

navigator でブラウザを調べる 1

ここから、**navigator** について少し調べていきたいと思います。主に閲覧しているブラウザの情報を取得します。ブラウザによってページを分けたり、スタイルシートを変更したりする時に用いることができます。

ブラウザを調べるのは、実は簡単ではありません。アプリケーション名、バージョン、エージェント名などをかき集めて、ようやく特定できます。

ここでは、Internet Explorer(IE), FireFox, Opera, Google Chrome, Safari の5つで、**navigator** で取得される値について調査してみることにしましょう。そして次のページで、それらを分類する方法について考えてみたいと思います。

アプリケーション名を取得する

まずはアプリケーション名を取得してみることにしましょう。HTML の BODY 内に以下のように記入してみてください。**navigator.appName** でアプリケーション名を取得することができます。

```
<script>
```

```
document.write(navigator.appName);
```

```
</script>
```

上記の結果： Netscape

各ブラウザでどのように表示されるかは、以下の通りです。

IE

Microsoft Internet Explorer

FireFox

Netscape

Opera

Opera

Chrome

Netscape

Safari

Netscape

この結果を見ると、Internet Explorer と Opera は直ぐに識別できることが分かります。しかし FireFox, Chrome, Safari はどれも「Netscape」という文字を返します。

バージョンを調べる

では次にブラウザのバージョンを調べてみることにしましょう。 **navigator.appVersion** で

取得できます。

```
<script>
```

```
document.write(navigator.appVersion);
```

```
</script>
```

<上記の結果>

5.0 (Windows)

他のブラウザでは大体以下のような感じです（私の環境での結果です）

IE

5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; AskTbFXTV5/5.12.2.16749; .NET4.0E)

Firefox

5.0 (Windows)

Opera

9.80 (Windows NT 6.1; U; en)

Chrome

5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko) Chrome/13.0.782.218 Safari/535.1

Safari

5.0 (Windows; U; Windows NT 6.1; ja-JP) AppleWebKit/533.21.1 (KHTML, like Gecko) Version/5.0.5 Safari/533.21.1

「Safari」という文字は Chrome と Safari の両方にあります。しかし「Chrome」という文字は Google Chrome しかありません。これらを if 文で分岐すれば、2 種類のブラウザを判別できます。他のブラウザはバージョンでは見分けることができません。

一応ここまでで IE, Opera, Chrome, Safari の 4 つは判別できるようになりました。残りが Firefox としても良いのですが、別のブラウザである可能性もあります。できたら「Firefox」という文字を見たいのですが、それは可能でしょうか？最後の部分で検証してみたいと思います。

エージェント名

ブラウザ判別の最後は、エージェント名です。navigator.userAgent で調べることができます。

```
<script>
```

```
document.write(navigator.userAgent);
```

```
</script>
```

<上記の結果>

Mozilla/5.0 (Windows NT 6.1; WOW64; rv:29.0) Gecko/20100101 Firefox/29.0

他のブラウザの結果は、大体以下の通りです（私の環境です）

IE

Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; AskTbFXTV5/5.12.2.16749; .NET4.0E)

Firefox

Mozilla/5.0 (Windows NT 6.1; WOW64; rv:6.0.1) Gecko/20100101 Firefox/6.0.1

Opera

Opera/9.80 (Windows NT 6.1; U; en) Presto/2.9.168 Version/11.50

Chrome

Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko) Chrome/13.0.782.218 Safari/535.1

Safari

Mozilla/5.0 (Windows; U; Windows NT 6.1; ja-JP) AppleWebKit/533.21.1 (KHTML, like Gecko) Version/5.0.5 Safari/533.21.1

無事「Firefox」の文字を確認できました。 ついでにブラウザのバージョンもエージェントで調べることができます（IE だけ少し特殊ですが）。

さて、`userAgent` を最初から使えば5つのブラウザを識別できるではないか、と考えられた方もおられると思います。 その通りです。しかし幾つかの資料を見てみると、このエージェントは詐称が可能であると述べられていました。 それで、他の方法も考えてみたわけです。

では次のページで、この3つの `navigator` を利用したブラウザ判別スクリプトを組んでみることにします。

`navigator` でブラウザを調べる 2

それを使って、実際にブラウザを判別するためのスクリプトを組んでみることにしましょう。 今のページでは Internet Explorer, Firefox, Opera, Google Chrome, Safari, その他のブラウザ、と分けてみたいと思います。

<サンプル>

現在使用中のブラウザ： **FireFox**

上記サンプルでは、ブラウザが正しく判別されているでしょうか？ もし間違っているなら、ぜひ連絡フォーム等で知らせて下さい。

ブラウザ判別の流れ

前のページでは、**appName** で **IE** と **Opera**, **appVersion** で **Safari** と **Chrome** が識別できることを見ました。ではその方法に則ったブラウザ判別プログラムを作ってみることにしましょう。もちろん **userAgent** を使えば一発で判別できますが、それはまた別所で紹介できたらと思います。

ブラウザ判別の概要をフローチャートで示すと以下ようになります。この流れでスクリプトを組んでみたいと思います。

ブラウザ判別スクリプト

HTML の BODY 内に以下のスクリプトを記入してみましょう（上記のサンプルもこのスクリプトで判別しています）。

<script>

```
var nvName = navigator.appName;  
var nvVersion = navigator.appVersion;  
var nvUsAgent = navigator.userAgent;
```

```
// 1. Internet Explorer か調べる
```

```
if ( nvName.indexOf("Internet Explorer") > -1 )  
{  
    document.write("Internet Explorer");  
}
```

```
else
```

```
{
```

```
    // 2. Opera か調べる
```

```
    if ( nvName.indexOf("Opera") > -1 )  
    {  
        document.write("Opera");  
    }
```

```
    else
```

```
{
```

```

// 3. 「Safari」が含まれるか調べる
if ( nvVersion.indexOf("Safari") > -1)
{
    // 4. Safari か Chrome を見分ける
    if( nvVersion.indexOf("Chrome") > -1)
    {
        document.write("Google Chrome");
    }
    else
    {
        document.write("Safari");
    }
}
else
{
    // 5. FireFox かその他のブラウザかを見分ける
    if( nvUsAgent.indexOf("Firefox") > -1 )
    {
        document.write("FireFox");
    }
    else
    {
        document.write("その他のブラウザ");
    }
}
}
}

```

</script>

文字列.indexOf("検索文字列")が大活躍しています。indexOf()は検索文字列があれば先頭から何文字目かを取得し、文字列が無ければ-1を返します。ですからif文を使って-1より大きいかな否かを調べることで文字が含まれているか判別し、処理を分岐できます。

次のページでは、ブラウザのバージョンを調べてみたいと思います。こちらはuserAgentに頼ることになります。

ブラウザのバージョンを調べる 1

このページでは、`navigator` を使ってブラウザのバージョンを調べる方法について考えます。前のページでは `appName` や `appVersion` も組み合わせてブラウザを判別しましたが、今回は煩雑になるので `userAgent` のみを使ってバージョンを取得したいと思います。

ブラウザのバージョンによって `JavaScript` の解釈が違ったり、レイアウトが崩れたりすることがあります（特に `Internet Explorer`）。それを避けるためにバージョン別にページを用意したりすることがあります。勿論そんな面倒なことを必ずしなくてはいけないという訳ではありませんが、知識として知っておくと役に立つこともあるでしょう。

このページでは `Internet Explorer` と `FireFox` のバージョンを取得する方法について考えます。

ブラウザバージョンを取得する

バージョンを調べるには、まずどのブラウザが判別しなければなりません。 `userAgent` を見ると、`Internet Explorer` には「`MSIE`」という文字列が、`FireFox` には「`Firefox`」という文字列が含まれます。それを `indexOf0` で検索して、その後続くバージョンを切り取れば OK です。

なお、各ブラウザにおける `userAgent` の値は、古籴氏の [userAgent 一覧](#) が非常に参考になります。

では `HTML` の `body` 内に、以下のスクリプトを記入してみてください。

```
<script>
```

```
var nvUA =navigator.userAgent; //userAgent を取得して格納
```

```
var cutSt,cutEd; //切り取り開始点と終了点
```

```
var bwVer; //ブラウザバージョンを格納する変数
```

```
//最初に Internet Explorer か調べます
```

```
cutSt = nvUA.indexOf("MSIE");
```

```
//「MSIE」の文字列があれば、バージョンを取得する
```

```
if ( cutSt != -1 )
```

```
{
```

```
    //バージョン番号の後に「;」があるのでそれを検索
```

```
    cutEd = nvUA.indexOf(";",cutSt);
```

```
    //「MSIE」+半角スペースの 5 文字を加えた位置から切り取る
```

```
    bwVer = nvUA.substring(cutSt+5,cutEd);
```

```
//頭に「Internet Explorer」を付けてバージョンを書き出す
document.write("Internet Explorer "+bwVer);
}

//今度は FireFox か調べます
cutSt = nvUA.indexOf("Firefox");

//「Firefox」の文字列があれば、バージョンを取得する
if (cutSt != -1)
{
    //FireFox の場合、バージョンは最後。つまり終了点は文字列全体の長さ。
    cutEd = nvUA.length;
    //「Firefox」 + 「/」の 8 文字を加えた位置から切り取ります
    bwVer = nvUA.substring(cutSt+8,cutEd);

    //頭に「FireFox」を付けてバージョンを書き出す
    document.write("FireFox " + bwVer);
}

</script>
```

<上記スクリプトの結果> FireFox 29.0

スクリプトの解説

では上記スクリプトの解説をしていきましょう。

変数の宣言

最初に[変数](#)をまとめて宣言しておきました。変数 nvUA には [navigator.userAgent](#) の値を格納しておきます。その中から必要部分を切り取るための開始位置と終了位置を格納するために変数 cutSt, 変数 cutEd を宣言し、取得したバージョンを格納する変数 bwVer も宣言しておきました。

Internet Explorer か調べる

ついで、IE かどうか調べます。IE なら「MSIE」という固有の文字列が存在します。これを [indexOf\(\)](#) で検索し、文字位置を変数 cutSt に代入します。

変数 cutSt の値が-1 ではない時（つまり「MSIE」が存在する時）、[if 文](#)の中の処理を行います。if 文の中で IE のバージョンを取得していきます。

`userAgent` の値を見ると「...MSIE 9.0;...」といった感じでバージョンが記入されています。ですからバージョンの直ぐ後にある「;」の位置を検索することで、バージョン部分を切り取ることができます。

`indexOf()`で検索文字を「;」とし、検索開始位置を変数 `cutSt` に指定します。こうすることで「MSIE」という文字列が出てくる位置から見て最初の「;」の位置を取得できます。取得した位置情報を、変数 `cutEd` に代入します。

続いて `substring()` で文字の切り取りを行います。切り取り開始位置ですが、そのまま変数 `cutSt` を指定すると「MSIE」の文字列の頭から切り取ります。今回はバージョンの数字部分だけ取得したいので、「MSIE」の4文字と、バージョンの間にある「半角スペース」の1文字を足した合計5文字を加えた位置から切り取ります。終了位置は変数 `cutEd` になります。

最後に `document.write()` でブラウザとバージョンを書き出します。

Firefox か調べる

続いて Firefox か調べています。Firefox なら「Firefox」という文字列が存在します。これを `indexOf()` を使って検索し、文字列が存在するか調べて if 文で処理を分岐します。

Firefox の場合、バージョン情報は `userAgent` の一番最後にあります。ですから切り取り終了点は文字列全体の長さで代用できます。文字列長を得るのは `length` でした。それを変数 `cutEd` に代入します。

`userAgent` の値を見ると「...Firefox/6.0.2」といった感じでバージョンが記載されています。ですからバージョンの数字部分だけ切り取るには、開始点を変数 `cutSt` から「Firefox/」の8文字を加えた部分にします。終了点は変数 `cutEd` です。

最後に、ブラウザとバージョンを `document.write()` で書き出しています。

ブラウザのバージョンを調べる 2

前のページでは Internet Explorer と Firefox のブラウザバージョンを調べる方法について見ました。このページでは残るブラウザ、Opera, Safari, Google Chrome のブラウザバージョンを調べてみたいと思います。

今回も `navigator.userAgent` を利用してブラウザバージョンを取得する方法について考えます。

ブラウザバージョンを調べる流れ

前のページと比べて、今回はちょっと手間が増えます。ブラウザを判別するために固有の文字列を `indexOf()` で探しますが、それがそのまま切り出し位置になりません。ということで開始位置取得にもう一度 `indexOf()` を使用することになります。

以下に私の環境での `navigator.userAgent` の値をもう一度記しておきます。これを見てそれぞれのブラウザの判別文字列、開始文字列、終了文字列を表にすると以下のようになり

ます。

Opera

Opera/9.80 (Windows NT 6.1; U; en) Presto/2.9.168 Version/11.50

Chrome

Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko)

Chrome/14.0.835.163 Safari/535.1

Safari

Mozilla/5.0 (Windows; U; Windows NT 6.1; ja-JP) AppleWebKit/533.21.1 (KHTML, like Gecko) Version/5.0.5 Safari/533.21.1

ブラウザ 判別文字列 開始文字列 終了文字列

Opera	Opera	Version	最後 (文字列長)
-------	-------	---------	-----------

Chrome	Safari○ Chrome○	Chrome	半角スペース
--------	--------------------	--------	--------

Safari	Safari○ Chome×	Version	半角スペース
--------	-------------------	---------	--------

ではこの表を元にスクリプトを組んでみましょう。

ブラウザ判別スクリプト

では、以下のスクリプトを HTML の BODY 内に記入してみてください。

```
<script>
```

```
var nvUA = navigator.userAgent;
```

```
var cutSt,cutEd; //バージョンを切り出す位置情報を取得
```

```
var bwVer; //バージョンを格納
```

```
//Opera か調べる
```

```
if ( nvUA.indexOf("Opera") != -1 )
```

```
{
```

```
    //開始位置の指標「Version」の位置を取得、終了位置は文字列長
```

```
    cutSt = nvUA.indexOf("Version");
```

```
    cutEd = nvUA.length;
```

```
    //バージョンの切り出し
```

```
    bwVer = nvUA.substring(cutSt+8,cutEd);
```

```
//頭に「Opera」を付けてバージョンを書き出す
document.write("Opera "+ bwVer);
}
```

```
//「Safari」が含まれるか調べる
if ( nvUA.indexOf("Safari") != -1 )
{
    //「Chrome」が含まれるか調べる
    if ( nvUA.indexOf("Chrome") != -1 )
    {
        cutSt = nvUA.indexOf("Chrome");
        cutEd  = nvUA.indexOf(" ",cutSt);

        bwVer =nvUA.substring(cutSt+7,cutEd);

        document.write("Google Chrome "+bwVer);
    }
    else
    {
        cutSt = nvUA.indexOf("Version");
        cutEd  = nvUA.indexOf(" ",cutSt);

        bwVer = nvUA.substring(cutSt+8,cutEd);

        document.write("Safari "+bwVer);
    }
}
```

</script>

<上記スクリプトの結果>

スクリプトの解説

上記スクリプトについて説明していきます。

変数の宣言

[変数](#)の宣言部分は[前項](#)と全く同じにしています。そちらを参考になさってください。

Opera か調べる

indexOf()を使って「Opera」があるかを調べています。文字列が存在していれば、[if 文](#)の中の処理を行います。

切り取り開始の指標となる「Version」の位置を、再び indexOf()で取得します。終了位置は文字列長で代用しています。文字列の長さを取得するのは [length](#) でした。

実際に切り取りたいのはバージョンの数字の部分だけです。それで変数 cutSt から「Version/」の 8 文字を加えた位置から [substring\(\)](#)で切り取ります。終了位置は変数 cutEd です。

最後に document.write()で、ブラウザ名とバージョンを書き出しています。

「Safari」が含まれるか調べる

次は Safari と Chrome を識別します。これは先に「Safari」を調べて、存在していれば「Chrome」を調べることで判別できました。「Chrome」があれば Chrome、無ければ Safari です。if 文を 2 回使ってこれらの文字列が含まれるかどうかで分岐しています。

「Chrome」が存在する場合

「Chrome」が含まれているならば、今度は切り取り位置を indexOf()で取得していきます。開始位置「Chrome」の位置を変数 cutSt に格納します。

終了位置は「半角スペース」ですが、半角スペースは沢山あります。ですから検索開始位置を変数 cutSt にして、そこから最初に出現する半角スペースの位置を取得し、変数 cutEd に格納します。

substring()で切り取る際は、切り取り開始位置を「Chrome/」の 7 文字分加えた所から切り取っています。

最後にブラウザ名とバージョンを document.write()で書き出します。

「Chrome」が存在しない場合

「Safari」があつて「Chrome」が無いなら、Safari ということになります。[else 文](#)の中で上の表に従って切り取り開始位置と終了位置を取得します。

切り取り開始位置は「Version」を検索します。終了位置は「半角スペース」ですが、こちらも検索開始位置を変数 cutSt にして間違った値を取得しないようにします。

実際に切り取る際には「Version/」の 8 文字を加えた位置から substring()で切り取ります。そして document.write()でブラウザ名とバージョンを書き出しています。