少し面倒くさいですが、このようにしてappendChildしていくやり方になります。例えば次のように、iframeオブジェクトに対してそのままappendChildする事はできません。

```
iframe.appendChild(elm);
```

あまり多くはありませんが、元ページのCSSに影響されてうまくデザインできない場合にはこのテクニックを使うのが楽なので、覚えておきましょう。

3-4-2 サイト内検索用のフォームを追加する

今自分が見ているサイト内を検索したい場合、そのサイトが提供している独自の検索システムを使うか、検索エンジンでドメインを指定して検索するか、どちらかしかありません。

独自の検索システムが用意してあれば楽なのですが、そうでない場合、わざわざ検索エンジンに行ってドメインを入力して…、という手順を踏む必要があります。これでは面倒で、気楽に検索する事はできません。

要するに、そんな面倒な仕事はGreasemonkeyに任せてしまいましょうと言う訳で、閲覧中のドメインを対象にした検索フォームを追加するユーザスクリプトです。

AddsSiteSearchByGoogle.user.js

ライセンス パブリックドメイン

```
// ==UserScript==
// @name        Adds Site Search By Google
// @include     *
// @exclude     http://www.google.tld/*
// @version     1.00
// ==/UserScript==

if(document.contentType != 'text/html') return;

var GM_infoPanel = document.createElement('div');

with(GM_infoPanel.style) {
  bottom  = 0;
  left    = 0;
```

```
  padding  = '2px';
  opacity  = 0;
  fontSize = 'x-small';
  color    = '#000000';
  backgroundColor = '#EEEEEE';
  border   = '1px solid #C0C0C0';

  zIndex   = 100;
  position = 'fixed';

  width    = '100px';
  cursor   = 'pointer';
}

GM_infoPanel.innerHTML = ' ';
GM_infoPanel.addEventListener('mouseover', makeForm, false);

document.body.appendChild(GM_infoPanel);

function makeForm() {
  var site = document.domain;

  this.style.opacity = '';
  this.style.width   = '';

  this.innerHTML =
    '<form action="http://www.google.com/search" method="get" '+
    'Accept-charset="UTF-8" style="margin:0px" target="_top">' +
    '<input type="hidden" name="as_sitesearch" value="' + site + '">'+
    '<input type="text" size="20" name="q">'+
    '<input type="submit" value="Site Search">'+
    '</form>'+
    '';

  this.removeEventListener('mouseover', makeForm, false);
}
```

解　説　このスクリプトをインストールして、ページの左下にマウスを持っていくと検索フォームが表示されます。検索したい文字列を入力してから「Site Search」ボタンを押すと、閲覧しているサイトのドメインを対象にGoogleで検索を行ないます（**図3-15**）。

図3-15

図3-16

Tips　初級編の「情報パネルを表示する」（P.079）で扱ったInfoPanelMouseoverを使って作られた単純なスクリプトです。

ポイントとして、form要素のパラメータで使用しているAccept-Charsetが重要な機能となっているのがおわかりでしょうか。通常、フォームで送信する

データは、閲覧中のページの文字コードになります。しかし、GoogleやYahoo!では受け取るデータをUTF-8で固定しているため、文字コードがUTF-8以外のページからデータを送信すると、文字化けが発生してしまいます。そこでAccept-Charsetの登場です。Accept-Charsetで文字コードを指定しておくと、閲覧中のページの文字コードが何であろうと、Accept-Charsetで指定した文字コードでデータを送信してくれるようになります。

特に我々日本人にとっては文字コードの問題は切っても切れない関係なので、この機能の存在は忘れてはいけません。ユーザスクリプトでformを作る場合は、なるべくAccept-Charsetを使って文字コードを指定した方が良いでしょう。

Googleだけだと何なので、少し変更してYahoo!に対応したスクリプトもご用意しました。

AddsSiteSearchByYahoo.user.js

ライセンス　パブリックドメイン

```
// ==UserScript==
// @name        Adds Site Search By Yahoo
// @include     *
// @exclude     http://search.yahoo.tld/*
// @version     1.00
// ==/UserScript==

if(document.contentType != 'text/html') return;

var GM_infoPanel = document.createElement('div');

with(GM_infoPanel.style) {
  bottom   = 0;
  left     = 0;
  padding  = '2px';
  opacity  = 0;
  fontSize = 'x-small';
  color    = '#000000';
  backgroundColor = '#EEEEEE';
  border   = '1px solid #C0C0C0';

  zIndex   = 100;
  position = 'fixed';

  width    = '100px';
  cursor   = 'pointer';
}

GM_infoPanel.innerHTML = ' ';
GM_infoPanel.addEventListener('mouseover', makeForm, false);

document.body.appendChild(GM_infoPanel);

function makeForm() {
  var site = document.domain;

  this.style.opacity = '';
  this.style.width   = '';

  this.innerHTML =
    '<form action="http://search.yahoo.co.jp/search" method="get" '+
    'Accept-charset="UTF-8" style="margin:0px" target="_top">' +
    '<input type="hidden" name="vs" value="' + site + '">'+
    '<input type="text" size="20" name="p">'+
    '<input type="submit" value="Site Search">'+
    '</form>'+
    '';

  this.removeEventListener('mouseover', makeForm, false);
}
```

解　説　先程のスクリプトと同じく、左下にマウスを持っていくと検索フォームが現れます。「Site Search」ボタンを押すとYahoo!でサイト内検索を行ないます。

Tips　フォームの送信先と、いくつかのパラメータを変更しただけです。検索エンジン毎にパラメータは異なりますが、大抵の検索エンジンでは、action属性と入力フィールドのname属性を変更するだけで対応可能です。