



# 福井大学 リカレント 教育プログラム プログラミング応用

---

福井工業高等専門学校

電子情報工学科

斉藤徹



# 自己紹介

福井工業高等専門学校 電子情報工学科 齊藤 徹



- 1990 頃 パソコン通信にはまる...
- 1995年 福井高専のホームページを立ち上げ  
Windows 95 インターネットの普及開始
- 2003年～ BLOG で情報発信
- 2006年～現在 丹南地区緊急連絡システム運用
- その他
  - <https://tsaitoh.net/>
  - <https://twitter.com/tohrusaitoh/>
  - <https://www.facebook.com/tsaitoh.net/>
  - [t-saitoh.net@gmail.com](mailto:t-saitoh.net@gmail.com)

@TohruSaitoh

Facebook

Google

# プログラミング応用の目標

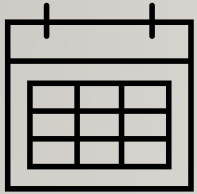
---



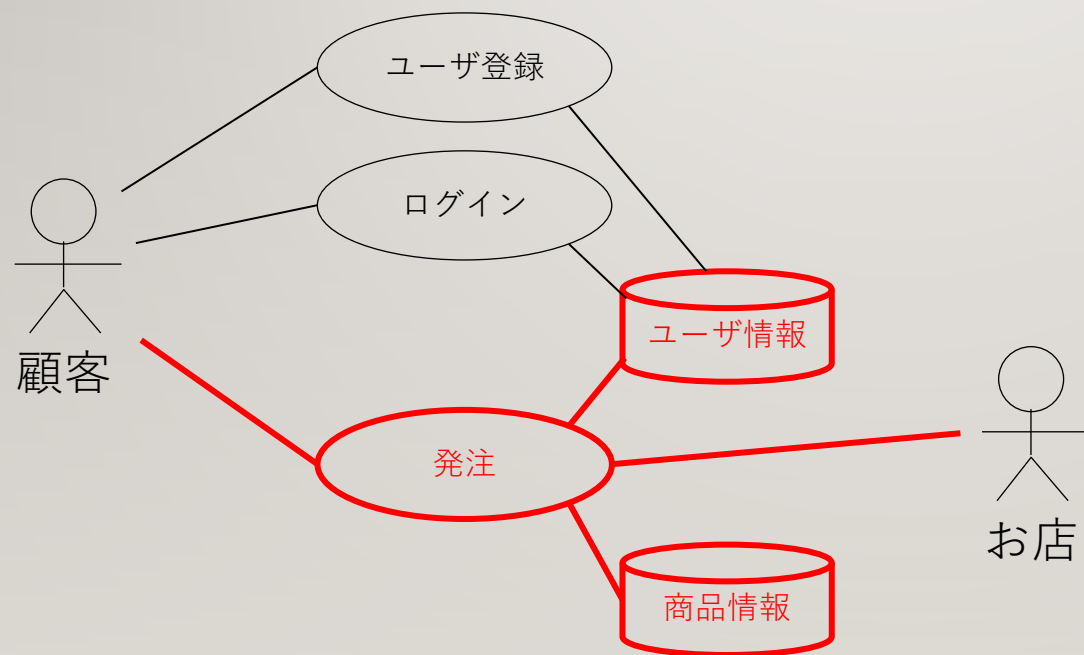
最近のWebでは、様々な技術を組み合わせて  
サービスが連携して動いている

- インターネットの仕組みを理解
- Webページを作ってみる。
- Webシステムで使われる様々なプログラミング言語を体験
- ブラウザで動くJavaScriptを体験
- サーバで動くプログラムPHPを体験
- インターネットのセキュリティについて考えてみる

# プログラミング応用の各週の予定



- 
1. **Webアプリケーションとプログラム言語** (11/07)
    - インターネットやWebの仕組みについて理解し、  
その中でJavaScriptやPHPなどのプログラム言語がどう使われるのか
  2. **JavaScriptによるフロントエンドプログラミング** (11/14)
    - Webブラウザ側で動くプログラム言語としてのJavaScriptについて、  
基本的な文法と使い方を理解を目指す
  3. **PHPによるバックエンドプログラミング** (11/21)
    - Webサーバで動くプログラム言語としての PHP について、  
基本的な文法と、ブラウザ側からのデータの受け渡しの仕組みを理解
  4. **Webプログラミングとセキュリティ** (11/28)
    - Web プログラミングと、サイバー攻撃について考える  
サイバー攻撃の仕組みの基本を理解し、その対処方法を学ぶ



- お買い物ページを作る
- 説明の的を絞るために
  - (1) ユーザ登録 - ×
  - (2) ログイン - ×
  - (3) 商品発注ページ - ○

## お買い物ページのユースケース図



# インターネットは どういう仕組みなのか？

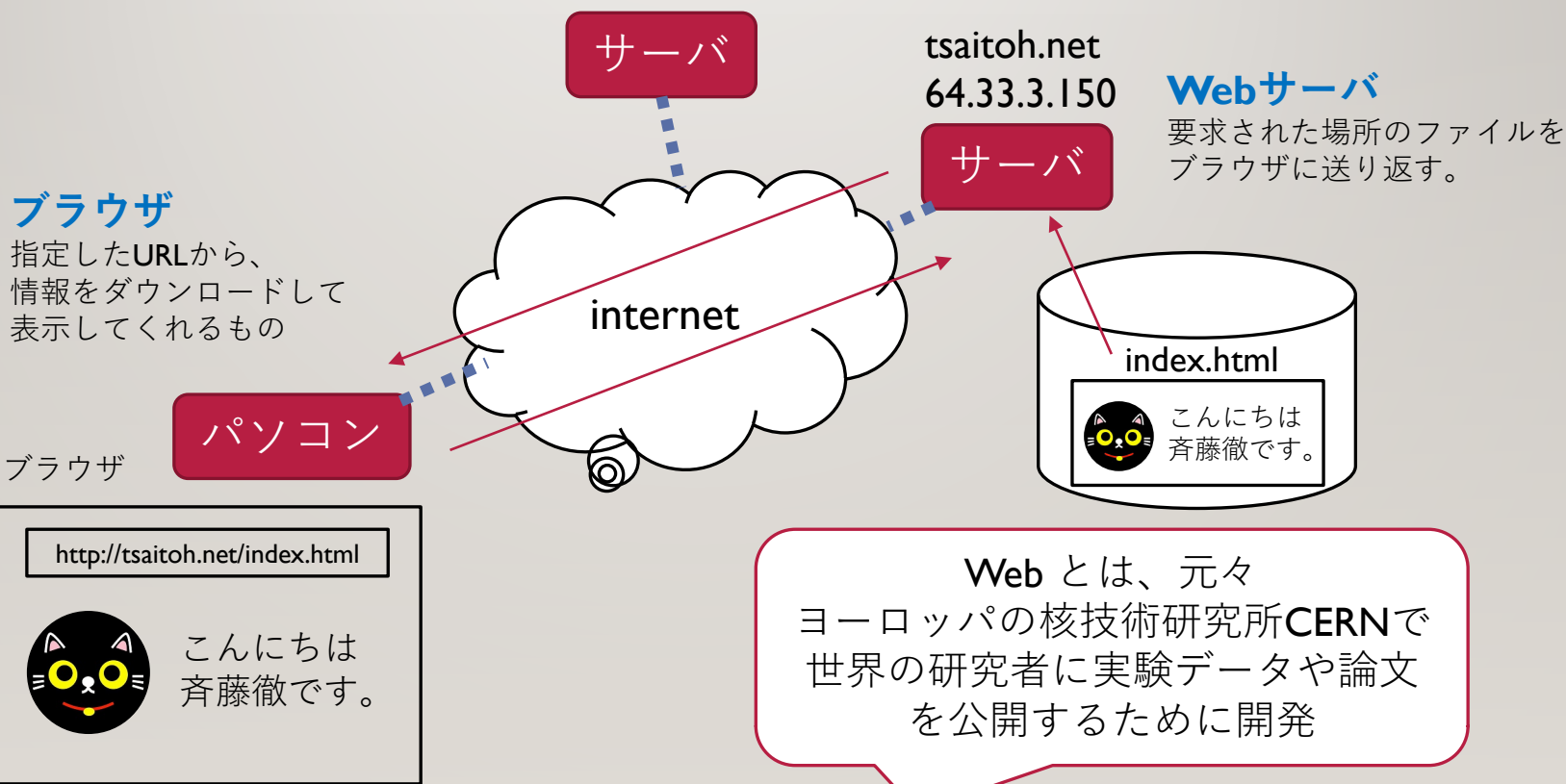
---

- インターネットの仕組み
- コンピュータ名と**DNS**
- ポート番号とプロトコル
- **URL**とは
- **HTML(Hyper Text Markup Language)**

# インターネットの仕組み

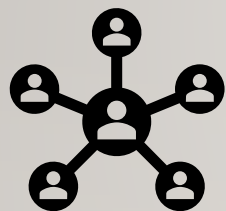


ティムバーナーズリー



# コンピュータ名とDNS

IPアドレス = 電話番号  
コンピュータ名=登録者名  
DNS = 電話帳



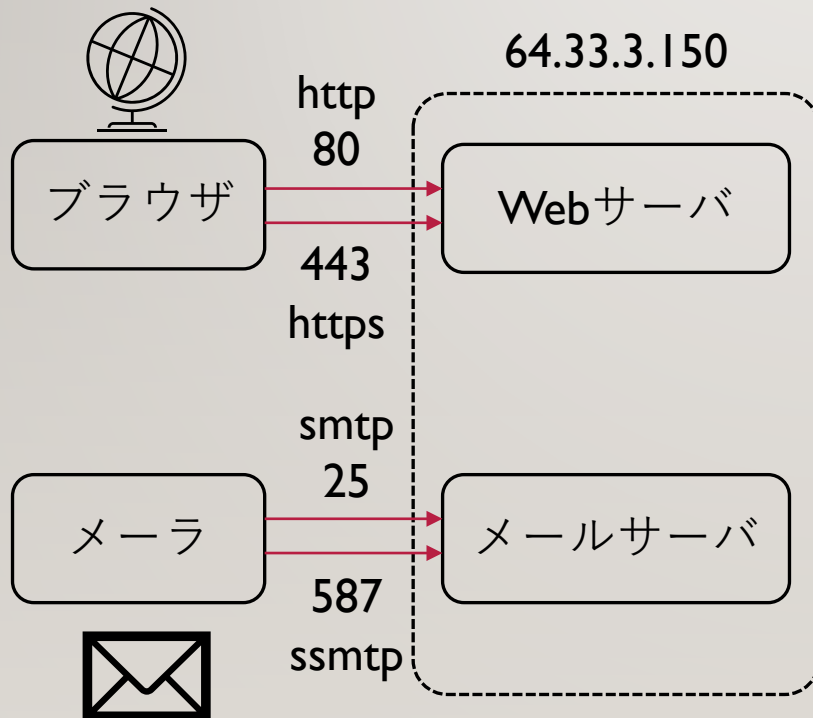
- IPアドレス - コンピュータ固有の番号  
64.33.3.150
- ドメイン名 - 覚えにくいから名前を付けよう  
tsaitoh.net
- **www . fukui-nct .ac .jp**
  - fukui-nct -- 組織ドメイン
  - .jp -- 国ドメイン
  - www -- コンピュータ名  
webサーバはwwwが多い
  - .ac -- 種別ドメイン
- DNS
  - IPアドレスとドメイン名を対応付けるサービス

最近は国ドメインや  
種別ドメインは  
流行らない...



# ポート番号とプロトコル

IPアドレス = 電話番号  
ポート番号 = 内線電話番号



- 1つのコンピュータの中で、通信する複数のプログラムを区別するためにポート番号を使う。
- 通信する時は、  
**IPアドレス:64.33.3.150**  
**ポート番号:80** ... でつながろう
- ポート番号覚えにくいよね  
名前を付けよう
- **80 : HTTP (Hyper Text Transfer Protocol)**
- **443: HTTPS (HTTPの暗号化版)**

# URLとは

<http://tsaitoh.net/~t-saitoh/index.php>

- **http** -- 通信プロトコルは**http**、ポート番号は**80**番を使おう
- **tsaitoh.net** – コンピュータ名、  
DNSに問い合わせて**64.33.3.150**につなごう

HTTP ステータスコード

200 : 見つかったよ!  
403 : 貴方には見せられないよ!  
404 : そんなページ無いよ!  
503 : 忙しくて返答できないよ!

ブラウザ

パソコン

見つかったよ! (200)  
←HTML形式のデータを返送

GET /~t-saitoh/index.php  
ファイルくれ!

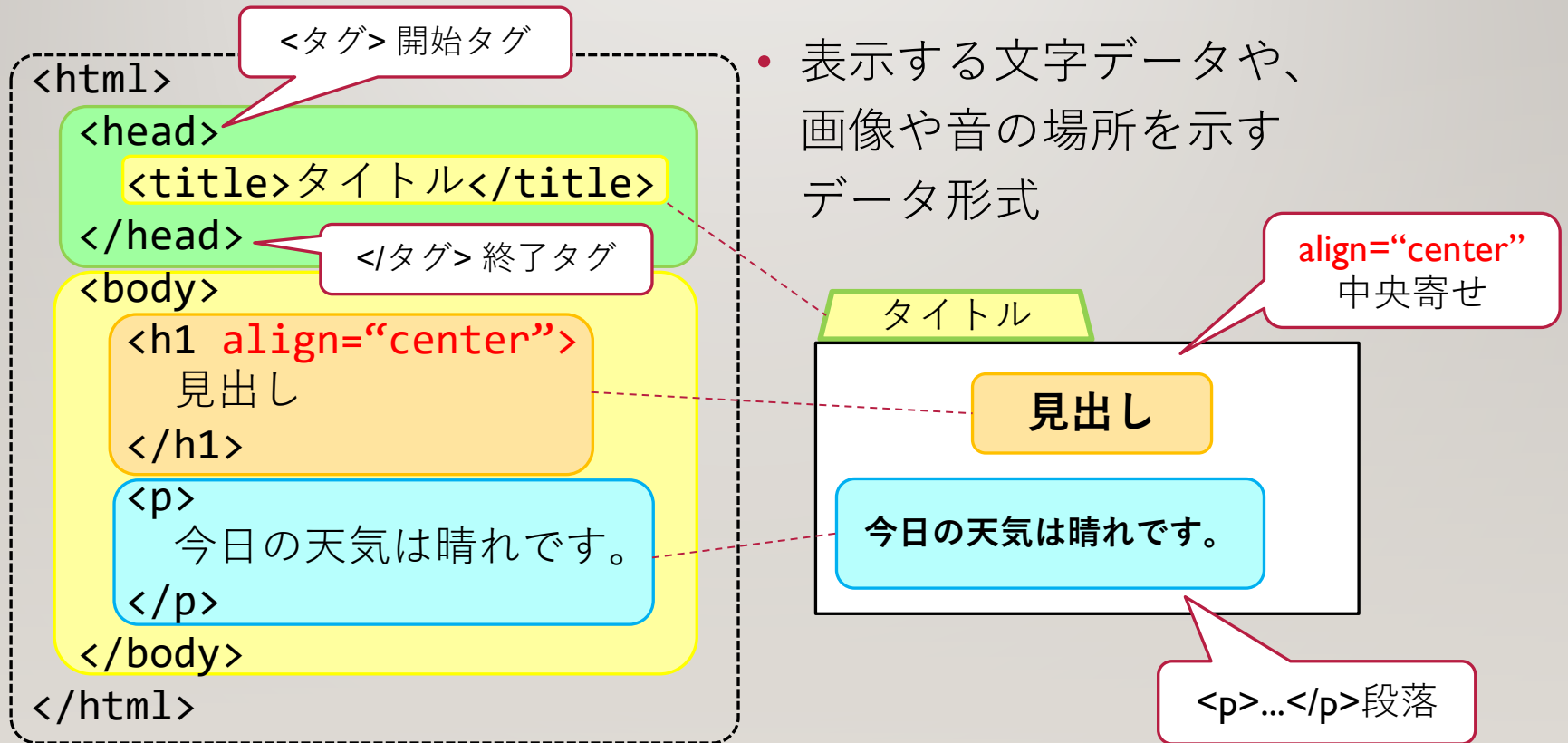
64.33.3.150

サーバ

/~t-saitoh  
index.php

```
<html>
<head>
  <title>タイトル</title>
</head>
<body>
  <h1 align="center">
    見出し
  </h1>
  <p>
    今日の天気は晴れです。
  </p>
</body>
</html>
```

# HTML(HYPER TEXT MARKUP LANGUAGE)



# 実際にサーバを動かして インターネットの仕組みを見てみよう

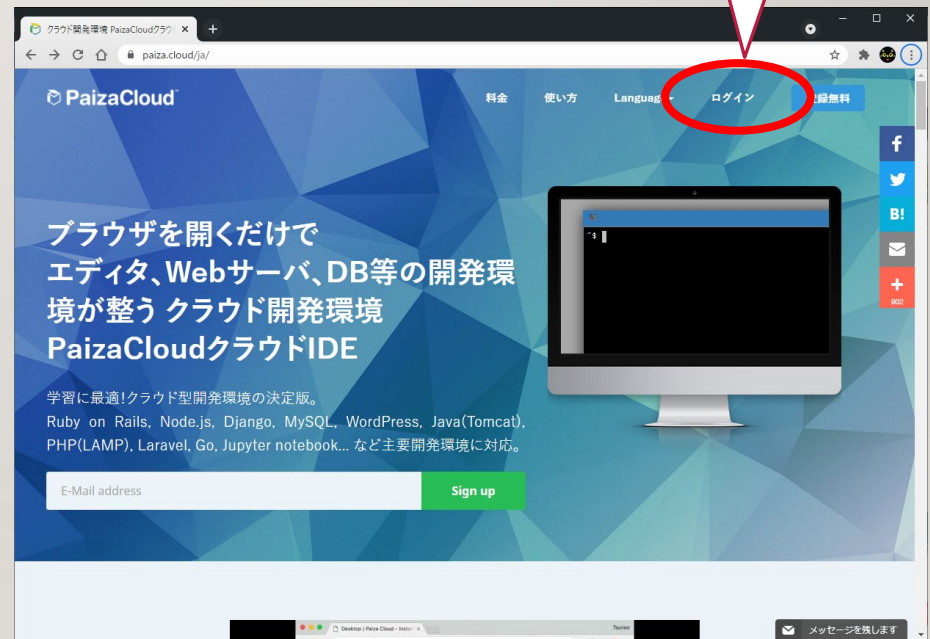
---

- **Paiza Cloud** の登録
- **Paiza Cloud** でサーバを作成
- **Paiza Cloud** の基本的な使い方
- **DNS**を確認してページを表示
- 教材データのダウンロード
- **Web**ページを触ってみよう

# PAIZA CLOUD の登録(I)

ログイン

- Webシステム構築の勉強用にサーバを借りれる
- googleのメールアドレスで登録
- <https://paiza.cloud/ja/>





# PAIZA CLOUD の登録(2)

手続きが面倒かな...

The screenshot shows the PaizaCloud signup page. The browser address bar shows 'paiza.cloud/signup'. The page has a green header with 'PaizaCloud' and navigation links for '新規サーバ作成' and 'マイサーバ'. A red circle highlights the 'アカウント作成' link in the header. The main form is titled 'アカウント作成' and contains fields for 'メールアドレス' (filled with 'tsaitoh.net@gmail.com'), 'ユーザ名' (filled with 't-saitoh'), and 'パスワード' (masked with dots). A blue arrow points from the 'メールアドレス' field to the text '手続きが面倒かな...'. Below the password field is a reCAPTCHA checkbox labeled '私はロボットではありません' with a green checkmark. A large blue button labeled 'アカウント作成' is below the form. Below this button is a line with 'または' and three social login buttons: 'GitHub ログイン', 'Facebook ログイン', and 'Google+ ログイン'. A red circle highlights the 'Google+ ログイン' button. A red arrow points from the 'アカウント作成' button to the 'Google+ ログイン' button. At the bottom, there is a link '既にアカウントをお持ちですか? ログイン' and a footer message 'メッセージを残します'.

アカウント作成

メールアドレス  
tsaitoh.net@gmail.com

ユーザ名  
t-saitoh ← アカウントの名前

パスワード

✓ 私はロボットではありません  
reCAPTCHA  
プライバシー・利用規約

アカウント作成

アカウントを作成すると、利用規約、およびCookieの使用を含むプライバシーポリシーに同意したことになります。

または

GitHub ログイン Facebook ログイン Google+ ログイン

既にアカウントをお持ちですか? ログイン

メッセージを残します

おすすめコース  
Google アカウントを  
使えば登録も簡単

# PAIZA CLOUD のサーバを作成



無料コースで使うので  
サーバは2時間使用すると  
使えなくなります

サーバ設定

サーバ名

tohrusaitoh

初期インストール&設定するものを選択してください  
※こちらにかかれていないものも手動設定は可能です。

- Web開発:  
Node.js PHP Tomcat(Java, JSP, Servlet) Ruby on Rails Ruby Sinatra Django Laravel  
Go(Revel)
- データベース: MySQL PostgreSQL MongoDB phpMyAdmin
- アプリケーション: WordPress Jupyter Notebook
- サーバ: Apache

☐ 常時起動(ベーシックプランのみ)

## サーバ設定

サーバ名

tohrusaitoh

サーバにつける名前  
空白ナシの分かり易い  
名前をつけてください。

初期インストール&設定するものを選択してください

※こちらにかかれていないものも手動設定は可能です。

• Web開発:

Node.js

PHP

Tomcat(Java, JSP, Servlet)

Ruby on Rails

Go(Revel)

PHP, Apache を  
選んで下さい。

• データベース:

MySQL

PostgreSQL

MongoDB

phpMyAdmin

• アプリケーション:

WordPress

Jupyter Notebook

• サーバ:

Apache

☐ 常時起動(ベーシックプランのみ)

☐ SSH利用

サーバタイプ: free, リージョン: tokyo

新規サーバ作成

キャンセル

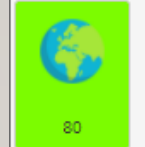
# Paiza Cloud の最初の画面



Webページや  
プログラムを編集



作ったページを  
ブラウザで表示



ターミナルで  
サーバを操作

tohrusaitoh | PaizaCloud - Instant x +

paiza.cloud/containers

PaizaCloud 新規サーバ作成

/home/ubuntu

public\_html

~/public\_html/index.php

保存 (S) ☐ 自動保存

```
1 <?php
2 // Open https://localhost/~ubuntu/index.php
3 echo "<h1>Hello " . "PHP</h1>";
4
```

https://localhost-tohrusaitoh.paiza-user-free.cloud/~ubuntu/index

## Hello PHP

browser

https://proxy-paiza-hatenablog-com.paiza.cloud/entry/2018/09/25/paizacloud\_php#

2018-09-25

pa

Pa

B!

メッセージを送ります

Apache2 Ubuntu Default Page: It works

<http://tsaitoh.net/~t-saitoh/index.php>

tsaitoh.net

T-Saitoh家族のホームページ

検索

ホーム » T-Saitoh

# 「T-Saitoh」 カテゴリーアーカイブ



# DNSを確認してページを表示

The image shows a PaizaCloud interface with a terminal window and a web browser. The terminal window displays the command `nslookup tsaitoh.net` and its output, which includes the IP address `64.33.3.150`. The web browser shows the URL `http://64.33.3.150/` and the website `tsaitoh.net`.

nslookup はDNSにコンピュータ名を問い合わせるコマンド

DNSで調べたIPアドレスでWebページを表示

`http://64.33.3.150/`

tsaitoh.net  
T-Saitoh家族のホームページ

# 教材データのダウンロード

ターミナル画面では、サーバで動かす命令を入力  
(先頭の\$は入力しなくていい)

Webページ用フォルダに移動して

**\$ cd public\_html**

この講義のサーバ用資料をまとめてダウンロード

**\$ git clone https://github.com/tohrusaitoh/recp.git**



```
~/public_html
~$ cd public_html
~/public_html$ git clone https://github.com/tohrusaitoh/recp.git
Cloning into 'recp'...
remote: Enumerating objects: 52, done.
remote: Counting objects: 100% (52/52), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 52 (delta 15), reused 46 (delta 9), pack-reused 0
Unpacking objects: 100% (52/52), done.
~/public_html$
```



メッセージを残します

# Webページを触ってみよう

ダブルクリックで  
フォルダ内を表示

ファイルを選んで  
右クリックで  
「開く」を選ぶ

ファイルを編集したら  
保存を押す。

右クリックで  
「ブラウザで開く」  
を選ぶ

```
1 <html>
2   <head>
3     <title>タイトル</title>
4   </head>
5   <body>
6     <h1 align="center">
7       見出し
8     </h1>
9     <p>
10      今日の天気は晴れです。
11    </p>
12  </body>
13 </html>
14
```

## 見出し

今日の天気は晴れです。

ブラウザで見たときの  
画面内容が確認できる





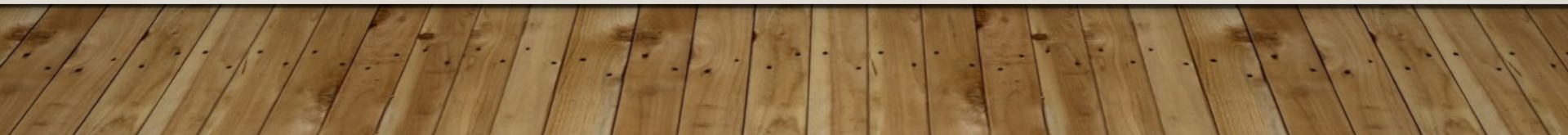
# INTERMISSION

引用：風と共に去りぬ

# 飾り気のないページを デコってみよう。

---

- お買い物ページの基本から
- 基本的なタグ
- スタイルシート





# お買い物ページの基本から

飾りっ気のないページ  
これではお客さんに興味を  
もってもらえない。

## Sample Page 1

### お客様情報

顧客ID	20010
顧客名	斉藤 徹
住所	東京都港区台場2-4-8

### 商品一覧

注文ID	商品名	単価	購入
1010	みかん	50円	<input type="button" value="購入"/>
1020	りんご	100円	<input type="button" value="購入"/>
1022	パイナップル	1000円	<input type="button" value="購入"/>

sample1.html

【sample1.html】

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="utf-8"/>
    <title>Sample Page 1</title>
  </head>
  <body>
    <h1>Sample Page 1</h1>
    <h2>お客様情報</h2>
    <table border="1">
      <tr><th>顧客ID</th><td>20010</td></tr>
      <tr><th>顧客名</th><td>斉藤 徹</td></tr>
      <tr><th>住所</th><td>東京都港区台場2-4-8</td></tr>
    </table>
    <h2>商品一覧</h2>
    <table border="1">
      <tr><th>注文ID</th><th>商品名</th>
        <th>単価</th><th>購入</th>
      </tr>
      <tr>
        <td>1010</td>
        <td>みかん</td>
        <td>50円</td>
        <td><input type="button" value="購入"></td>
      </tr>
```

# 表示内容をHTMLで飾るには

`<h1 align="center">タイトル</h1>`

align属性で表示位置をセンタリング

`<font size="4" color="blue">文字列</font>`

size="大きさ" 最大(default=3)

color="色", red, green, blue #RRGGBB (#FFFF00=yellow)

`<b>文字列</b>`      ボールド体(太字)

`<em>文字列</em>` 強調(斜体表示が多い)

ページ全体でのデザインの統一性を出すためには、HTMLが煩雑になる

# 統一感のあるデザインにしたいけど

```
<table border="1">
  <tr><th>注文ID</th><th>商品名</th><th>単価</th><th>購入</th></tr>
  <tr>
    <td><font color="grey">1010</font></td>
    <td><font size="4">みかん</font></td>
    <td><font color="red">50円</font></td>
    <td><input type="button" value="購入"></td>
  </tr>
  <tr>
    <td><font color="grey">1010</font></td>
    <td><font size="4">りんご</font></td>
    <td><font color="red">100円</font></td>
    <td><input type="button" value="購入"></td>
  </tr>
```

注文IDは灰色  
商品名は大きく  
単価は赤色

デザイナーさんが  
「デザイン変更！」  
単価は青太字にして！

# スタイルシートを使ってみよう

- 表示内容与设计は、別に記載しよう。

- **HTML**

<p>段落を14ポイントの赤太字で表示したい</p>

- **スタイルシート**

```
p.BODY {  
  color:red ;  
  font-size: 14pt ;  
  font-weight: bold ;  
}
```

p タグ(段落)の  
class="BODY"は  
ここに書いてある  
スタイルで表示

- **HTMLの先頭でスタイルシートを指定**

```
<!DOCTYPE html>  
<html lang="ja">  
  <head>  
    <meta charset="utf-8"/>  
    <title>Sample Page 2</title>  
    <link rel="stylesheet" type="text/css"  
          href="sample2.css" />  
  </head>  
  <body>  
    <p class="BODY">段落</p>  
    :
```

Sample Page 1

tsaitoh | PaizaCloud - Instant Linux

paiza.cloud/containers

# PaizaCloud

新規サーバ作成

新規ファイル

ターミナル

ブラウザ

80

+ Apps

/home/ubuntu

- public\_html
  - index.php
  - recp
    - jquery
    - Makefile
    - README.md
    - sample0.html
    - sample1.html
    - sample2.css
    - sample2.html

修正したら**保存**

スタイルシートファイル sample2.cssを選んで 右ボタン-**開く**

保存 (S) ☐ 自動保存

```
~$ cd public_html
~/public_html$ git ci
Cloning into 'recp'...
remote: Enumerating objects 1
remote: Counting objects 1
remote: Compressing objects 1
remote: Total 82 (delta 0)
Unpacking objects: 100%
~/public_html$
```

body {  
 background-color: white ;  
}  
h1 {  
 text-align: center ;  
 color: blue ;  
 font-weight: bold ;  
}  
h2 {  
 font-size: 1.2em;  
 border-left-width: 0.5em;  
 border-left-style: solid;  
 border-bottom-width: 2px;  
 border-bottom-style: solid;  
 padding-top: 2px;

練習問題  
見出し<h1>を赤く  
名前を青文字  
にしてみよう！

修正したら **リロード**

保存 (S) ☐ 自動保存

```
1 <!DOCTYPE html>
2 <html lang="ja">
3   <head>
4     <meta charset="utf-8"/>
5     <title>Sample Page 2</title>
6     <link rel="stylesheet" type="text/css"
7   </head>
8   <body>
9     <h1>Sample Page 2</h1>
10    <h2>お客様情報</h2>
11    <table border="1">
12      <tr>
13        <th>顧客ID</th><td class="ID">20010</td>
14      </tr>
15      <tr>
16        <th>顧客名</th><td>斉藤 徹</td>
17      </tr>
18      <tr>
19        <th>住所</th><td>東京都港区台場2-4-8</td>
20      </tr>
```

HTMLファイル sample2.htmlを選んで 右ボタン-**開く**

browser

https://paizacloud-agent-tsaitoh.paiza-user-agent.cloud/filesystem/home/ubuntu/p

## Sample Page 2

### お客様情報

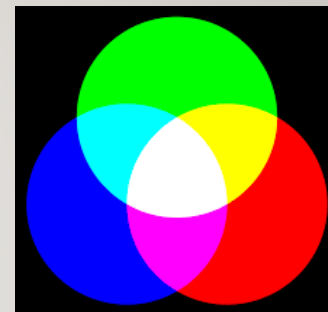
顧客ID	20010
顧客名	斉藤 徹
住所	東京都港区台場2-4-8

HTMLファイル sample2.htmlを選んで 右ボタン-**ブラウザで開く**

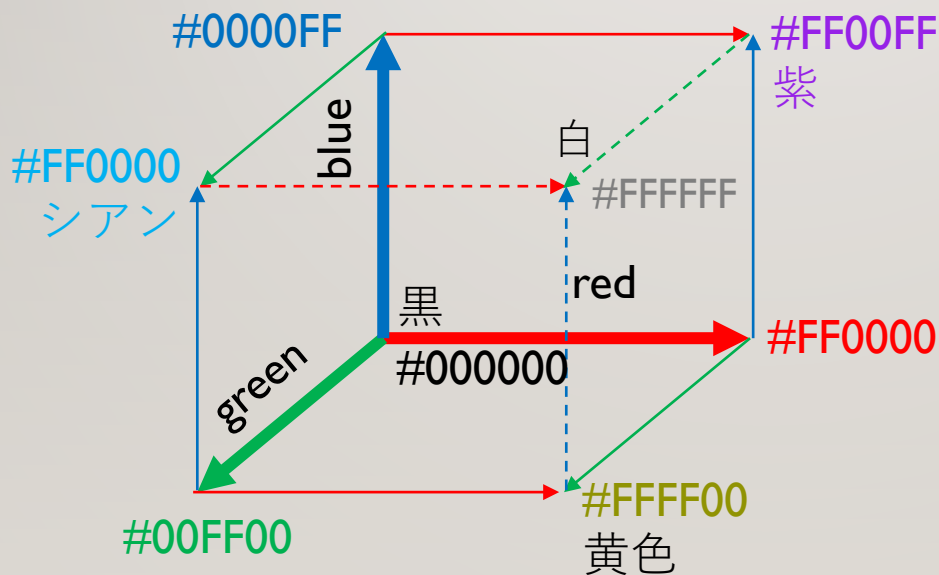
メッセージを残します



# スタイルシートの色指定



- `<font color="#80FF80">文字の色は?</font>`



- 色の各桁は0-9,A-Fの16進数
- 16進数2桁0~255までのRED, GREEN, BLUEの明るさ
- FFが一番明るい
- 80が中ぐらい
- 00が一番暗い

# WEBページのデータをどこで作るか?

---

```
<table border="1">
```

```
<tr><th>注文ID</th><th>商品名</th><th>単価</th><th>購入</th></tr>
```

```
<tr>
```

```
<td class="ID">1010</td>
```

```
<td class="NAME">みかん</td>
```

```
<td class="PRICE">50円</td>
```

```
<td><input type="button" value="購入"></td>
```

```
</tr>
```

【これ以降、各商品のデータがたくさん並ぶ...】

【スタイルシート】

```
td.ID {  
    color: gray ;  
}  
td.NAME {  
    color: blue ;  
    font-size: 1.2em;  
}  
td.PRICE {  
    color: red ;  
    font-weight: bold ;  
}
```

# 一つの商品に対するHTMLを 人が編集するのは馬鹿げてる

---

## 商品情報

注文ID	商品名	単価
I010	みかん	50
I020	りんご	100
I022	パイナップル	1000

プログラムで  
ひな型に合わせて  
自動生成すればいい

## HTMLファイル

```
<tr>
  <td class="ID">1010</td>
  <td class="NAME">みかん</td>
  <td class="PRICE">50円</td>
  <td>... </td>
</tr>
<tr>
  <td class="ID">1020</td>
  <td class="NAME">りんご</td>
  <td class="PRICE">100円</td>
  <td>... </td>
</tr>
```

# サーバで**CGI**によってページを生成

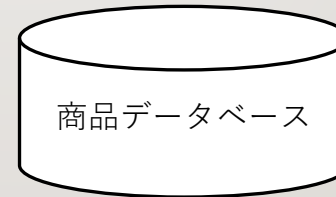
## Common Gateway Interface

- **一昔前**のやり方...

注文ID	商品名	単価
1010	みかん	50
1020	りんご	100
1022	パイナップル	1000

### 商品一覧 HTMLファイル

注文ID	商品名	単価	購入
1010	みかん	50	購入
1020	りんご	100	購入
1022	パイナップル	1000	購入



データベースの情報を使って  
サーバで**HTML**を生成

```
<tr>
  <td class="ID">1010</td>
  <td class="NAME">みかん</td>
  <td class="PRICE">50円</td>
  <td>... </td>
</tr>
<tr>
  <td class="ID">1020</td>
  <td class="NAME">りんご</td>
  <td class="PRICE">100円</td>
  <td>... </td>
</tr>
```

# サーバで**PHP**によってページを生成

- 一般的なやり方

商品一覧 HTMLファイル

注文ID	商品名	単価	購入
1010	みかん	50	購入
1020	りんご	100	購入
1022	パイナップル	1000	購入

ブラウザ

商品データベース

Webサーバ  
PHP

**\$itemlist**

id	name	price
1010	みかん	50
1020	りんご	100
1022	パイナップル	1000

```
<!php foreach( $itemlist as $item ) : ?>
<tr>
  <td class="ID">
    <?php echo $item.id ; ?></td>
  <td class="NAME">
    <?php echo $item.name ; ?></td>
  <td class="PRICE">
    <?php echo $item.price ;?></td>
  <td>... </td>
</tr>
<!php endforeach ; ?>
```

HTMLとPHPプログラムが混在



# JavaScriptによってページを生成 (最近のやり方)

## 商品一覧 HTMLファイル

注文ID	商品名	単価	個数
1010	みかん	50	5 個
1020	りんご	100	2 個
1022	パイナップル	1000	1 個
合計 1450 円			

1. JSON形式をもらってHTMLを生成
2. 買い物結果を送信

フロントエンド

HTMLとJavaScript  
プログラムを送信

商品データベース

データ送信  
JSON形式

データ受信  
JSON形式

id	name	price
1010	みかん	50
1020	りんご	100
1022	パイナップル	1000

## 購入結果

id	name
1010	5
1020	2
1022	1

バックエンド

## 【JSON形式】

JavaScript で扱いやすい  
データ形式

# WEBシステムの解りにくい点

## 様々なデータ形式・プログラム言語を使う

---

- データ形式
  - HTML
  - スタイルシート
  - JSON データ
    - JavaScript向けデータ形式
- JavaScript(ライブラリ)
  - Ajax
  - jQuery
- プログラム言語
  - JavaScript – フロントエンド向け
  - PHP - バックエンド向け
  - SQL - データベース操作
  - 他にも
    - Python - 最近流行り
  - スマートフォン
    - Java - Android端末
    - Swift - iOS 端末
    - Kotlin

# プログラム言語が色々あります

- フロントエンドとバックエンドで  
慣れたプログラム言語で書きたい...

フロントエンド	バックエンド	データベース
JavaScript	PHP	SQL
JavaScript	<b>Node.js</b>	色々
<b>Java</b>	JavaServlet	色々
色々	Django( <b>Python</b> )	
	COBOL ??	SQL ??

# 今日のまとめ

---

- Webシステムの仕組み
- Webページのデータの構造
  - HTML と スタイルシート
- Webサーバを使ってみよう
  - Paiza Cloud の使い方
- フロントエンドとバックエンドとは
  - ブラウザで動くJavaScript      フロントエンド
  - サーバで動くPHP                  バックエンド