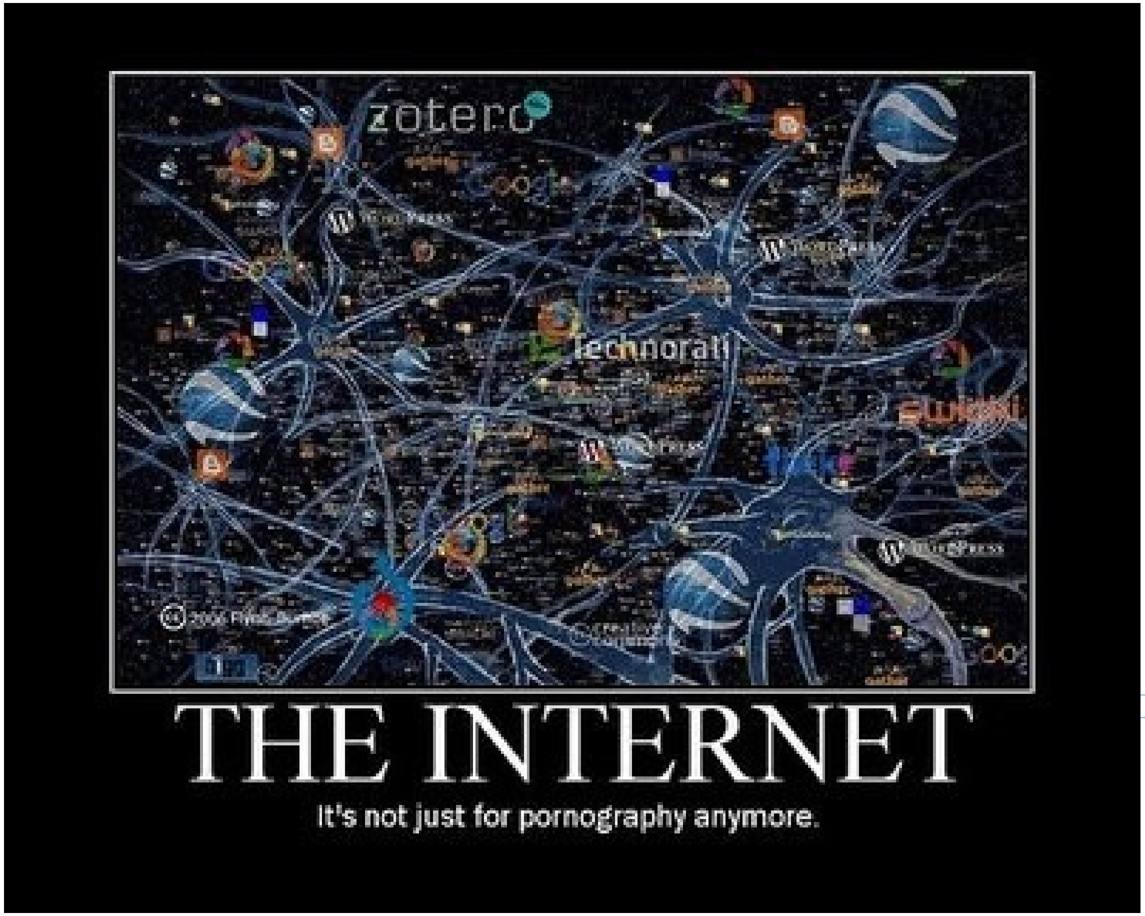
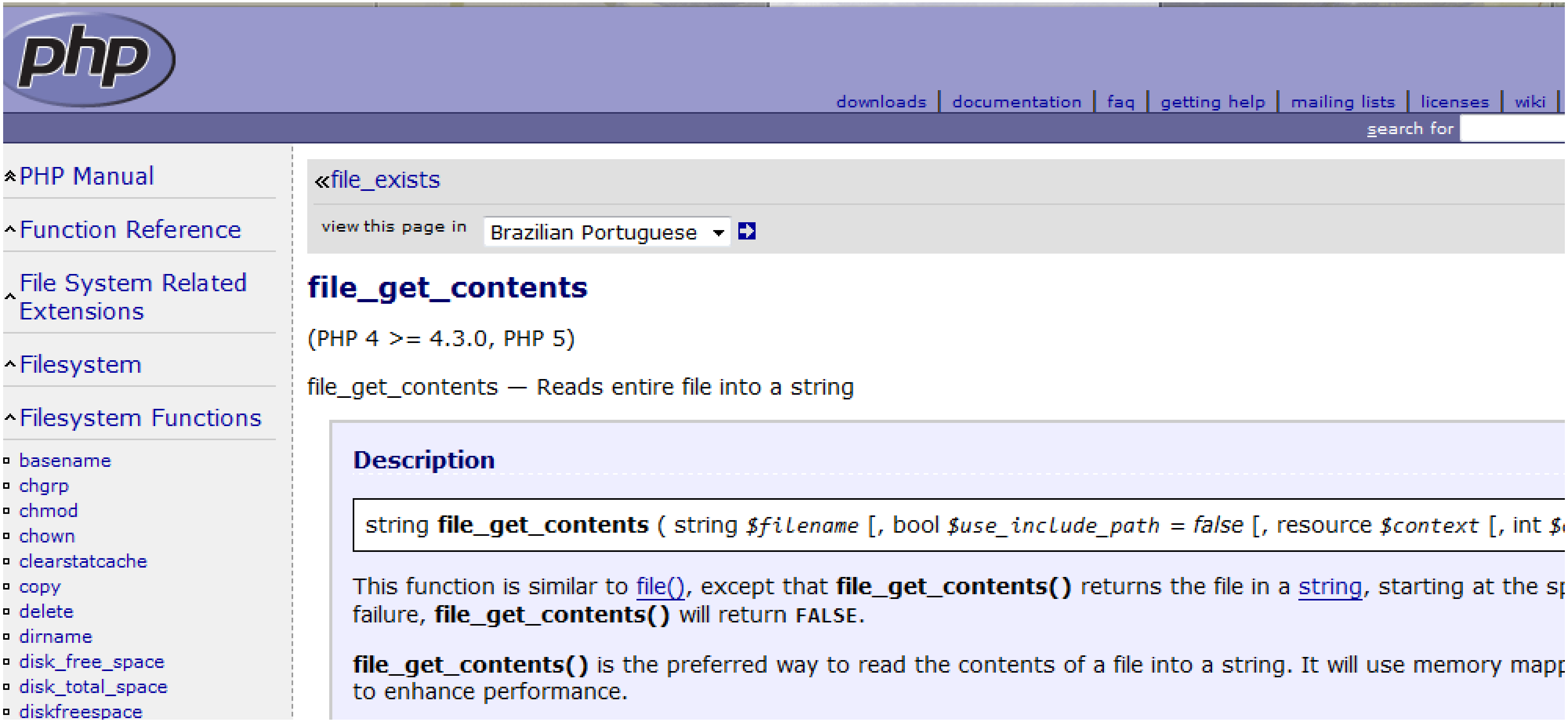
(How to make your own PHP powered)

Proxies, Web Crawlers, Search Engines



(this will be you)



file_get_contents

(PHP 4 >= 4.3.0, PHP 5)

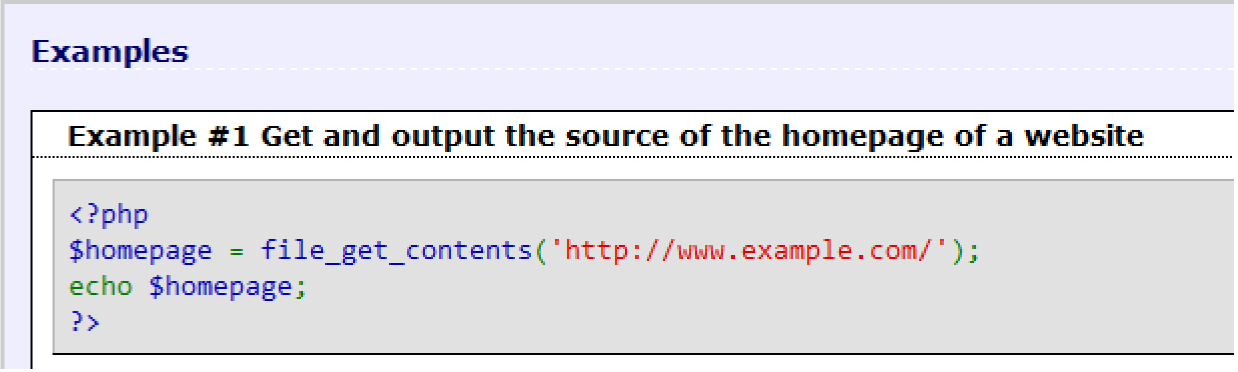
file_get_contents — Reads entire file into a string

Description

string **file_get_contents** (string *$filename* [, bool *$use_include_path* = *false* [, resource *$context* [, int *$*

This function is similar to file(), except that **file_get_contents()** returns the file in a string, starting at the sp failure, **file_get_contents()** will return FALSE.

file_get_contents() is the preferred way to read the contents of a file into a string. It will use memory mapp to enhance performance.

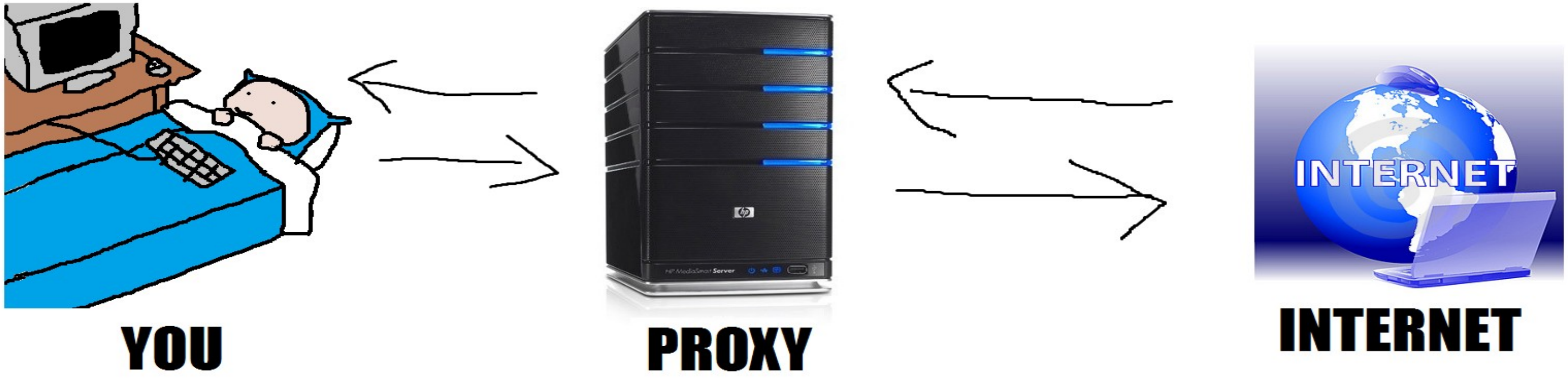


Examples

Example #1 Get and output the source of the homepage of a website

```
<?php
$homepage = file_get_contents('http://www.example.com/');
echo $homepage;
?>
```

<-- NICE

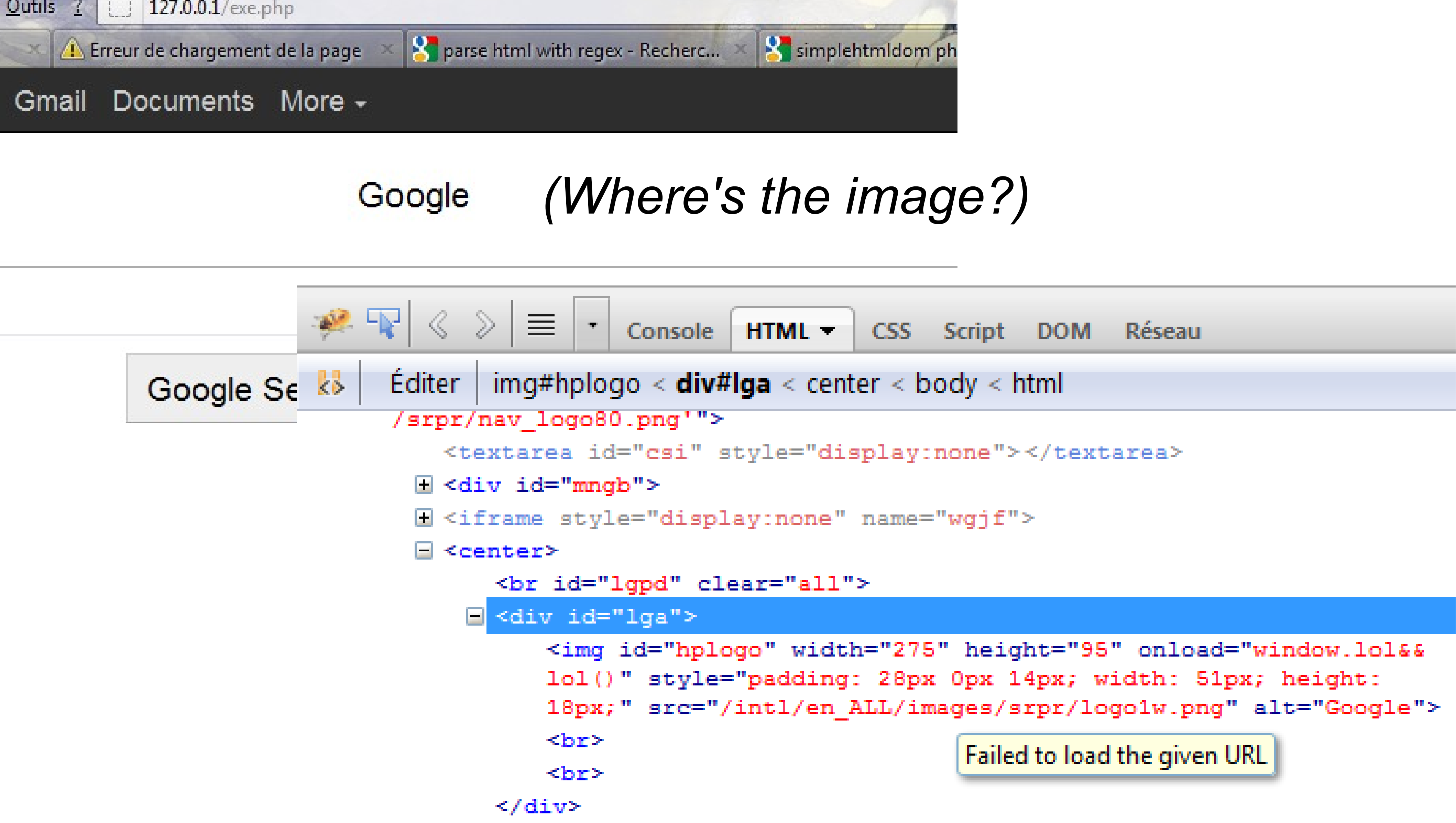


A proxy is a server we can choose to route our internet requests through. When we use one, our requests appear to be made from the proxy server instead.

We just implemented one!

Exercise: make a proxy "site" where the user can give a url to go to.

Using...
HTML forms?
URL parameters?
POST/GET ???


Outils
?
127.0.0.1/exe.php
Erreur de chargement de la page
parse html with regex - Recherc...
simplehtmldom ph
Gmail
Documents
More
Google
Google Se
Console
HTML
CSS
Script
DOM
Réseau
Éditer
img#hplogo < div#lga < center < body < html

(Where's the image?)

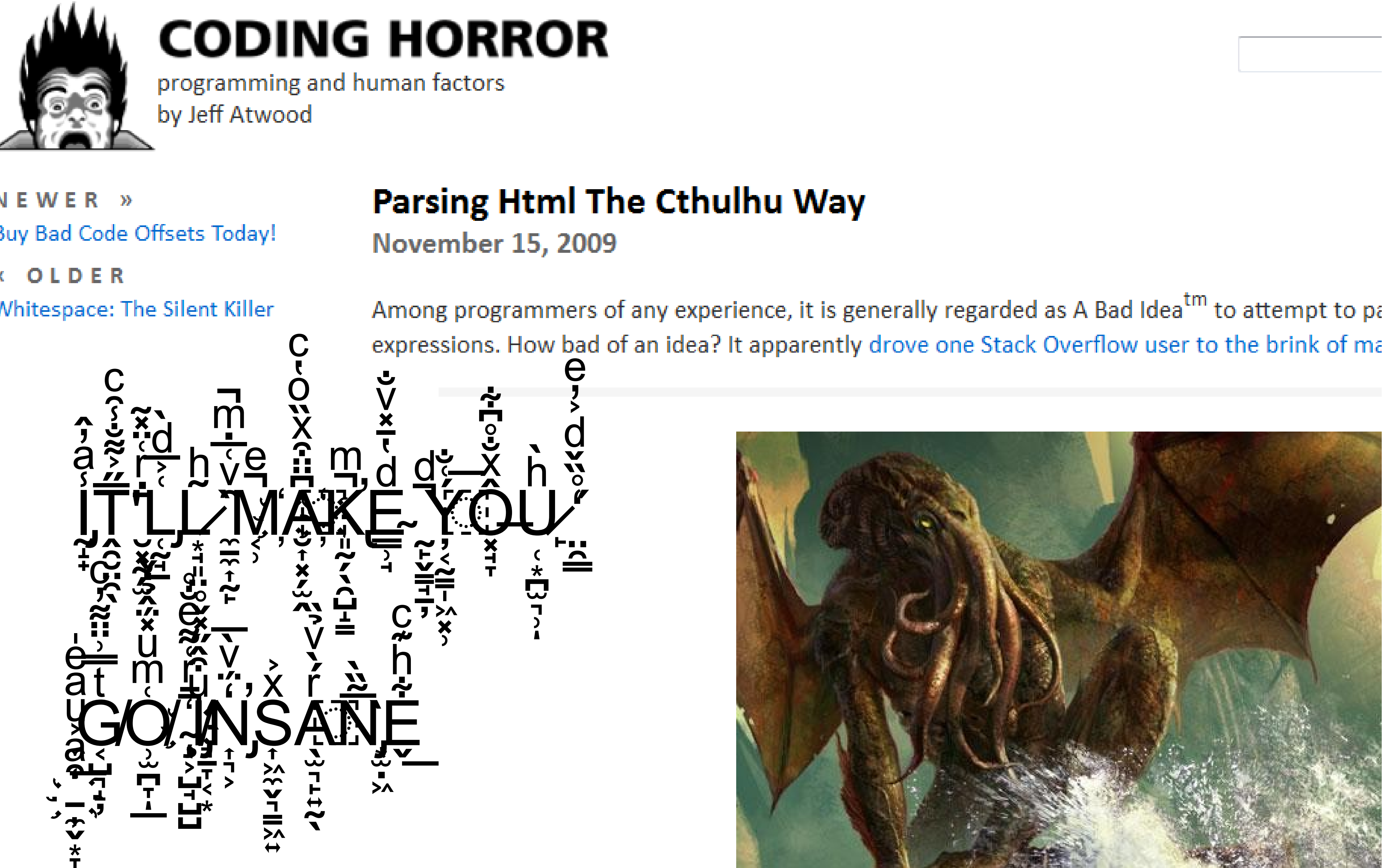
```
/srpr/nav_logo80.png'">
          <textarea id="csi" style="display:none"></textarea>
        ⊞ <div id="mngb">
        ⊞ <iframe style="display:none" name="wgjf">
        ⊟ <center>
              <br id="lgpd" clear="all">
            ⊟ <div id="lga">
                  <img id="hplogo" width="275" height="95" onload="window.lol&&
                  lol()" style="padding: 28px 0px 14px; width: 51px; height:
                  18px;" src="/intl/en_ALL/images/srpr/logo1w.png" alt="Google">
                  <br>
                  <br>
              </div>
```

Failed to load the given URL

We have a problem with relative urls. (It tries to load it from our php folder).

We'll need to fix the HTML string itself. How to do that?

NOT just with regexes! (it's actually impossible)



CODING HORROR

programming and human factors

by Jeff Atwood

NEWER »

Buy Bad Code Offsets Today!

« OLDER

Whitespace: The Silent Killer

Parsing Html The Cthulhu Way

November 15, 2009

Among programmers of any experience, it is generally regarded as A Bad Idea^{tm} to attempt to pa

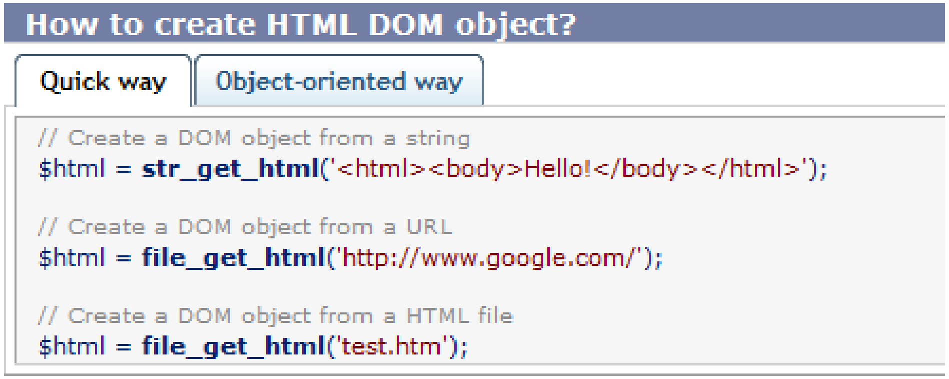
expressions. How bad of an idea? It apparently drove one Stack Overflow user to the brink of ma

IT'LL MAKE YOU GO INSANE

How to do it?

Use a library!

(I found a very simple and easy to use one here)
http://simplehtmldom.sourceforge.net/

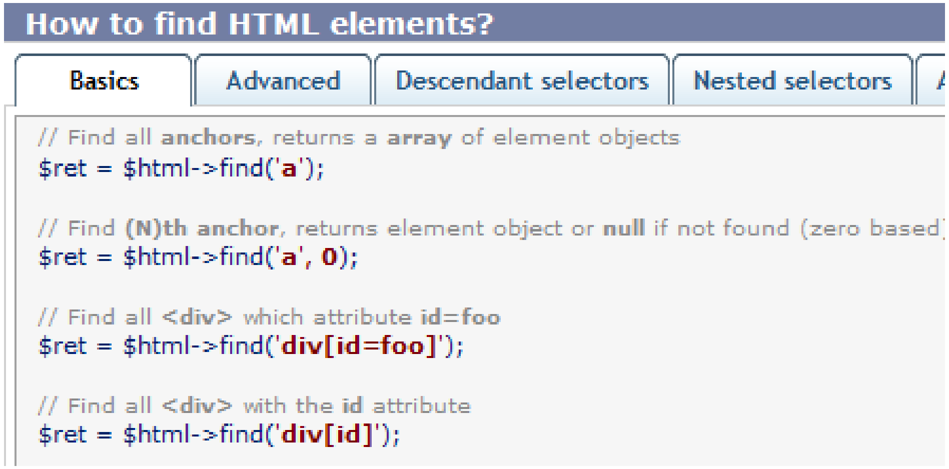


How to create HTML DOM object?

```
// Create a DOM object from a string
$html = str_get_html('<html><body>Hello!</body></html>');

// Create a DOM object from a URL
$html = file_get_html('http://www.google.com/');

// Create a DOM object from a HTML file
$html = file_get_html('test.htm');
```



How to find HTML elements?

```
// Find all anchors, returns a array of element objects
$ret = $html->find('a');

// Find (N)th anchor, returns element object or null if not found (zero based
$ret = $html->find('a', 0);

// Find all <div> which attribute id=foo
$ret = $html->find('div[id=foo]');

// Find all <div> with the id attribute
$ret = $html->find('div[id]');
```

http://sourceforge.net/projects/simplehtmldom/

Just download the .php file and place it in your folder.

Then be sure to call:

```
include("simple_html_dom");
```

in your own file.

A few things to keep in mind #1

Object-Oriented PHP

```
class MyClass {
  protected $field_name;

  public function method1() {
    //code goes here...
  }
}
```

What we need to know...

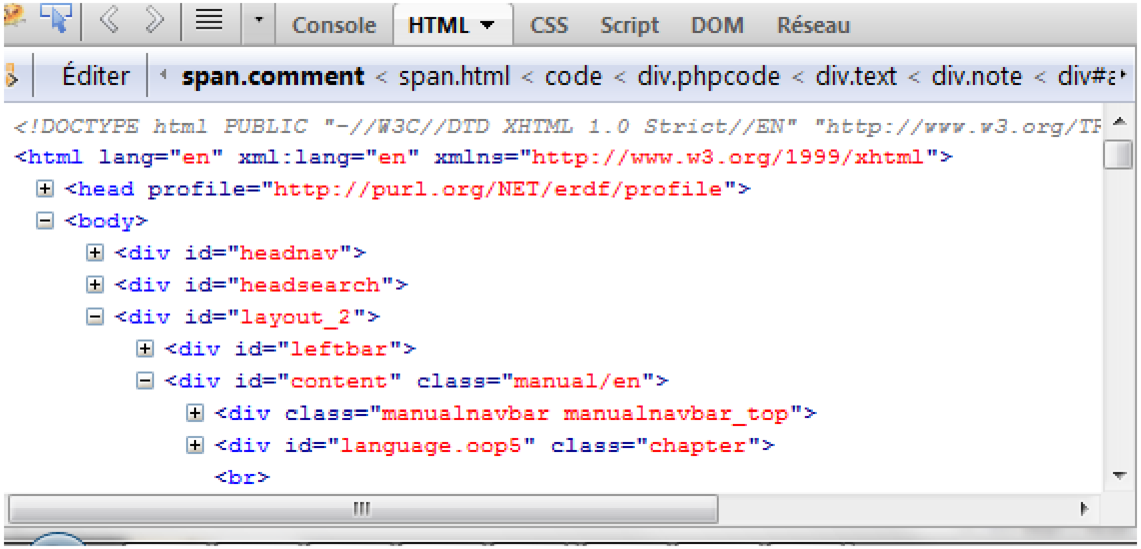
```
$instance = new MyClass();

$instance->method1();
```

(Equivalent of java's `instance.method();`)

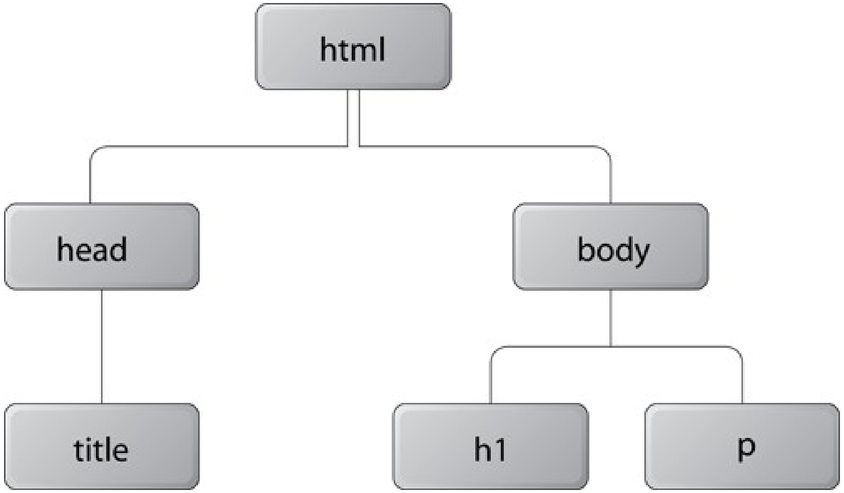
A few things to keep in mind #2

The DOM


Console
HTML
CSS
Script
DOM
Réseau
Éditer
span.comment < span.html < code < div.phpcode < div.text < div.note < div#a

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TF
<html lang="en" xml:lang="en" xmlns="http://www.w3.org/1999/xhtml">
 ⊞ <head profile="http://purl.org/NET/erdf/profile">
 ⊟ <body>
     ⊞ <div id="headnav">
     ⊞ <div id="headsearch">
     ⊟ <div id="layout_2">
         ⊞ <div id="leftbar">
         ⊟ <div id="content" class="manual/en">
             ⊞ <div class="manualnavbar manualnavbar_top">
             ⊞ <div id="language.oop5" class="chapter">
               <br>
```

HTML is loaded as a tree-like structure where nested elements are children.



Exercise #1:
Have our proxy grab every <img> tag, then display them.
(try this on amazon.com!)

Exercise #2:
Have our proxy grab every <a> tag.
Then, get each tag's href property.
IF the href is relative, make it absolute.
Finally, print every href.

Some syntax...

```
include("simple_html_dom.php");

$html = '<p id="lol">sup</p>';
$html = str_get_html($html);
$ret = $html->find('p');

foreach ($ret as $tag) {
   $id = $tag->id;
   echo $tag;
   echo $id;
}
```

PHP Function that will make relative URL absolute

(and do nothing to absolute URL)

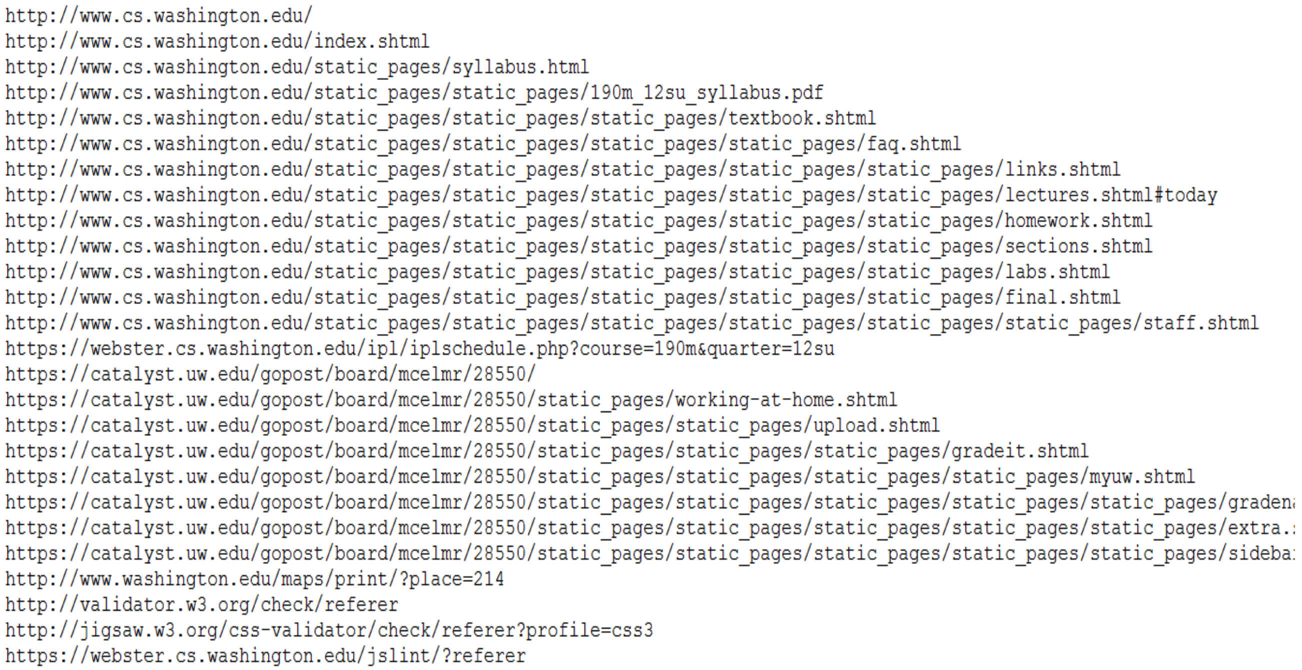
```php
/**
 * Function to convert relative URL to absolute given a base URL
 *
 * @param   string   the relative URL
 * @param   string   the base URL
 * @return  string   the absolute URL
 */
function rel2abs($rel, $base) {
    if (strlen($rel) == 0)
        return $base;
    else if (parse_url($rel, PHP_URL_SCHEME) != '')
        return $rel;
    else if ($rel[0] == '#' || $rel[0] == '?')
        return $base.$rel;

    extract(parse_url($base));

    $abs = ($rel[0] == '/' ? '' : preg_replace('#/[^/]*$#', '', $path))."/$rel";
    $re  = array('#(/\.?/)#', '#/(?!\.\.)[^/]+/\.\./#');

    for ($n = 1; $n > 0; $abs = preg_replace($re, '/', $abs, -1, $n));
        return $scheme.'://'.$host.str_replace('../', '', $abs);
}
```

Where to go from here?



```
http://www.cs.washington.edu/
http://www.cs.washington.edu/index.shtml
http://www.cs.washington.edu/static_pages/syllabus.html
http://www.cs.washington.edu/static_pages/static_pages/190m_12su_syllabus.pdf
http://www.cs.washington.edu/static_pages/static_pages/static_pages/textbook.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/faq.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/links.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/lectures.shtml#today
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/homework.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/sections.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/labs.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/final.shtml
http://www.cs.washington.edu/static_pages/static_pages/static_pages/static_pages/static_pages/static_pages/staff.shtml
https://webster.cs.washington.edu/ipl/iplschedule.php?course=190m&quarter=12su
https://catalyst.uw.edu/gopost/board/mcelmr/28550/
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/working-at-home.shtml
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/upload.shtml
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/static_pages/gradeit.shtml
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/static_pages/static_pages/myuw.shtml
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/static_pages/static_pages/static_pages/graden
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/static_pages/static_pages/static_pages/extra.
https://catalyst.uw.edu/gopost/board/mcelmr/28550/static_pages/static_pages/static_pages/static_pages/static_pages/sideba
http://www.washington.edu/maps/print/?place=214
http://validator.w3.org/check/referer
http://jigsaw.w3.org/css-validator/check/referer?profile=css3
https://webster.cs.washington.edu/jslint/?referer
```

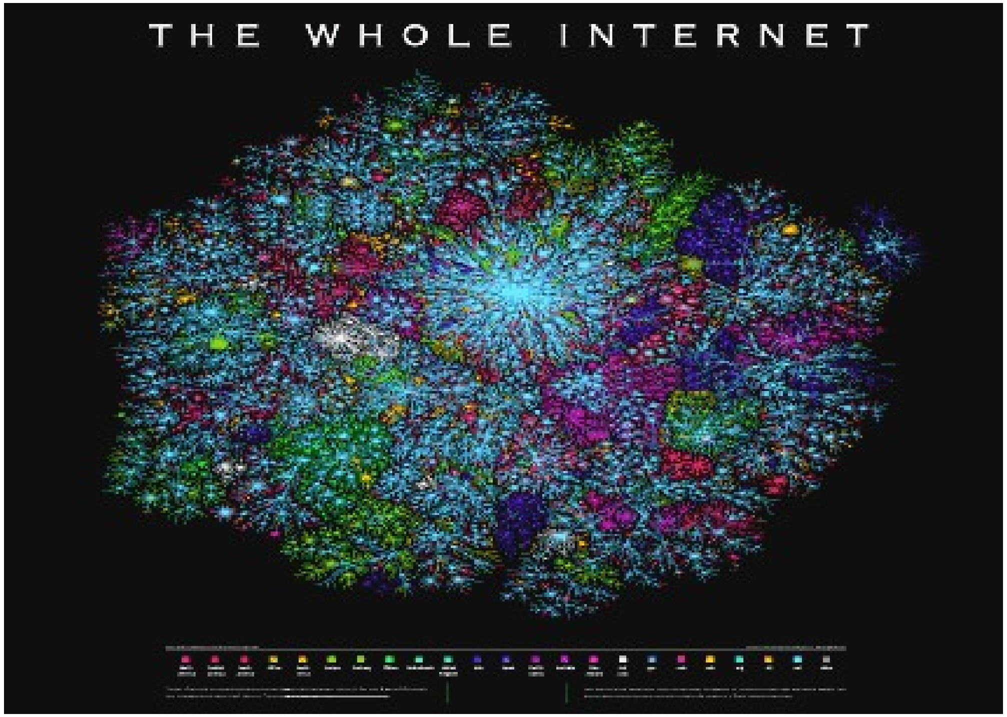
Every single one of these links is another URL which we can load. It'll also have a list of href's, all of which we can load as well. And so on...

This should sound like a **RECURSIVE** problem to you...

It only gets **EXPONENTIALLY BETTER** from here...

The CSE 190m frontpage has about 20 links.
Each of these links will have 20 links of their own.
Then those will have 20 links each, and so on.

Only 3 “levels” in, and that's already 20^3 + 20^2 + 20^1 or **8420** pages!
(We obviously can't do this by brute force)



There are over **1,000,000,000,000** pages on the internet.

How would you parse then categorize them all?
Google it!

What we **CAN** do

Write a relatively shallow (2 levels) search engine.

Apply regex on the html strings to find all occurrences of the query.

Some notes:

```
error_reporting(E_ERROR | E_PARSE);
```

(to disable warnings)

```
set_time_limit(0);
```

(to set unlimited time)

```
$htmlstring = file_get_contents($url);
```

check if the string is truthy before parsing

```
preg_match("/[^.]+shiny[^.]+/i",$htmlstring,$matches);
```

Matches will be in $matches array

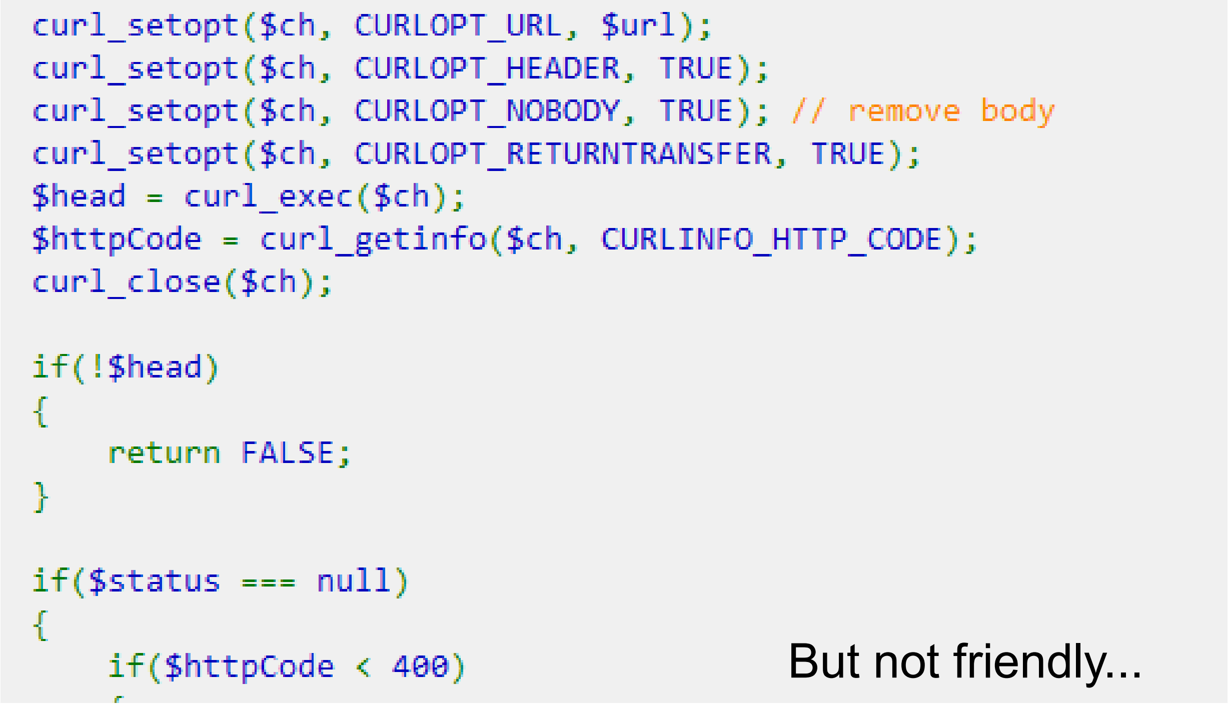
Problems?

file_get_contents() is incredibly slow and not suited for this task.

A fast web crawler is possible in PHP, but we'll need to use something else to make our requests.

(I recommend cURL, a PHP binding to a C library -- so you know it's fast!)

http://lu.php.net/manual/en/book.curl.php



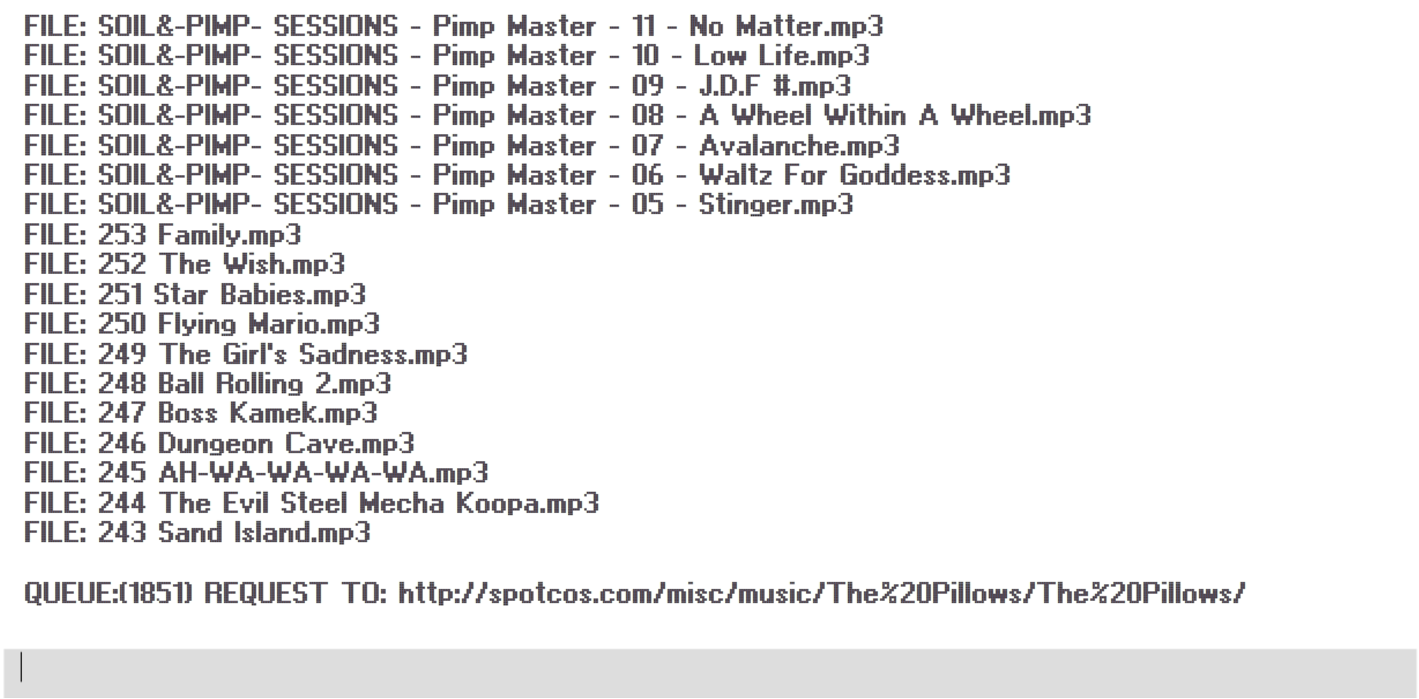
```
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_HEADER, TRUE);
curl_setopt($ch, CURLOPT_NOBODY, TRUE); // remove body
curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);
$head = curl_exec($ch);
$httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
curl_close($ch);

if(!$head)
{
    return FALSE;
}

if($status === null)
{
    if($httpCode < 400)
    {
```

But not friendly...

What can you do with web crawling?



```
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 11 - No Matter.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 10 - Low Life.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 09 - J.D.F #.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 08 - A Wheel Within A Wheel.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 07 - Avalanche.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 06 - Waltz For Goddess.mp3
FILE: SOIL&-PIMP- SESSIONS - Pimp Master - 05 - Stinger.mp3
FILE: 253 Family.mp3
FILE: 252 The Wish.mp3
FILE: 251 Star Babies.mp3
FILE: 250 Flying Mario.mp3
FILE: 249 The Girl's Sadness.mp3
FILE: 248 Ball Rolling 2.mp3
FILE: 247 Boss Kamek.mp3
FILE: 246 Dungeon Cave.mp3
FILE: 245 AH-WA-WA-WA-WA.mp3
FILE: 244 The Evil Steel Mecha Koopa.mp3
FILE: 243 Sand Island.mp3

QUEUE:(1851) REQUEST TO: http://spotcos.com/misc/music/The%20Pillows/The%20Pillows/
```

Check out StreamPlayer:
http://spotcos.com/misc/scrapeplayer/scrapeplayer.swf

To use, type:

```
load spotcos.com/misc
```

then

```
random
```