

Javascript 01



G's ACADEMY
FUKUOKA



本日の内容

講義 + 作業 : 2.5 h程度
演習 : 1.5 h程度

アジェンダ

- javascriptの役割, できること, 概要
- 変数
- 条件分岐(if文)
- ランダム関数(Math)
- jQuery
- おみくじアプリ演習
- 課題発表→チュータリング(演習)タイム
- 写経のススメ(?)

授業のルール

- 授業中は常にエディタを起動！
- 隣の人と相談するときは周りの迷惑にならない大きさで.
- 周りで困ってそうな人がいたらおしえてあげましょう！
- まずは**打ち間違い**を疑おう！

{ } ' " ; など

- 書いたら保存しよう！

command + s

ctrl + s

javascript概要

javascript概要

■html(マスター済み)

- ・ コンテンツの指定
- ・ タイトル, 文章, 画像などの記述

■css(マスター済み)

- ・ コンテンツの装飾
- ・ 色, 大きさ, 配置などの指定

■javascript

- ・ ユーザー操作, イベント発生による動きを実現

javascript基礎

javascriptを記述！

■書き方

- ・ `<script></script>`の間に処理を記述

■どこに書くの??

- ・ htmlファイルの`</body>`のすぐ上に書こう！
- ・ ほかにいくつか書ける場所があります.
- ・ 別にファイルを作るやり方もあります.

サンプルファイルで確認

js_practice.html

```
<!doctype html>  
<html lang="ja">  
<head>.....</head>
```

```
<body>  
  <h1 id="echo">js練習</h1>  
  <script>  
    //ここに記述！  
  </script>  
</body>  
  
</html>
```

<script>は</body>のすぐ上！
ほかのhtmlより下に書こう！

まずは書いて動かそう！

js_practice.html

■alert();

```
<script>
```

```
    alert("Hello world");  //文字列を「"」で括る. 「'」でもOK
```

```
</script>
```

■console.log(); 検証ツール→consoleで確認！

```
<script>
```

```
    console.log("Hello world");
```

```
</script>
```

※ //でコメントアウトできます(実行しないようにします)

変数

変数とは??

js_practice.html

■変数

- ・ 文字列や数値を入れる箱のことです！

■例

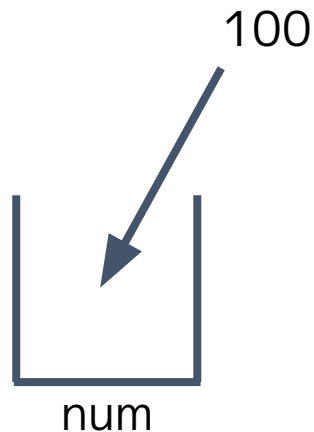
```
<script>
```

```
  let name1 = "hello";
```

```
  let name2 = "world";
```

```
  let num = 100;
```

```
</script>
```



「=」は「等しい」ではない
「numに100を代入」の意味

変数とは??

js_practice.html

■const

- ・ 上書きできない変数

```
const a = 10;
```

```
a = 20; //エラーになります！
```

■let

- ・ 上書きできる変数(上の例で実行可能)

■var

- ・ やや古い書き方(はじめは全部varでもOK！)

変数とは??

js_practice.html

■NGパターン

- ・ 「"」 「'」 の混在
- ・ 変数名の先頭が数字

■まずいですよ先輩！

```
<script>
```

```
    let name = "hello";
```

```
    let name2 = world;
```

```
    let 1num = 100;
```

```
</script>
```

配列

■配列

- ・ 複数の値に順番をつけてまとめて扱う方法.
- ・ 順番を「インデックス」と呼ぶ。 **「0」からスタート!**

■例

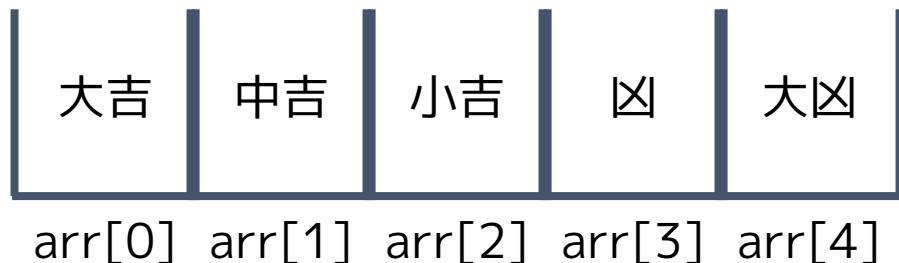
```
<script>
```

```
const arr = ["大吉", "中吉", "小吉", "凶", "大凶"];
```

```
</script>
```

■インデックス

```
alert(arr[0]);    //大吉
```



演算子と計算

■演算子

- ・「+」とか「-」のこと.

<script>

```
let sum1 = 1 + 9;    //10
```

```
let sum2 = 1 - 5;    //-4
```

```
let sum3 = 2 * 4;    //8
```

```
let sum4 = 10 / 2;   //5
```

```
let sum5 = 10 % 3;   //1
```

</script>

■数値や文字列の計算

```
<script>
```

```
    let name1 = "hello";
```

```
    let name2 = "world";
```

```
    let num1 = 100;
```

```
    let num2 = 200;
```

```
    let sum1 = name1 + name2;    //helloworld
```

```
    let sum2 = num1 + num2;      //300
```

```
</script>
```

使用できない単語

■変数名には使用できない単語が存在する

- ・ 構文で使用するもの, 将来的に使われるもの, など

■予約語とキーワード

- ・ 「MDN javascript 予約語」で検索！！

条件分岐

■条件分岐処理

```
<script>
```

```
    if(条件式){
```

```
        条件式を満たす場合の処理
```

```
    }
```

```
    else{
```

```
        条件式を満たさない場合の処理
```

```
    }
```

```
</script>
```

■条件分岐処理(複数条件)

```
<script>
```

```
    if(条件式1){
```

```
        条件式1を満たす場合の処理
```

```
    }
```

```
    else if(条件式2){
```

```
        条件式1と満たさなくて条件式2を満たす場合の処理
```

```
    }
```

```
</script>
```

■条件式の書き方(比較演算子)

```
<script>
```

```
    const val = 10;
```

```
    if(val == 10){
```

```
        alert(値は10です ! );
```

```
    }
```

```
    else{
```

```
    }
```

```
</script>
```


■比較演算子の種類

| | |
|----|-----------------|
| == | 等しければtrue |
| != | 等しくなければtrue |
| > | 左側のほうが大きければtrue |
| < | 右側のほうが大きければtrue |
| >= | 左側が右側以上ならtrue |
| <= | 右側が左側以上ならtrue |

■条件分岐・問題

```
const a = 1;
```

```
const b = "福岡";
```

```
if(条件式1){      //aが1以上の場合
```

```
    alert("1以上です！");
```

```
}
```

```
if(条件式2){      //bに"福岡"が代入されていない場合
```

```
    alert("福岡ではない！");
```

```
}
```

■条件の組み合わせ

```
const val = 10;
```

```
if(val >= 5 && val <= 15){ //valが5-15の数値であればtrue
```

```
    ...
```

```
}
```

```
if(val == 5 || val = 10){ //valが5か10であればtrue
```

```
    ...
```

```
}
```

Mathオブジェクト

■Mathオブジェクトとは??

数値関連の関数がたくさん含まれる。よく使用するのは乱数生成。

■Math.random

```
const num = Math.random();  
alert(num);    //0から1の間でランダムな値が表示される。
```

■Math.floor

端数を切り捨てする関数

```
const num = Math.floor(Math.random * 5)  
alert(num);    //0から4までのどれかが表示される！
```

■範囲を決めてみよう

```
const num1 = Math.floor(Math.random * ○○);  
alert(num1);           //0から○○までのどれかが表示される！  
  
const num2 = Math.floor(Math.random * ○○ + 1);  
alert(num2);           //1から○○までのどれかが表示される！
```

■ポイント

- ・ サンプルで動かしたらそのまま使おう！！
- ・ 最小値と最大値を決めて表示する処理も作ってみよう！

■練習

以下の数をランダムで発生させてalert();で表示させよう！

①0から9

②1から9

③5から10

④50から99

演習①

演習：おみくじアプリ作成①

omikuji01.html

■仕様

- ・ ランダムに「大吉・中吉・小吉・凶・大凶」をalert()で表示！

■ヒント

- ・ Math関数で0から4を発生させ，出た数値に応じてalert();で出力！

DOM

HTML = document

htmlに記述されている各要素のこと(document object modelでググろう)

ログインしていません トーク 投稿記録 アカウント作成 ログイン

ページ

ノート

閲覧

編集

履歴表示

Wikipedia内を検索



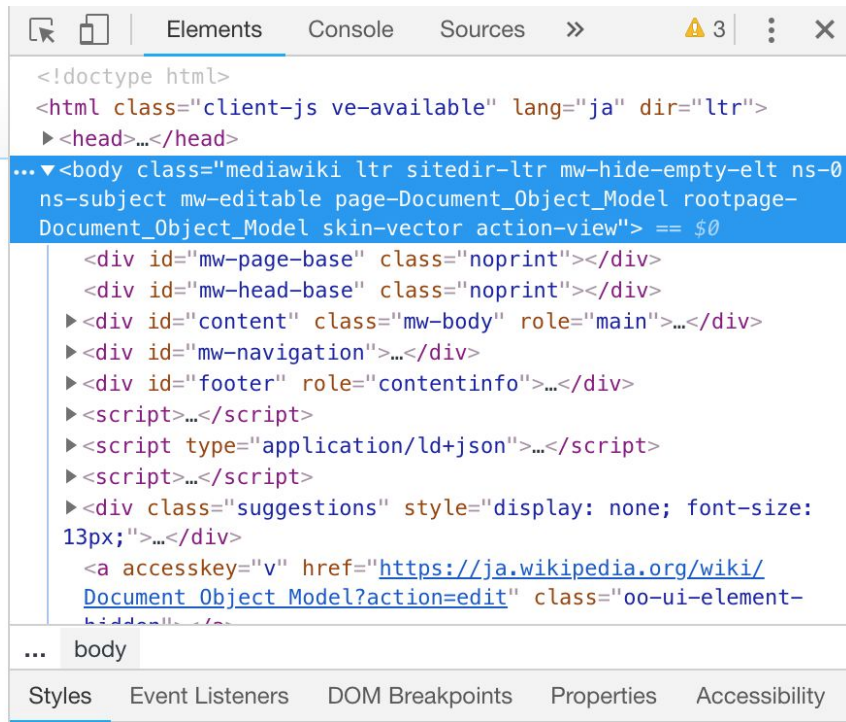
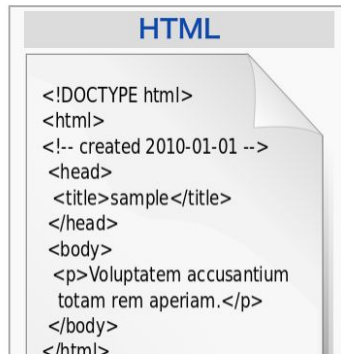
Document Object Model

出典: フリー百科事典『ウィキペディア (Wikipedia)』

Document Object Model (DOM) は、**HTML** 文書や**XML**文書を各種プログラムから利用するための仕組みである^[1]。**WHATWG**がLiving Standardとして定義している。

WHATWG以前は**W3C**が仕様を策定しており、Level 1からLevel 4まで勧告している。

XMLを読み込むAPIである**SAX**と異なり、XML



selector

各要素を取得

■考え方

どこに対して，何をするか，どのタイミングで

■idの指定

「どこに対して」ではidを用いることが多い！

```

```

```
→ const button = document.getElementById('btn');
```

```
→ const button = document.querySelector('#btn');
```

```
console.log(button.id);
```

各要素を取得

■ いろいろな要素を取得

```
<div id="main" class="class01" ></div>
```

- ・ div直下のpを取得

```
document.querySelector('div>p');
```

- ・ input要素のbutton要素を取得

```
document.querySelector('input[type=button]');
```

- ・ classがclass01の最初の要素を取得

```
document.querySelector('.class01');
```

event

どのタイミングで実行されるかを指定

■考え方

どこに対して、何をするか、**どのタイミングで**

■イベント

ページの読み込み、クリックなど何かしらの動作が起こったとき

■イベントハンドラ

onload ページや画像の読み込み時に発生

onclick 指定した要素がクリックされたときに発生

onchange フォームなどの入力内容が変更されたときに発生

どのタイミングで実行されるかを指定

■ クリックイベントの記述例

```
document.querySelector('btn').onclick = function(e){  
    alert('hello world');  
};
```

※ 上記 `function(e){}` の `e` は省略可能.

`e` については「javascript e」などでググろう！
(今はわからなくてもOK)

イベントハンドラー一覧

■たくさんあります！

クリック，内容変更，マウスが乗ったとき，動かしたときなど．．．

<http://phpjavascriptroom.com/?t=js&p=event>

書くことが多い！！



jQueryライブラリ

jQueryとは

■特徴

- ・cssと同じ要領で対象箇所を指定できる
- ・素のjsよりも短く書ける ← **重要**
- ・アニメーションなど手軽に設定できる。ライブラリも豊富！
- ・書き方(順序や考え方)はjsと同様！
- ・1行追加するだけで導入が簡単(フレームワークなどは環境構築で詰む)

⇒ **jQueryはjsを短縮して書けるライブラリです**

【参考】 <https://furien.jp/columns/140/>

jQueryはオワコン??



web サイト 用途なら今でも全然使える

- ・ web サイト内の要素をグリグリ動かすなど、用法用量さえ間違えなければこれほど便利なライブラリは他にない（現在進行形）
- ・ オワコンと言っているのは、より複雑な web アプリを作っているごく一部の『酔狂な人々』であり、彼らの多くは件の web サイト制作のことは一切視野に入れていない

【参考】

<https://speakerdeck.com/wakamsha/rmp-enziniabutokiyantu-2017-youkoso-web-hurontoendofalseshi-jie-he?slide=53>

まずはたくさん書くほうが大事！

jQueryの準備

jQuery最初の準備(超重要)

■基本

まずはjQueryを読み込もう。思考停止で以下を追加！

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
```

```
<script>
```

```
    $(function(){
```

```
        //ここに自分の処理を追加！
```

```
    });
```

```
</script>
```

※記述は</body>の上！

※忘れたら「jquery google」でググる

jQueryの基本

jQueryの文法(覚えよう！)

■特徴

セレクト → イベント → メソッド

どこを → どのタイミングで → どうする

■例(基本の形はだいたいこれです！)

idが「email」の要素をクリックしたら記述内容を「gs@gs.com」に変更

```
$('#email').on('click', function(){  
    $('#email').text('gs@gs.com');  
});
```

jQueryの文法(覚えよう！)

セレクト	→ イベント	→	メソッド
どこを	→ どのタイミングで	→	どうする

【参考】 <http://www.hp-stylelink.com/news/2013/11/20131122.php>

セレクト

■cssで設定するときと同様

#id → **\$('#id')**

.class → **\$('.class')**

Element → **\$('Element')**

[name=a] → **\$('[name=a]')**

※「#id」などは「"」か「'」で囲む！

※どちらを使っても良いが統一しよう！！

セレクト

■参考(まだまだあります. . . !)

複数指定 → `$('div, p, a')`

親子関係 → `$('div > p')`

先祖子孫 → `$('div p')`

最初の要素 → `$('li:first')`

奇数の要素 → `$('li:odd')`

偶数の要素 → `$('li:even')`

jQueryの文法(覚えよう！)

セレクト	→ イベント	→	メソッド
どこを	→ どのタイミングで	→	どうする

イベント

■例

クリックしたときにfunction(){}の中を実行する！

```
$('#id').on('click', function(){
```

```
    $('#セレクタ').css('color', 'red');
```

```
    $('#セレクタ').append('<p>12345</p>');
```

```
});
```

※「click」部分をほかのものに変更することでイベントを指定できます！

'click', 'dbclick', 'mouseout', 'mousedown', 'mouseup', 'change', 'focus',
'submit', 'resize', 'scroll', 'keydown', 'keyup', ...

jQueryの文法(覚えよう！)

セレクト	→ イベント	→	メソッド
どこを	→ どのタイミングで	→	どうする

メソッド(=命令)

■例

```
var elem = '<a href="#">次ページ</a>';  
$('#id').html(elem);           //htmlを表示  
$('#id').text(elem);           //文字列として表示  
$('#id').css('color', 'red');   //cssを適用  
$('#id').show(4000);           //非表示→表示(逆はhide)  
$('#id').prepend(elem);        //先頭にhtmlを追加  
$('#id').append(elem);         //末尾にhtmlを追加  
$('#id').empty();              //子要素を削除
```

もう一度まとめ

jQueryの文法(覚えよう！)

■特徴

セレクト → イベント → メソッド

どこを → どのタイミングで → どうする

■例(基本の形はだいたいこれです！)

idが「email」の要素をクリックしたら記述内容を「gs@gs.com」に変更

```
$('#email').on('click', function(){  
    $('#email').text('gs@gs.com');  
});
```

まずは形を入力することに慣れよう！

```
$('#id名').on('click', function(){  
    ...  
});
```

(口に出しながら書くと定着する)

だらーあいでいーおんくりつくふあんく
しょんかっこかっこ...

とにかく試そう！！

■以下の処理を書いてみよう！

- ①「テスト01」をクリックしたら赤くしよう！
- ②「テスト02」をクリックしたら「test02」に書き換えよう！
- ③「テスト03」をクリックしたら下に「テスト04」を追加しよう！
追加内容は「<p id="test04">テスト04</p>」にしよう.

演習②

■仕様

- ・ おみくじボタンをクリックしたら以下のどれかを表示！
- ・ 「大吉・中吉・小吉・凶・大凶」

■記述例

- ・ ボタン押したときの動作

```
$('#btn').on('click', function(){.....});
```

- ・ 表示

```
$('#id名').html('大吉');
```

課題

【課題】じゃんけんアプリ作成

janken.html

■じゃんけんアプリの仕様

- ①「グー」「チョキ」「パー」のボタンを設置
- ②どれかをクリックしたら「コンピュータの出した手は？」を変更
「コンピュータ：グー」「コンピュータ：チョキ」など
- ③「結果は？」の箇所に
「あなたの負け」「あなたの勝ち」「あいこ」のどれかを表示！

※上記を最低ラインとして製作

※これを土台にしてガンガン発展させよう！！

提出は毎週木曜日「23:59:59」まで！！
詳細は別資料参照！

課題ができねえええ
(` ; ω ; ´)

詰んだ．．． どうしようもない．．． という方は

写★経

※写経とは

誰かが書いた動作するコードをひたすら書き写すこと

```
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <meta http-equiv="X-UA-Compatible" content="ie=edge">
8     <link rel="stylesheet" href="css/sample.css">
9     <title>じゃんけん</title>
10 </head>
11
12 <body>
13     <header>
14         <h1>じゃんけん</h1>
15     </header>
16
```



```
12 <body>
13   .... <header>
14     .... <h1>じゃんけん</h1>
15   .... </header>
16
17   .... <main>
18     .... <ul>
19       .... <li id="gu_btn">グー</li>
20       .... <li id="cho_btn">チョキ</li>
21       .... <li id="par_btn">パー</li>
22     .... </ul>
23     .... <div id="com_hand">コンピュータの出した手は？</div>
24     .... <div id="judgment">結果は？</div>
25   .... </main>
26
27   .... <footer>フッター</footer>
28
```

```
29     <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
30     <script>
31         $(function(){
32             const result = ['グー', 'チョキ', 'パー'];
33
34             $('#gu_btn').on('click', function(){
35                 // 自分の手はグー
36                 let rand = Math.floor(Math.random() * 3);
37                 // 0コンピュータグー, 1コンピュータチョキ, 2コンピュータパー
38                 // コンピュータの手を表示
39                 $('#com_hand').text('コンピュータの手: ' + result[rand]);
40                 if (rand == 0){
41                     $('#judgment').text('あいこ');
42                 } else if (rand == 1){
43                     $('#judgment').text('勝ち');
44                 } else {
45                     $('#judgment').text('負け');
46                 }
47             });
48
```

```
49 ..... $('#cho_btn').on('click', function () {  
50 ..... // 自分の手はチョキ  
51 ..... let rand = Math.floor(Math.random() * 3);  
52 ..... // 0コンピュータグー, 1コンピュータチョキ, 2コンピュータパー  
53 ..... // コンピュータの手を表示  
54 ..... $('#com_hand').text('コンピュータの手: ' + result[rand]);  
55 ..... if (rand == 0) {  
56 .....     $('#judgment').text('負け');  
57 ..... } else if (rand == 1) {  
58 .....     $('#judgment').text('あいこ');  
59 ..... } else {  
60 .....     $('#judgment').text('勝ち');  
61 ..... }  
62 ..... });  
63
```

```
64     $('#par_btn').on('click', function() {
65         // 自分の手はチョキ
66         let rand = Math.floor(Math.random() * 3);
67         // 0コンピュータグー, 1コンピュータチョキ, 2コンピュータパー
68         // コンピュータの手を表示
69         $('#com_hand').text('コンピュータの手: ' + result[rand]);
70         if (rand == 0) {
71             $('#judgment').text('勝ち');
72         } else if (rand == 1) {
73             $('#judgment').text('負け');
74         } else {
75             $('#judgment').text('あいこ');
76         }
77     });
78
79 });
80 </script>
81
82 </body>
83
84 </html>
```

**「写経」これでいける！！
提出は毎週木曜日「23:59:59」まで！！**

チュータリングタイム

17:00までは一人でもくもく
後半は近くのメンバーで教え合おう！