

6. フレームとJavaScript

08t4083t 真下飛瑠

6.1 フレームの概要

6.1.1 フレームを作成するためのHTMLタグ

フレーム機能を使用するには、以下が必要

- フレームを管理するHTMLドキュメント
- 個々のフレームを表示するHTMLファイル

フレームを管理するHTMLドキュメントでは
<body></body>タグの代わりに

- **<frameset></frameset>**タグ
- **<frame src>**タグ

を記述

- `<frameset></frameset>`タグではウィンドウを縦横どちらに分割するかを設定する
 - 横分割 **cols**アトリビュート
 - 縦分割 **rows**アトリビュート
- 分割するサイズはパーセントorピクセル数で指定
- ピクセル数指定の場合、分割するフレームのうち最低1つは「*」を指定する。「*」で指定したフレームのサイズはWebブラウザが自動で計算する
- ウィンドウを縦に分割し、さらにその下側のフレームを左右に分割したい場合は
`<frameset></frameset>`タグを入れ子にする

例 ウィンドウを縦に2分割、下のフレームを横に2分割する

```
<html>
<head>
<title>フレームテスト</title>
</head>
<frameset rows="80, * ">
  <frame src="top.html" name="top">
  <frameset cols="120, * ">
    <frame src="left.html" name="left">
    <frame src="bottom.html" name="bottom">
  </frameset>
</frameset>
</html>
```

- <frame>タグではフレームに表示するドキュメントのURLやスクロール方法などを、アトリビュートで設定する

アトリビュート	説明
src= "URL"	ドキュメントのURL
name= "フレーム名"	フレーム名
marginwidth= "ピクセル数"	フレームとテキストの横方向のマージン
marginheight= "ピクセル数"	フレームとテキストの縦方向のマージン
scrolling= "スクロール方法"	「yes」は常にスクロールバー表示
	「no」は常にスクロールバー非表示
	「auto」はドキュメントのサイズに応じて表示
noresize	これを設定しておくで、ユーザがフレームサイズを変更できなくなる

- フレーム機能をサポートしていないWebブラウザのために、**<noframe>**タグが用意されている
- **<noframe></noframe>**タグの間に記述したHTMLは、フレーム機能をサポートしていないWebブラウザのみで表示される
- フレーム機能を使う場合のHTML4.01の文書型宣言は次のようになる

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
  Frameset//EN"  
  "http://www.w3.org/TR/html4/frameset.dtd">
```

6.1.2 フレームを使用した場合のリンク先の指定

- `<a href>`タグの**target**アトリビュートを使用すれば、指定したフレームにリンク先のページを表示できる

`履歴`

- targetアトリビュートは次のような指定も可能である

設定	説明
_blank	新規ウィンドウを開いて表示
_self	自分自身に表示
_parent	1つ前のフレーム分割に戻って表示
_top	フレーム分割を解除し、ウィンドウ全体に表示

6.2 frameオブジェクト

6.2.1 フレームを管理するframeオブジェクト

- JavaScriptでは<frame>タグによって生成されたフレームを**frame**オブジェクトとして管理
- 個々のframeオブジェクトにとって、<frameset>タグが記述されているHTMLドキュメントは親ウィンドウとなる
- 親ウィンドウを参照するには**parent**プロパティを使用
- 親ウィンドウからは、それぞれのframeオブジェクトはプロパティとして参照できる

別フレームのオブジェクトへのアクセス方法

- フレーム名によってアクセスする場合
windowリファレンス. フレーム名
- frame配列によってアクセスする場合
windowリファレンス. **frames** []

例

```
<frameset rows="50, * ">  
  <frame src="top.html" name="top">  
  <frame src="bottom.html" name="bottom">  
</frameset>
```

のとき、bottomフレームのURLを変更したい場合、

```
parent.bottom.location="url";
```

別フレームの関数や変数へのアクセス方法

- 前述と同様に、parentプロパティを使用
- topフレームのHTMLドキュメントに定義された関数go()を呼び出すには、
parent. top. go();
- <frameset>タグを記述したHTMLドキュメントに、すべてのフレームで利用する関数を定義することができる
- その場合、parentプロパティの後に関数を直接ピリオド「.」で接続して記述する

別ウィンドウのフレームへのアクセス

- 現在のウィンドウとは別のウィンドウにあるフレームへアクセスすることもできる
- その場合、先頭にウィンドウのリファレンスを指定する

```
window2.rightFrame.document.backgroundColor="green"
```

6.2.3 frameオブジェクトのメソッド、プロパティ、イベントハンドラ

- 基本的にwindowオブジェクトのものと同じ

プロパティ	説明
frames 配列	現在のフレームに含まれるframeオブジェクトの配列
name	フレーム名。<frame>タグのnameアトリビュートの値
parent	親フレームへの参照
self	自フレームへの参照
window	現在のフレームのウィンドウ自身への参照

メソッド	説明
setTimeout()	タイマーをスタート
clearTimeout()	タイマーを停止
blur()	フォーカスを外す
focus()	フォーカスを取得する

イベントハンドラ	説明
onblur	フォーカスを失ったとき
onfocus	フォーカスを得たとき

- windowオブジェクトのイベントハンドラの1つであるonloadイベントハンドラは、<frameset>タグ内に記述することができる。その場合、すべてのフレームのロードが完了した時点でonloadイベントハンドラが呼ばれる

課題

- ウィンドウを横に2分割、さらにその右側のフレームを縦に2分割する(順にframe1,2,3)。
- それぞれのフレームにボタンを1つずつ用意
- frame1,frame2のボタンはクリックでframe2へHTMLファイル読み込み
- frame3のボタンはクリックで全体へHTMLファイル読み込み