

●DOM(Document Object Model)を使って表示操作

```

<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
        background-color:red; }
  </style>
</head>

```

```

<body>
  <h1 id="title">HELLO!</h1>
  <p id="msg">I am a boy.</p>
</body>

```

←<body>に
書いたのに
←書いたのに



```

<script>
  var title= document.querySelector('#title');-----
  var msg = document.querySelector('#msg');-----
  title.textContent = "こんにちは";
  msg.textContent = "表示した後に上書き出来る。";
</script>

```

↑ 後から上書きされてる

※表示した後に文字列を変更できる

</html>

変数 = document.querySelector (要素の指定);
オブジェクト

```

var title= document.querySelector('#title');
title.textContent = "こんにちは";

```

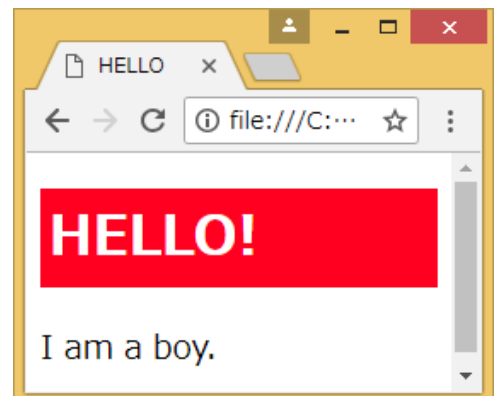
↑ 要素名の指定
要素名からオブジェクトをつかむ
その要素を変更
プロパティ

《先に head の中に入れても html は変化しない》

```

<head>
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
        background-color:red; }
  </style>
  <script>
    var title= document.querySelector('#title');
    var msg = document.querySelector('#msg');
    title.textContent = "こんにちは";
    msg.textContent = "表示した後に上書き出来る。";
  </script>
</head>
<body>
  . . . . .
</body>

```



DOMはhtml要素を読み込んでから動くので先に書くと動かない。

●onload イベントを使って html が読み込まれた時に実行させる

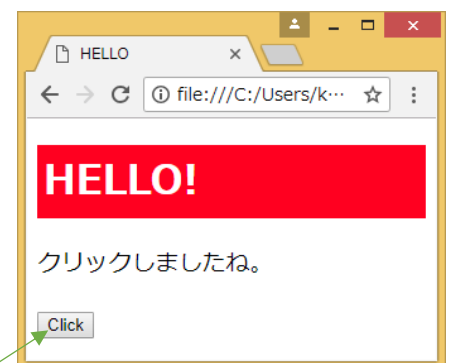
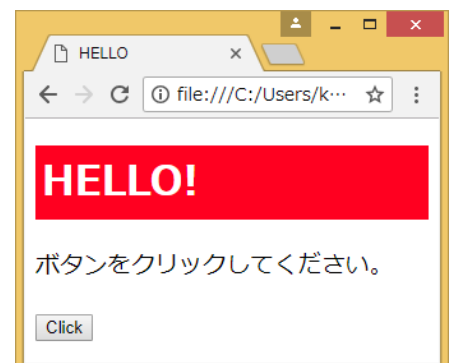
```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
      background-color:red; }
  </style>
  <script>
    function init(){
      var title= document.querySelector('#title');
      var msg = document.querySelector('#msg');
      title.textContent = "こんにちは";
      msg.textContent = "表示した後に上書き出来る。";
    }
  </script>
</head>
<body onload="init();">
  <h1 id="title">HELLO!</h1>
  <p id="msg">I am a boy.</p>
</body>
</html>
```

↑<body>のロードが完了したら関数が動き出す

●画面クリックに反応させる

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
      background-color:red; }
  </style>
  <script>
    function clickNow(){
      var msg = document.querySelector('#msg');
      msg.textContent = "クリックしましたね。";
    }
  </script>
</head>
<body>
  <h1 id="title">HELLO!</h1>
  <p id="msg">ボタンをクリックしてください。</p>
  <button onClick="clickNow();">Click</button>
</body>
</html>
```

※ボタンにクリックを割り当てた例



※<p id="msg" onclick="clickNow();">ここをクリックしてください。</p>

↑こうすると、この文字をクリックすると表示が変わる。

●フォームの利用

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
        background-color:red; }
  </style>
  <script>
    function clickNow(){
      var input1 = document.querySelector('#input1');
      var value = input1.value;
      var msg = document.querySelector('#msg');
      msg.textContent = "こんにちは、" + value + "さん";
    }
  </script>
</head>
<body>
  <h1 id="title">HELLO!</h1>
  <p id="msg">あなたの名前は？</p>
  <input type="text" id="input1">
  <button onclick="clickNow();">Click</button>
</body>
</html>
```

ID:input1 の要素取得
input1 の値取得
ID:msg の要素取得
値の表示方法作成

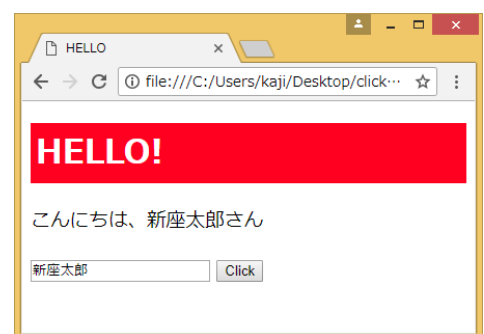
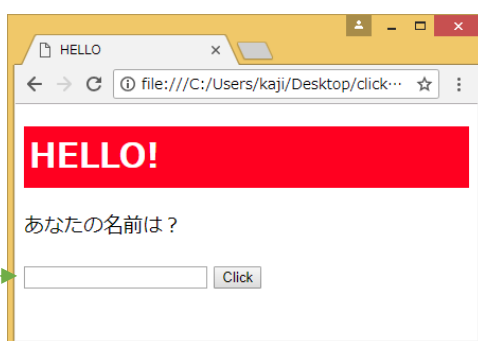
msg に変更があったら変化する

```
<input type="text" id="input1">
```

```
var value = input1.value;
```

```
msg.textContent = "こんにちは、" + value + "さん";
```

テキスト入力枠の ID
その中身(value)
msg に載せる



```
<input type="入力フォームの種類" id="入力フォームの名前">
```

(text)

(checkbox)

(radio)

(file)

(submit)

(reset)

など。

↳HTML 上に置かれる部品にも名前をつける。

※名前を付ける理由：操作がし易くなる。

名前:

パスワード:

学年: ☐ 1年生 ☐ 2年生 ☐ 3年生 ☐ 4年生 ☐ 5年生 ☐ 6年生

好きな科目: ☐ 国語 ☐ 英語 ☐ 算数 ☐ 理科 ☐ 社会 ☐ 体育

宿題ファイル: 選択されていません

●チェックボックスとラジオボタンを使ってみる

```
<!DOCTYPE html>
<html lang="ja">
<head>
```

```
  <meta charset="UTF-8">
  <title>HELLO</title>
  <style>
    body { font-size:14pt; font-weight:plain; }
    h1 { color:white; padding:5px; font-size:24pt;
        background-color:red; }
  </style>
  <script>
    function clickNow(){
      var ck = document.querySelector('#check1');
      var rA = document.querySelector('#radioA');
      var rB = document.querySelector('#radioB');
      var res = "";
      if( ck.checked ){
        res = "あなたは大人ですね。";
      } else {
        res = "まだ子供ですね。";
      }
      if( rA.checked ){
        res += "女性ですね。";
      }
      if( rB.checked ){
        res += "男性ですね。";
      }
      var msg = document.querySelector('#msg');
      msg.textContent = res;
    }
  </script>
```

```
</head>
<body>
```

```
  <h1 id="title">HELLO!</h1>
  <p id="msg">あなたの名前は？</p>
  <p>
    <input type="checkbox" id="check1">
    <label for="check1">成人</label>
  </p>
  <p>
    <input type="radio" name="radio1" id="radioA">
    <label for="radioA">女性</label><br>
    <input type="radio" name="radio1" id="radioB">
    <label for="radioB">男性</label>
  </p>
  <button onclick="clickNow();">Click</button>
```

```
</body>
</html>
```

※<label for="check1">成人</label>



これで挟むと、「成人」の文字をクリックしてもラジオボタンと同様に動く。

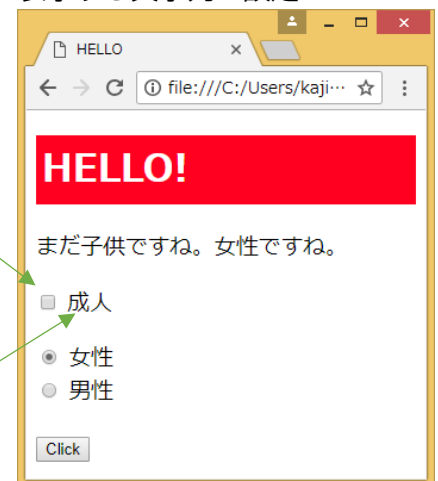


チェックボックスを取得
 チェックボックスを取得
 チェックボックスを取得
 文字列は空にする
 もし、ckにチェックがあったら
 大人ですね。
 そうじゃなかったら
 子供ですね。

もし、rAにチェックがあったら
 女性ですね。

もし、rBにチェックがあったら
 男性ですね。

msg要素を取得
 表示する文字列を設定



《紛らわしいので注意が必要》

input type=	name	id	関数内 var
checkbox		check1	ck
radio	radio1	radioA	rA
radio	radio1	radioB	rB