

配列

- Array -

JavaScriptの配列

【 配列 (array) 】

複数の値に順番をつけてまとめて扱う方法です。
配列の順番を識別する番号を「**インデックス**」と呼びます。
インデックスは「**0**」から始まります。

```
<script>  
  var list1 = ['大吉','中吉','小吉','吉','凶'];  
  var list2 = new Array('大吉','中吉','小吉','吉','凶');  
</script>
```

【 配列のアクセスイメージ (array) 】

インデックス「0」から値が**格納**されます。
list[1]では「css」が取得可能



連想配列

- Object/オブジェクト -

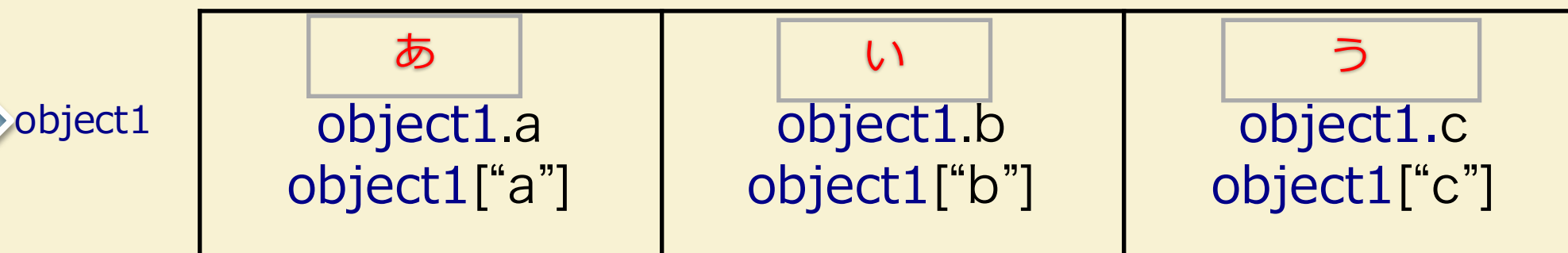
【オブジェクト/連想配列 (Object)】

配列と違い「インデックス」ではなく「**プロパティ**」で値を管理することができます。プロパティは「名前:値」のペアになっており最近ではよく使用されるデータ保持の方法の1つです。

```
<script>
var object1 = { 'a':'あ', 'b':'い', 'c':'う' };
var object2 = {
  'obj1' : { 'c1':'A', 'c2':'B', 'c3': 'C' },
  'obj2' : { 'c1':'A', 'c2':'B', 'c3': 'C' }
};
</script>
```

【オブジェクトの参照イメージ】

object1オブジェクト内のプロパティに値が**格納**されます。object1の中の値を取得する方法は、「object1 **プロパティ名**」 「object1 **["プロパティ名"]**」で取得可能



object1	<div>あ</div> <div>object1.a</div> <div>object1["a"]</div>	<div>い</div> <div>object1.b</div> <div>object1["b"]</div>	<div>う</div> <div>object1.c</div> <div>object1["c"]</div>
---------	---	---	---

反復処理

【 反復処理 (Iterate) 】

プログラム中で、ある条件が満たされているかどうかによって、次に実行するコードを切り替える命令

- **for文**

条件が真の間だけ処理を続けていくループ文

- **while文**

条件が真の間だけ処理を続けていくループ文

【 インクリメント・デクリメント (increment) 】

演算子の短縮したようなものです。

i++	i = i+1	インクリメント
i--	i = i - 1	デクリメント

【 for 処理 】

for文を使用することで、反復処理をおこなうことができる

```
for( 条件式 ){  
    条件式の結果がtrueの場合実行されるスクリプト  
}
```

例) for文

```
for( var i=0; i<10; i++ ){  
    console.log( i ); // 変数iに代入されてる値を表示  
}
```

練習 : for文

```
//1.変数の入れ物を作成  
var str="";  
//2.繰返し処理で、文字列を作成  
for ( var i=0; i<10; i++ ) {  
    str = str + "ループ：" + i + "回目<br>";  
}  
//3.変数「str」に入ってる文字列を pタグid="view"に表示  
document.querySelector("#view").innerHTML = str;
```

【例）配列と反復処理】

配列と一緒に使用することが多い。

//1.配列を作成（必要な分だけ。。。今回は適当数に記述）

```
var ar = ["A","B","C","D","E","F","G","H","I","J"];
```

//2.変数の入れ物を作成

```
var str = "";
```

//3.繰り返し処理で、文字列と配列を組み合わせ作成

```
for ( var i=0; i<10; i++ ) {  
    str = str + "配列：" + ar[i] + "<br>"; //配列:ar[i]  
}
```

//4.変数「str」に入ってる文字列を pタグid="view"に表示

```
document.querySelector("#view").innerHTML = str;
```


【例）反復処理】

HTML要素を作成する

```
<!-- ここにセレクトボックスの値が生成されます -->
<select id="date"></select>

<script>
  //1.変数strを作成:空文字
  var str = "";
  //2.繰り返し処理で、HTMLタグと文字を作成
  for( var i=2011; i<2099; i++ ) {
    str = str + "<option>" + i + "</option>";
  }
  //4.変数「str」に入ってる文字列を pタグid="view"に表示
  document.querySelector("#date").innerHTML = str;

</script>
```

クイズを作る

クイズ練習（1個）

あなたの好きな食べものは？

☐ ケーキ ☐ 生ハム ☐ クレープ

練習：正解 or はずれ！をここにも表示しましょう



フッター

作り方 1 (問題が 1 つの場合)

例)

```
//問題文を表示
var q  = "あなたの好きな食べものは? ";
$("#quest").text( q ); //id="quest"の箇所に表示

//ラジオボタンをクリックしたらイベント発生！
$('input').on("click",function() {
    //1.変数に値を代入
    var ans = 2;    //答えの
    //2.IF文でradioボタンの値と、答えを比較！
    if( $(this).val() == ans ) { //選択値と変数ansを比較
        alert( 'OK!!' );
    }else{
        alert( 'Boo!!' );
    }
});
```

スクリプトが
少ない。

クイズ練習（複数）

1 問目

通るときには閉まって、通らないときには開
いているもの

○踏み切り ○洗濯

練習：正解 or はずれ をここ

フッター

クイズ練習（複数）

2 問目

話することがとても好きな道具は

ページ 127.0.0.1:59162 の記述:

これ以上ダイアログボックスを生成しない

OK

フッター

3 問目

クイズ練習（複数）

世界の真ん中にいる虫は何？



ページ 127.0.0.1:59162 の記述:

正解！

☐ このページでこれ以上ダイアログボックスを生成しない

OK

フッター

作り方 2-1 (問題が複数の場合)

サンプル
jquery/quiz02.html

```
//問題文[配列]
```

```
var qs = [
```

```
  "通るときには閉まって、通らないときには開いているものは何？",
```

```
  "話すことがとても好きな道具は何？",
```

```
  "世界の真ん中にある虫は何？"
```

```
];
```

```
//回答選択肢[多重配列]
```

```
var toi = [
```

```
  [ "踏み切り", "洗濯機", "冷蔵庫" ],
```

```
  [ "スプーン", "シャベル", "しゃもじ" ],
```

```
  [ "てんとう虫", "カマキリ", "蚊" ]
```

```
];
```

```
//答え[配列]
```

```
var ans = [ 1, 2, 3 ];
```

クイズ練習 (複数)

通るときには閉まって、通らないときには開いているものは何？

☐ 踏み切り ☐ 洗濯機 ☐ 冷蔵庫

練習：正解 or はずれ をここにも表示しましょう

フッター

ここまで記述しましょう！

作り方 2-2 (問題が複数の場合)

サンプル
jquery/quiz02.html

```
//問題文[配列]
```

```
var qs = ["通るときには閉まって、通らないときには開いているものは何？",  
          "話すことがとても好きな道具は何？", "世界の真ん中にいる虫は何？" ];
```

```
//-----
```

全ページで記述した配列は以下方法で表示できます。

//1.最初の問題文&回答選択肢&回答値を設定

```
$("#quest").html( qs[0] );    //id="quest"の箇所に表示、最初はゼロ"0"  
とすると qs[0]と0が固定数値を入れるので、複数問題にしたい場合はNG.
```



以下のように記述

//1.最初の問題文&回答選択肢&回答値を設定

```
var i = 0;                      //配列カウント用で用意（追加）  
$("#quest").html( qs[i] );    //id="quest"の箇所に表示、iは最初はゼロ"0"
```

作り方 2-2 (問題が複数の場合)

サンプル
jquery/quiz02.html

```
//回答選択肢[多重配列]
```

```
var toi = [ [ "踏み切り", "洗濯機", "冷蔵庫" ], [ "スプーン", "シャベル", "しゃもじ" ], [ "てんとう虫", "カマキリ", "蚊" ] ];
```

```
//-----
```

```
//1.最初の問題文&回答選択肢&回答値を設定
```

```
var i = 0; //配列カウント用で用意 (追加)
```

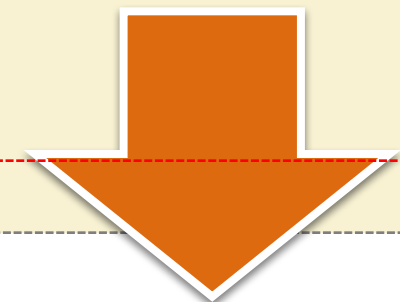
```
$("#quest").html( qs[i] ); //id="quest"の箇所に表示、iは最初はゼロ"0"
```

以下を追加記述
(配列 toi に入ってる文字列を表示する)

```
$("#toi1").html( toi[i][0] ); //選択肢を表示
```

```
$("#toi2").html( toi[i][1] ); //選択肢を表示
```

```
$("#toi3").html( toi[i][2] ); //選択肢を表示
```



作り方 2-2 (問題が複数の場合)

サンプル
jquery/quiz02.html

//1.最初の問題文&回答選択肢&回答値を設定

```
var i = 0; //配列カウント用で用意
```

//2.ラジオボタンをクリックしたらイベント発生！

```
$('#[name=toi]').on("click",function() { //ラジオボタンをクリックしたら動作
    if( $(this).val() == ans[i] ) {      //選択値と配列ans[i]を比較
        alert('正解！');
    } else {
        alert('はずれ！');
    }
    $i++;                                //1をプラスする
    $("#quest").html( qs[i] );           //id="quest"の箇所に表示
    $("#toi1").html( toi[i][0] );        //選択肢を表示
    $("#toi2").html( toi[i][1] );        //選択肢を表示
    $("#toi3").html( toi[i][2] );        //選択肢を表示
});
```

演習

課題：以下内容を来週授業までに！

サンプル
jquery/quiz.html

- ・ 問題数を 6 まで増やす。
- ・ 問題 6 問が終わったら
 - ・ 「 i++; 」 を実行しないようにする。
※次の配列が無いのに参照するため。
 - ・ 「 終了 」 アラートを出し、正解の数を表示
- ・ h1 タグのところに何問目かを表示

クイズ練習（複数）クイズ練習（複数）クイズ練習（複数）

通るときには閉まって、通らないと

1 問目 いているものは何？

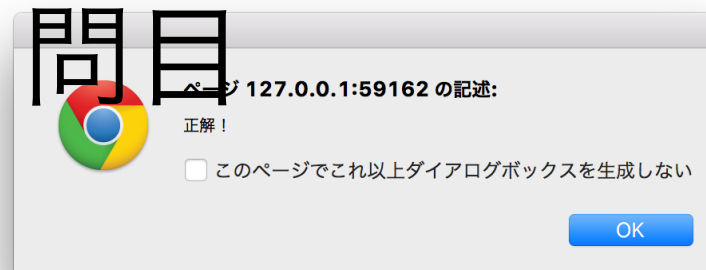
☐ 踏み切り ☐ 洗濯機 ☐ 冷蔵庫

練習：正解 or はずれ をここにも表示しま

フッター

話すことがとても好きな道具は何

2 問目



フッター

世界の真ん中にある虫は何？

3 問目



フッター