

資料編

図1：サイト全体図（トップページ）

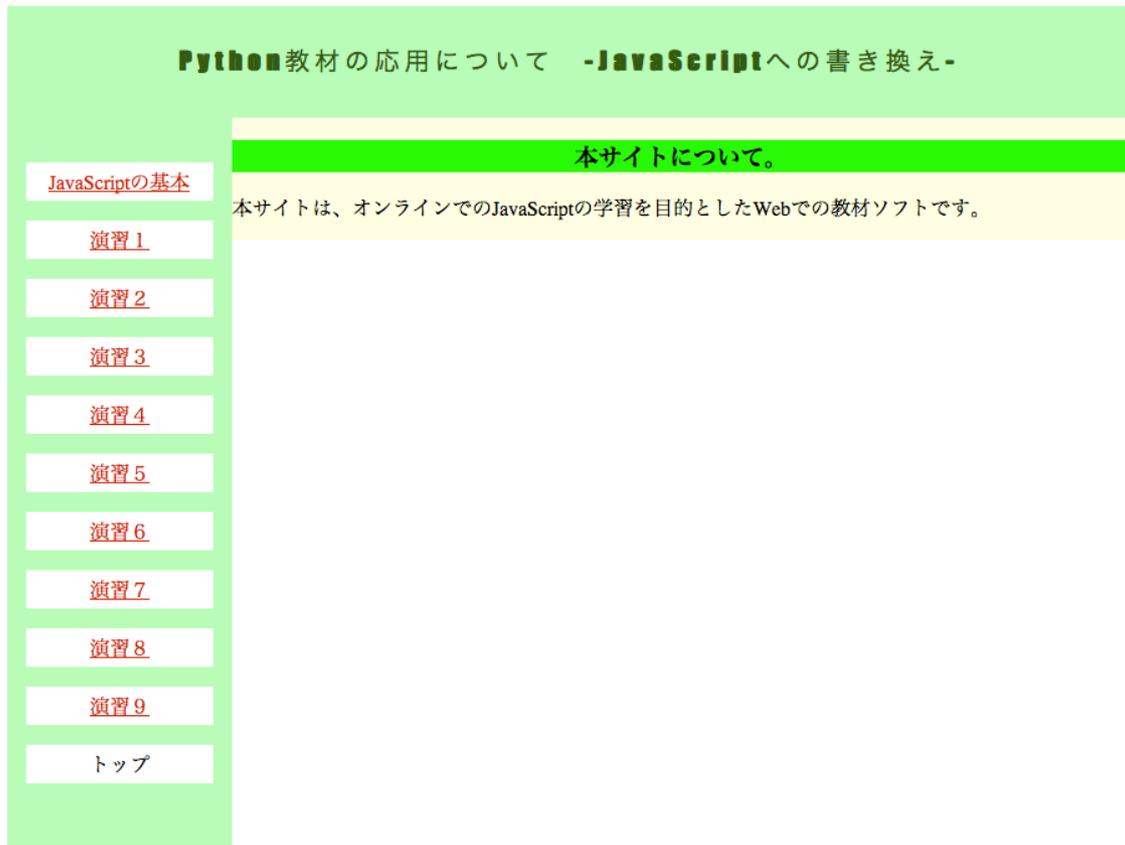


図 2 : JavaScript の基本

Python教材の応用について -JavaScriptへの書き換え-

JavaScriptの基本	JavaScriptの基本
演習 1	JavaScriptとは JavaScript(ジャバ スクリプト)とは、インターネット上のWebページを記述するプログラム言語の1つである。JavaScriptを組み込むと、文字や画像、ページ等に種々の自動的な「バリエーション」や「動き」が与えられ、計算や情報の検索も可能な仕組みに変化させることができる。JavaScriptは、補助言語で、文字や画像、ページ等を動かす仕掛けを作るものである。このJavaScriptを組み込まれたページは、HTMLで作成された「静的ページ」に対し、「動的ページ」とも呼ばれる。
演習 2	
演習 3	
演習 4	
演習 5	HTML文章内でのJavaScript記述方法 JavaScriptをHTML文章の中で使用する方法は4つある。
演習 6	1. <script>タグ内に記述する
演習 7	2. HTMLタグのイベントハンドラとして記述する
演習 8	3. タグ内に&{数式};で記述する
トップ	4. 外部ファイル (別ファイルのソース) として呼び出す
	<script>タグについて HTML文章内の<script>タグを使用して記述する場合、以下のようにする。 <pre><script> JavaScript命令および関数... </script></pre> ※デフォルトの言語として指定する場合↓をヘッダに記述 <meta http-equiv="Content-Script-Type" content="text/javascript"> ※<script>タグ内で指定する場合 <script type="text/javascript"> JavaScript命令および関数... </script>

図 3 : 演習 1

Python教材の応用について -JavaScriptへの書き換え-

演習 1

[JavaScriptの基本](#)

演習 1

[演習 2](#)

[演習 3](#)

[演習 4](#)

[演習 5](#)

[演習 6](#)

[演習 7](#)

[演習 8](#)

[トップ](#)

電卓としての利用

JavaScriptでは数値の計算をすることができる。
例えば、 $1 + 2 = 3$ という計算をJavaScriptを用いて行う場合、

```
<script type="text/javascript">
  document.write(1 + 2);
</script>
```

と入力する。すると数値が計算され、値が表示される、

[実行してみる](#)

算術演算子一覧

足し算...+
引き算...-
かけ算...*
割り算.../

変数を使用して計算する方法もある、
例えば、 $1 + 2 = 3$ という計算を変数を用いて計算する場合...

```
<script type="text/javascript">
  var a;      ←変数aを宣言
  a = 5;     ←aという変数に5を代入
  var b;     ←変数bを宣言
  b = 5;     ←bという変数に5を代入
  document.write(a + b);
</script>
```

var a = 5;
var b = 5;

というように省略することも可能、

図 4 : 演習 2

Python教材の応用について -JavaScriptへの書き換え-

演習 2

[JavaScriptの基本](#)

[演習 1](#)

[演習 2](#)

[演習 3](#)

[演習 4](#)

[演習 5](#)

[演習 6](#)

[演習 7](#)

[演習 8](#)

[トップ](#)

文字列の足し算と数値の足し算

「123」と「"123"」は異なったものである。
「123」は数値で、「"123"」は数字(文字列)である。
実際にそれぞれの場合の足し算を動かしてみる。

```
<script type="text/javascript">
  var a = 123;
  var b = 123;
  document.write(a+b);
</script>
```

[実行してみる](#)

```
<script type="text/javascript">
  var a = "123";
  var b = "123";
  document.write(a+b);
</script>
```

[実行してみる](#)

上は数値「123」と「123」が足し算されるので答えは「246」になる。
下は文字列「123」と「123」が足され、文字列を追加する形になるので
答えは「123123」になる。

文字列を変数に代入する

```
<script type="text/javascript">
  var name = "Kazuki Sugita";
  var mail = "wf0748054@hi.otani.ac.jp";
  document.write("From:", name, "<", mail, ">");
</script>
```

[実行してみる](#)

図 5 : 演習 3

Python教材の応用について -JavaScriptへの書き換え-

演習 3

[JavaScriptの基本](#)

[演習 1](#)

[演習 2](#)

演習 3

[演習 4](#)

[演習 5](#)

[演習 6](#)

[演習 7](#)

[演習 8](#)

[トップ](#)

リスト

JavaScriptでのリストの処理はとても便利である。
JavaScriptにおいてのリストは、複数の項目をカンマで区切り、[]で囲んだものである。

```
<script type="text/javascript">
  var cities = ["東京", "大阪", "名古屋", "京都", "札幌"];
  document.write(cities);
</script>
```

と入力する。すると結果が表示される。

[実行してみる](#)

数値のリスト

数値をリスト化する場合、次のようになる。

```
<script type="text/javascript">
  var numbers = ["1", "2", "4", "8", "16"];
  document.write(numbers);
</script>
```

[実行してみる](#)

リストの中のリスト

```
<script type="text/javascript">
  var list_in_list = [["Fukuda", 123], ["Miyashita", 456], ["Watanabe", 789], ["Kataoka", 234]];
  document.write(list_in_list);
</script>
```

中の[]は文字列と数値のリストである。
このように異なった型のもので、カンマで区切って[]で囲めばリストになる。

[実行してみる](#)

図 6 : 演習 4

Python教材の応用について -JavaScriptへの書き換え-

演習 4

[JavaScriptの基本](#)

[演習 1](#)

[演習 2](#)

[演習 3](#)

[演習 4](#)

[演習 5](#)

[演習 6](#)

[演習 7](#)

[演習 8](#)

[トップ](#)

制御文

プログラムを開発する際に、条件により様々な処理内容に分岐させる場合がある。このようなプログラムの流れを制御する場合に活躍するのが「制御文」である。

if / else文

「もし〜ならば」と考える場合に使用する制御文である。
else と出てきたら「そうでなければ」と訳す。

以下は、変数[pass]が[12345]の場合、アラートダイアログに「ログインできました」と、[12345]以外の場合「ログインに失敗しました」と表示するプログラムである。

```
<html>
<body>
<script type="text/javascript">
function inputYourName(){
  var pass = prompt("パスワードを入力してください");
  if (pass == "12345"){
    alert("ログインできました");
  } else {
    alert("ログインに失敗しました");
  }
}
</script>
<input type="button" value="クリック" onclick="inputYourName()"/>
</body>
</html>
```

[実行してみる](#)

else if文

3つ以上の分岐処理の場合は「else if文」を用いる。

```
<html>
<body>
<script type="text/javascript">
```

図 7 : 演習 5

Python教材の応用について -JavaScriptへの書き換え-

JavaScriptの基本	演習 5
演習 1	制御文
演習 2	プログラムを開発する際に、条件により様々な処理内容に分岐させる場合がある。このようなプログラムの流れを制御する場合に活躍するのが「制御文」である。
演習 3	for文
演習 4	for文は、同じ処理を繰り返したい場合に有効である。
演習 5	以下は、0～9までの数値を、HTMLドキュメントにプログラムである。
演習 6	<pre><script type="text/javascript"> for (var i = 0; i < 10; i++) { document.write(i, '
'); } </script></pre>
演習 7	
演習 8	実行してみる
トップ	while文
	繰り返しの回数をあらかじめ決められない場合は「while文」を用いる。
	<pre><script type="text/javascript"> var i = 0; while (i > 10) { document.write(i, '
'); i++; } </script></pre>
	実行してみる
	～問題～
	1. 1から10までの数を合計するプログラムを書いてみよう。

図 8 : 演習 6

Python教材の応用について -JavaScriptへの書き換え-

演習6

[JavaScriptの基本](#)

[演習1](#)

[演習2](#)

[演習3](#)

[演習4](#)

[演習5](#)

[演習6](#)

[演習7](#)

[演習8](#)

[トップ](#)

関数

関数とは、簡単に説明すると、複数の処理を1つにまとめて扱えるようにしたものである。同じような処理を繰り返す場合、関数を使うと効率よく作業ができる。関数への入力値を「引数」、関数の実行結果を「戻り値」と呼ぶ。

関数の定義

function文による関数定義

```
<script type="text/javascript">
  function calcCone(radius, height) {
    var dimension = radius * radius * Math.PI;      ←「Math.PI」で円周率
    var taiseki = dimension * height / 3;
    return taiseki;
  }
  var radius = prompt("半径");
  var height = prompt("高さ");
  var kekka = calcCone(radius, height);
  alert("円錐の体積は"+kekka+"cm3です。");
</script>
```

[実行してみる](#)

```
function 関数名(引数1, 引数2...){
  処理の文
  return 返す値
}
```

というような書き方をする。

～問題～

1. 名前、身長、体重を聞き、計算して「あなたのBMI値は〇〇です。」と表示するプログラムを作ってみよう。

図 9 : 演習 7

Python教材の応用について -JavaScriptへの書き換え-

演習7

[JavaScriptの基本](#)

[演習1](#)

[演習2](#)

[演習3](#)

[演習4](#)

[演習5](#)

[演習6](#)

[演習7](#)

[演習8](#)

[トップ](#)

九九の表

1×1 1×2 1×3 ... 1×8 1×9
2×1 2×2 2×3 ... 2×8 2×9
3×1 3×2 3×3 ... 3×8 3×9

上のような式の計算結果だけを表示するプログラムを作る。

```
<script type="text/javascript">
var i = 1;
while (i < 10) {           ←iが10未満の場合に以下の作業を繰り返す
  var a = 1
  while (a <10) {         ←aが10未満の場合に以下の作業を繰り返す
    document.write(i * a + " ");
    a++;                 ←aを1増やす
  }
  document.write("<br>");
  i++;                   ←iを1増やす
}
</script>
```

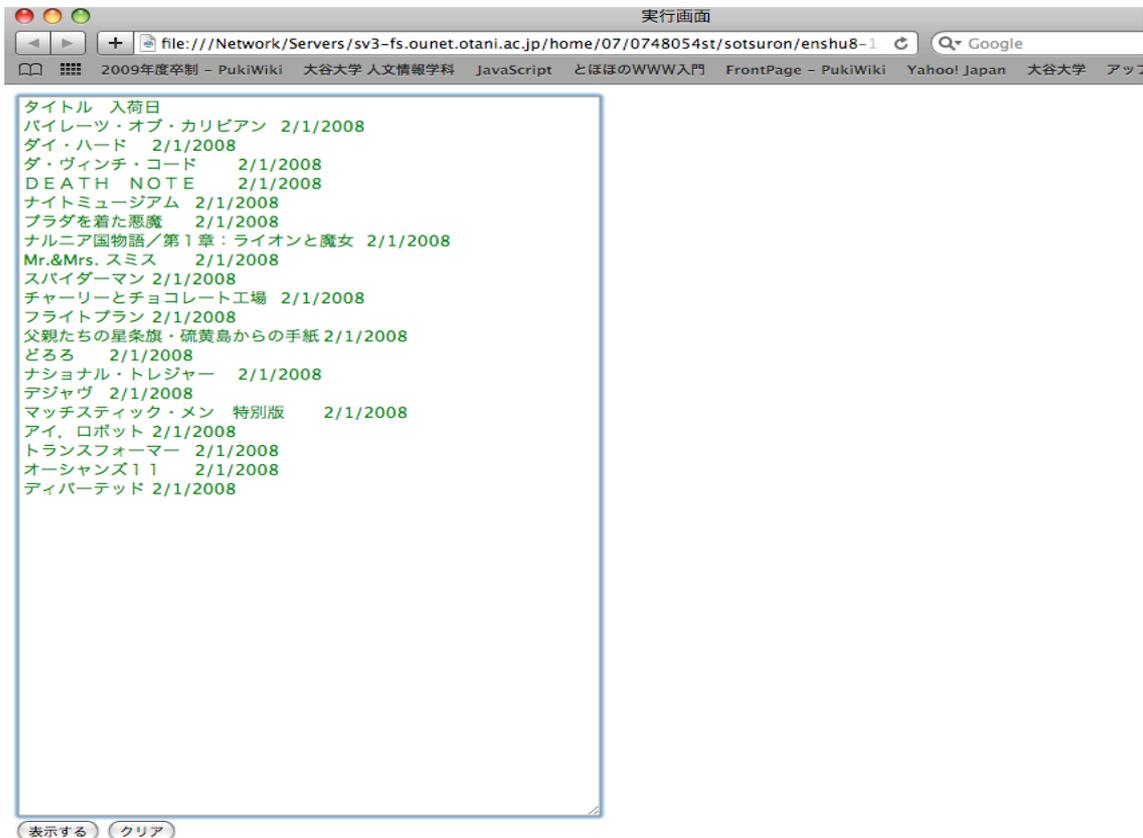
[実行してみる](#)

図 10 : 演習 8

Python教材の応用について -JavaScriptへの書き換え-

演習 8	
JavaScriptの基本	タブ区切りのデータからHTMLの表を作る
演習 1	タブ区切りのデータ
演習 2	Excelなどの表計算ソフトやデータベースで作ったデータを再利用するには、一行のレコードが項目毎にタブで区切られ、各レコードは改行で区切られた、複数行のテキストデータにするのがよい。
演習 3	福田洋一 タブ 大谷大学 タブ チベット仏教 タブ 東京 改行 渡辺啓真 タブ 大谷大学 タブ 倫理学 タブ 京都 改行
演習 4	こういったテキストデータをHTMLのテーブルに直す。
演習 7	こういうテキストデータを処理する場合、一行を読み込み、タブを区切りにしてリストに分解、リストの項目を番号順にforで繰り返し処理をする。
演習 8	行を区切りに、リストに分解するには、 <pre>リスト変数 = タブ区切りの文字列.split("\n")</pre> タブを区切りに、リストに分解するには、 <pre>リスト変数 = タブ区切りの一行の文字列.split("\t")</pre> とする。 すると、リスト変数の中身は、 <pre>['福田洋一','大谷大学','チベット仏教','東京']</pre> というリストになる。
トップ	HTMLのテーブル 次にこれをHTMLのテーブルに直す。 HTMLテーブルは table 要素の中に作り、一行(一レコード)はそれぞれtrタグに囲まれる。

図 11 : 演習 8 例文の実行例



実行画面

file:///Network/Servers/sv3-fs.ounet.otani.ac.jp/home/07/0748054st/sotsuron/enshu8-1

2009年度卒制 - PukiWiki 大谷大学 人文情報学科 JavaScript とほほのWWW入門 FrontPage - PukiWiki Yahoo! Japan 大谷大学

タイトル	入荷日
パイレーツ・オブ・カリビアン	2/1/2008
ダイ・ハード	2/1/2008
ダ・ヴィンチ・コード	2/1/2008
DEATH NOTE	2/1/2008
ナイトミュージアム	2/1/2008
ブラダを着た悪魔	2/1/2008
ナルニア国物語 / 第1章:ライオンと魔女	2/1/2008
Mr.&Mrs. スミス	2/1/2008
スパイダーマン	2/1/2008
チャーリーとチョコレート工場	2/1/2008
フライトプラン	2/1/2008
父親たちの星条旗・硫黄島からの手紙	2/1/2008
だろろ	2/1/2008
ナショナル・トレジャー	2/1/2008
デジャヴ	2/1/2008
マッチスティック・メン 特別版	2/1/2008
アイ, ロボット	2/1/2008
トランスフォーマー	2/1/2008
オーシャンズ11	2/1/2008
ディパーテッド	2/1/2008

図 12 : テキストボックス内の解説

リストの項目を取り出す

リストの項目を個々に表示するには、番号をつけて何番目の項目か指定する。

```
<script type="text/javascript">
  var list_in_list = [{"Fukuda", 123}, {"Miyashita", 456}, {"Watanabe", 789}, {"Kataoka", 234}];
  document.write(list_in_list[0] + "<br>");           //0から始まることに注意
  document.write(list_in_list[1] + "<br>");
  document.write(list_in_list[0][1] + "<br>");         //リストの中のリスト
  document.write(list_in_list[0] + " → " + list_in_list[2][1]); //文字列も追加できる。
</script>
```

"
"で改行できる。

[実行してみる](#)

図 13 : 「実行してみる」 リンク→実行画面

九九の表

1×1 1×2 1×3 ... 1×8 1×9
2×1 2×2 2×3 ... 2×8 2×9
3×1 3×2 3×3 ... 3×8 3×9

上のような式の計算結果だけを表示するプログラムを作る。

```
<script type="text/javascript">
  var i = 1;
  while (i < 10) {          //iが10未満の場合に以下の作業を繰り返す
    var a = 1
    while (a < 10) {       //aが10未満の場合に以下の作業を繰り返す
      document.write(i * a + " ");
      a++;                 //aを1増やす
    }
    document.write("<br>");
    i++;                   //iを1増やす
  }
</script>
```

[実行してみる](#)



実行画面

file:///Network/Servers/sv3-fs.ounet.otani.ac.jp/home/07/0748054st/sotsuron/enshu

2009年度卒制 - PukiWiki 大谷大学 人文情報学科 JavaScript とほほのWWW入門 FrontPage - Puk

```
1 2 3 4 5 6 7 8 9
2 4 6 8 10 12 14 16 18
3 6 9 12 15 18 21 24 27
4 8 12 16 20 24 28 32 36
5 10 15 20 25 30 35 40 45
6 12 18 24 30 36 42 48 54
7 14 21 28 35 42 49 56 63
8 16 24 32 40 48 56 64 72
9 18 27 36 45 54 63 72 81
```