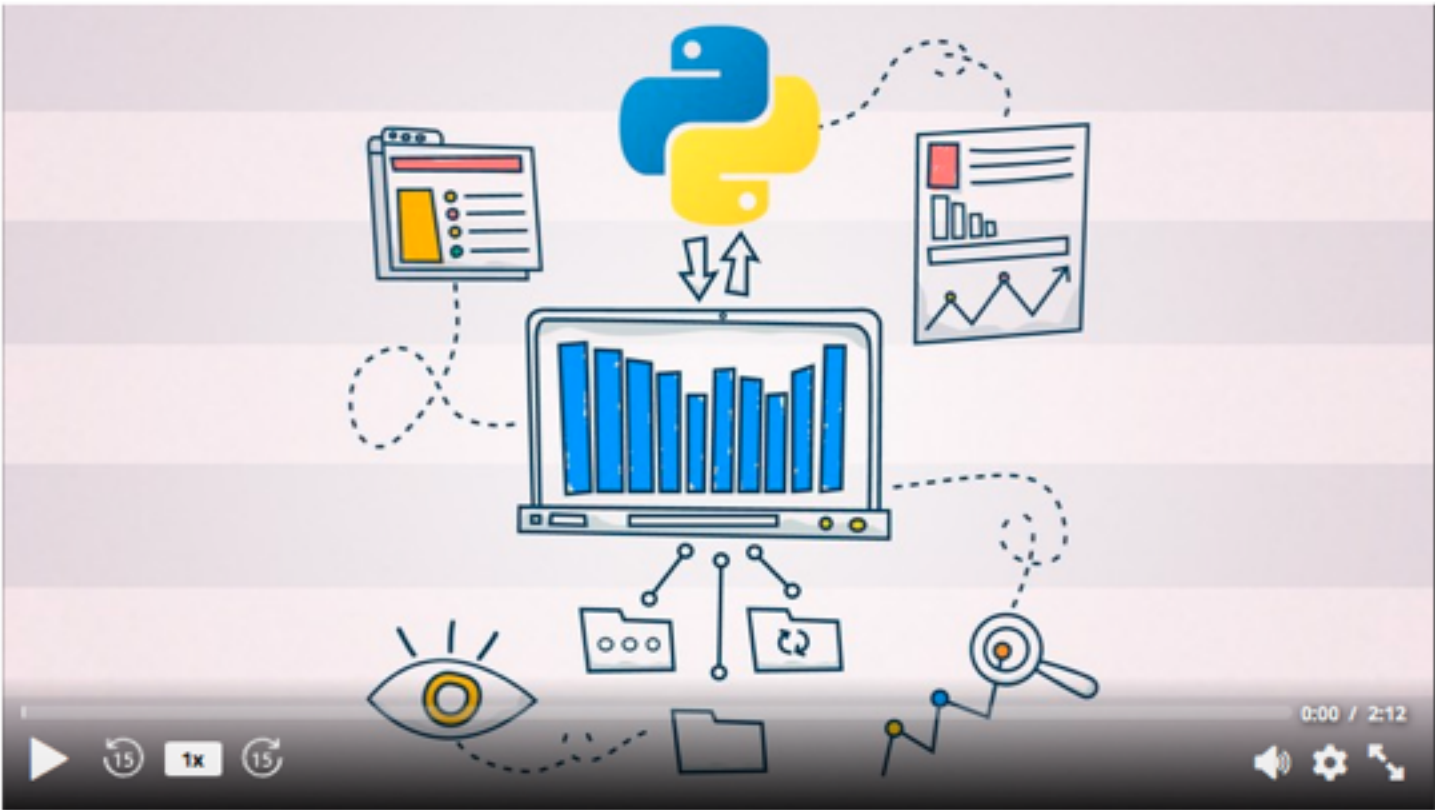
講師になりたい方へ

【世界で2万人が受講】 実践 Python データサイエンス

データ解析の基本、可視化、統計、機械学習などデータサイエンスに関するあらゆる実践的なスキルがPythonで身に付く！

★★★★★ 182 評価、1683 生徒が登録済み

講師： Shingo Tsuji, Jose Portilla 開発 / プログラミング言語



0:00 / 2:12

¥6,000

このコースを受講する

クーポンを利用する
無料プレビューを開始する
その他のオプション▼

レクチャー	104
ビデオファイル	17.5 時間
スキルレベル	すべてのレベル
言語	日本語
その他：	学習期限なし 30日間返金保証 iOS・Androidどちらも受講可能 修了証明書

♡ ほしい物リスト

2016/8/7までの割引クーポンです（先着40名様）

<https://www.udemy.com/python-jp/?couponCode=stapy0706-30D>

Sir Tim Berners-Lee

- ❖ Webの発明者
- ❖ 2012年ロンドン五輪の開会式でパフォーマンス
- ❖ 欧州原子核研究機構（CERN）で働いていた1989年から90年にかけてWorld Wide Webの基礎を作る



インターネットとWeb

1960

1970

1980

1990

2000

2010

Web (HTTP)

電子メール (SMTP)

ARPAnet

インターネット

Webの構成要素

- ❖ 通信の方法

- ❖ HTTP (Hyper Text Transport Protocol)

- ❖ 情報の記述

- ❖ HTML (Hyper Text Markup Language)

- ❖ 情報の場所

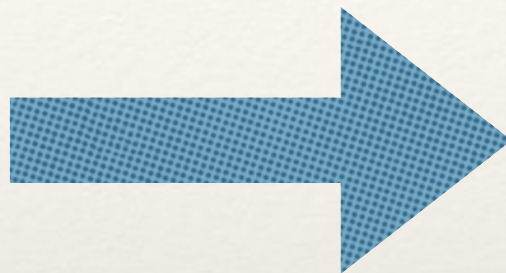
- ❖ URL (Uniform Resource Locator)

Webは超シンプル



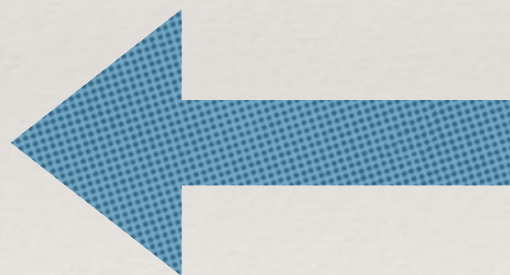
Webブラウザ
(クライアント)

①



HTTPリクエスト

②



HTTPレスポンス



HTTPサーバ
(サーバ)

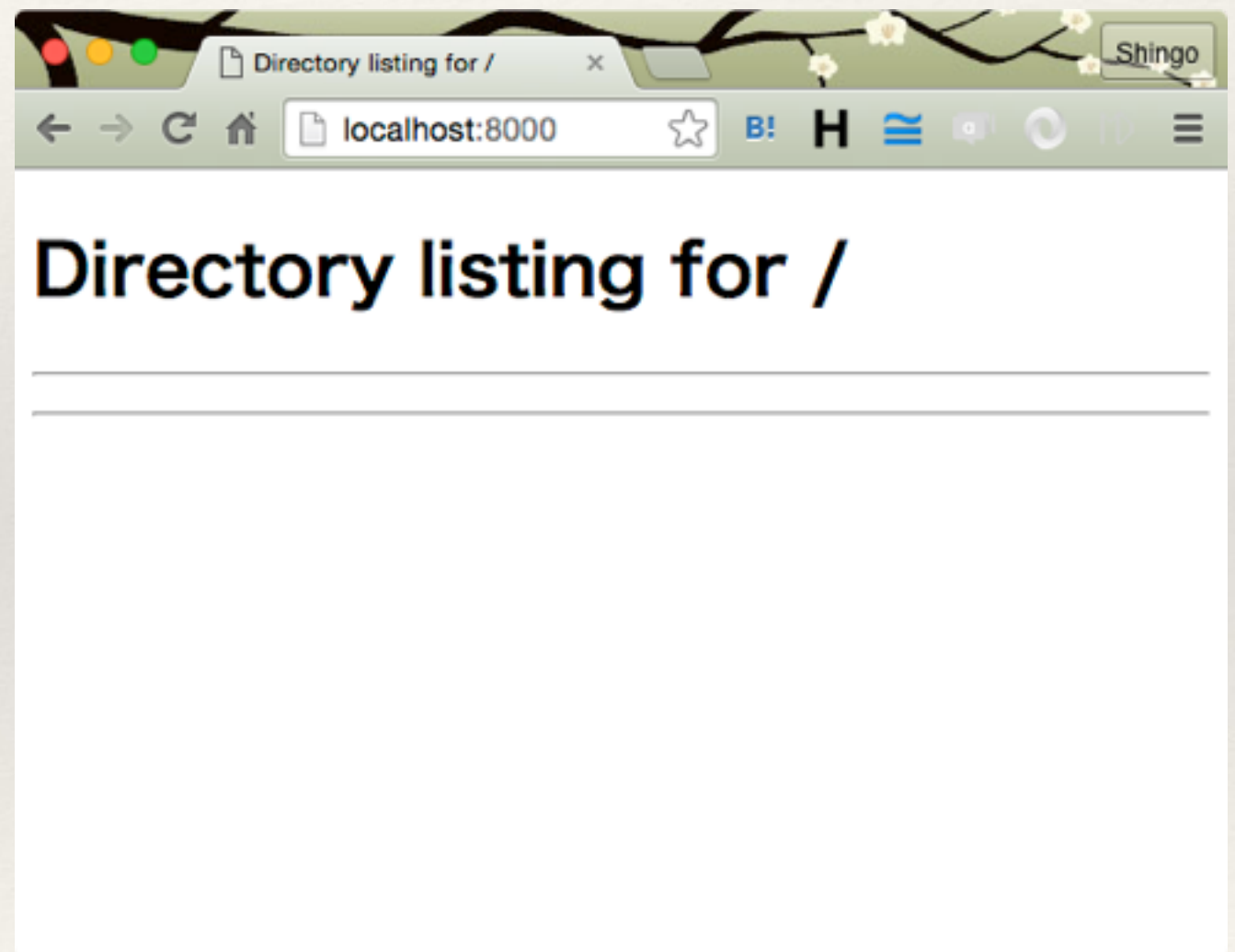
HTTPサーバあります

```
$ python -m http.server
```

```
Serving HTTP on 0.0.0.0 port 8000 ...
```

OSのシェルからhttp.serverを直接実行
カレントディレクトリのファイルを配信

http://localhost:8000/



HTMLを書こう

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyper Text</title>
</head>
<body>
この勉強会は、<a href="http://www.leadinge.co.jp/">リーディング・エッジ社</a>の協賛を得ています。

</body>
</html>
```



Webブラウザが、imgタグを見て、別のサーバへHTTPのGETリクエストを出す

HTTP通信の実体

Webブラウザが

http://localhost:8000

にアクセスする動作を手動でやってみる

```
import telnetlib
tn = telnetlib.Telnet('localhost', 8000)

tn.write(b'GET / HTTP/1.1\r\n\r\n')

res = tn.read_all()
print(res.decode('utf-8'), end='')
```



HTTPリクエスト

```
HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.4.3
Date: Sun, 03 Jul 2016 09:31:06 GMT
Content-type: text/html
Content-Length: 105
Last-Modified: Sun, 03 Jul 2016 09:24:38 GMT
```

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyper Text</title>
</head>
<body>
  この勉強会は、<a href="http://www.leadinge.co.jp">
<img src="http://www.leadinge.co.jp/images/log
</body>
</html>
```

HTTPレスポンス

HTTPレスポンス

```
HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.4.3
Date: Sun, 03 Jul 2016 09:31:06 GMT
Content-type: text/html
Content-Length: 105
Last-Modified: Sun, 03 Jul 2016 09:24:38 GMT
```

スタートライン (リクエスト・ステータスライン)

ヘッダ

空行

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyper Text</title>
</head>
<body>
この勉強会は、<a href="http://www.leadinge.co...
<img src="http://www.leadinge.co.jp/images/logo...
</body>
</html>
```

ボディ

HTTPレスポンス

HTTP リクエスト

```
GET / HTTP/1.1
```

```
Accept: text/html,application/xhtml+xml,...  
Accept-Language: ja,en-US;q=0.8,en;q=0.6  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac...
```

スタートライン (リクエスト・ステータスライン)

ヘッダ

空行

POSTのときはデータが入ります

POSTのとき

ボディ

key0=value0

key1=value1



GETのとき

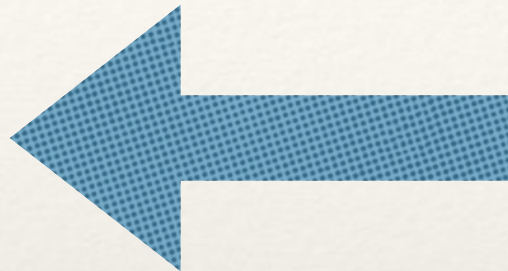
`http://localhost:8000/page?key0=value0&key1=value1`

HTTP通信の困った特徴

- ❖ ステートレス
 - ❖ 毎回、自己紹介から始める社内会議
- ❖ 通信経路が暗号化されない

HTTP Cookie

Cookieを受け取る時

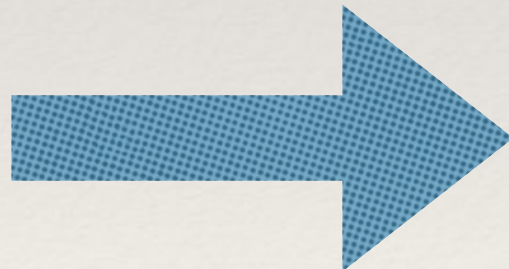


HTTPレスポンスのヘッダ

Set-Cookie: _name="my cookie"; expires=Wed, 06 Jul 2016 01:25:56 GMT



以降、同じURLのホストにアクセスする時



HTTPリクエストのヘッダ

Cookie: _name="my cookie"



公開鍵暗号

普通の暗号



秘密の鍵を共有してデータを暗号化

公開鍵暗号



秘密鍵



公開鍵で暗号化すると、
秘密鍵でしかもどらない



公開鍵



なぜ、そんなことが？

6 を素因数分解してみよう

$$2 \times 3$$

じゃあ、549156609299は？

$$660599 \times 831301$$

大雑把に言うと

公開鍵

秘密鍵

まとめ

- ❖ Webを構成する要素
 - ❖ HTTP、HTML、URL
- ❖ HTTP通信は超シンプル
- ❖ HTTP Cookieと通信経路の暗号化が現代のWebを支える
- ❖ 「Webを支える技術（山本陽平）」技術評論社
 - ❖ もっと知りたい方へ。すごい良著です。